UNIVERSIDADE FEDERAL DO PARANÁ LEANDRO ALBERTO NOVAK

MÚLTIPLAS EXTRAPOLAÇÕES DE RICHARDSON PARA REDUZIR E ESTIMAR O ERRO DE DISCRETIZAÇÃO EM CONDUÇÃO DE CALOR

LEANDRO ALBERTO NOVAK

MÚLTIPLAS EXTRAPOLAÇÕES DE RICHARDSON PARA REDUZIR E ESTIMAR O ERRO DE DISCRETIZAÇÃO EM CONDUÇÃO DE CALOR

Tese apresentada como requisito parcial para obtenção do título de doutor em Engenharia Mecânica no Programa de Pós-Graduação em Engenharia Mecânica da UFPR, na área de concentração de Fenômenos de Transporte e Mecânica dos Sólidos

Orientador: Prof. Dr. Carlos Henrique Marchi.

TERMO DE APROVAÇÃO

LEANDRO ALBERTO NOVAK

MÚLTIPLAS EXTRAPOLAÇÕES DE RICHARDSON PARA REDUZIR E ESTIMAR O ERRO DE DISCRETIZAÇÃO EM CONDUÇÃO DE CALOR

Tese aprovada como requisito parcial para obtenção do título de doutor em Engenharia Mecânica no Programa de Pós-Graduação em Engenharia Mecânica na área de Fenômenos de Transporte e Mecânica dos Sólidos, Setor de Tecnologia, Universidade Federal do Paraná, pela seguinte banca examinadora:

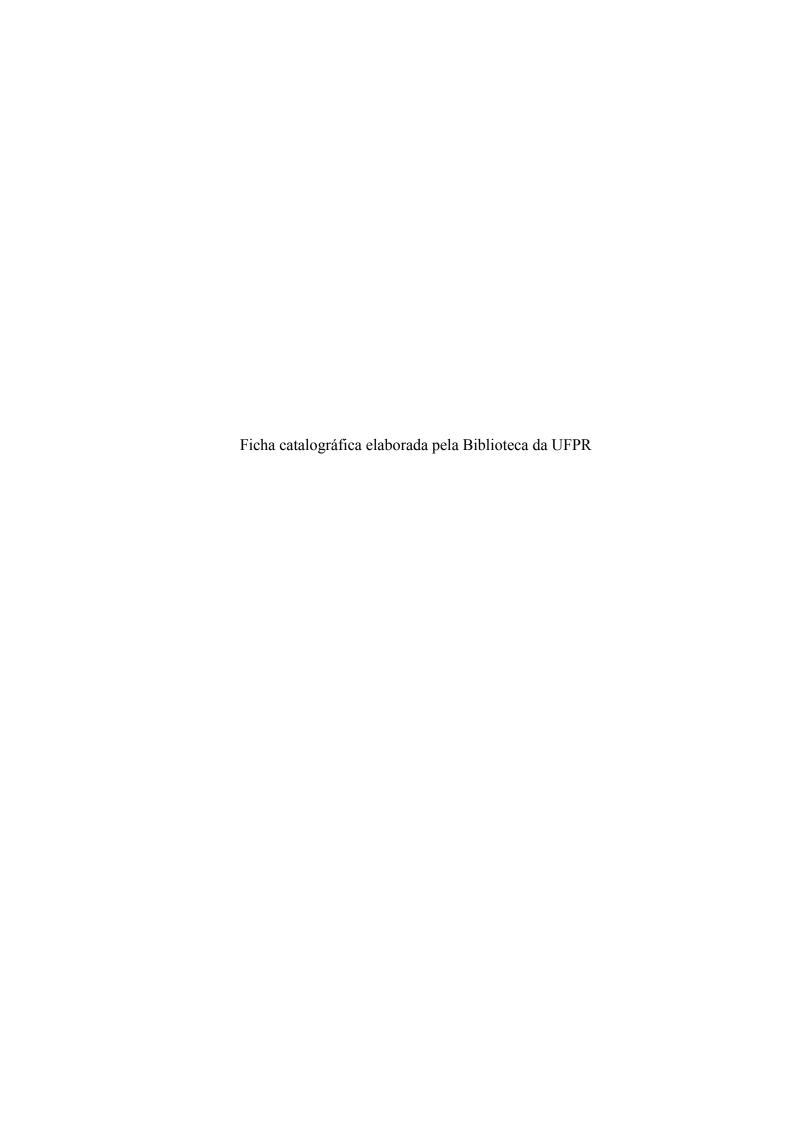
Prof. Dr. Carlos Henrique Marchi
Universidade Federal do Paraná (UFPR)

Prof. Dr^a. Viviana Cocco Mariani
Pontificia Universidade Católica do Paraná (PUCPR)

Prof. Dr. Rigoberto Eleazar Melgarejo Morales
Universidade Tecnológica Federal do Paraná (UTFPR)

Prof. Dr. Luciano Kiyoshi Araki
Universidade Federal do Paraná (UFPR)

Universidade Federal do Paraná (UFPR)



AGRADECIMENTOS

Aos meus Pais Ivonete e Élio Novak com grande carinho.

A minha esposa Karina pela paciência e compreensão.

Ao meu orientador com grande respeito e admiração.

Ao grupo do LENA.

A Educação.

A ciência.

RESUMO

O erro de discretização é a maior preocupação para um usuário de mecânica dos fluidos e transferência de calor computacional durante uma aplicação numérica. É aquele erro que ocorre da conversão das equações que regem os modelos físicos em um domínio contínuo para o domínio discreto do espaço. Ele é reduzido e a acurácia dos cálculos é aumentada quando o parâmetro de malha h tende ao contínuo devido à solução numérica ser sensível a este espaçamento. No entanto, este procedimento de redução do erro de discretização é inversamente proporcional ao custo computacional, isto é, quanto menor h, maior é a acurácia e maior será o custo computacional. Uma ferramenta capaz de melhorar a acurácia da solução numérica sem aumentar o custo computacional é a múltipla extrapolação de Richardson (MER). Esta ferramenta para ser empregada eficientemente na redução do erro de discretização precisa ser ainda avaliada, aperfeiçoada e generalizada para o uso em problemas em mecânica dos fluidos e transferência de calor devido a apresentar problemas de convergência em situações onde as soluções apresentam extremos locais e/ ou globais. Para avaliar, aperfeiçoar e generalizar a ferramenta MER foram utilizados dois problemas clássicos em transferência de calor computacional governados pela equação de Laplace bidimensional e pela equação de Poisson unidimensional. Para a equação de Laplace, o domínio de cálculo é quadrado e discretizado com malhas uniformes. São obtidos resultados para variáveis principais e secundárias com a temperatura no centro do domínio, média do campo de temperaturas, taxa de transferência de calor em dois contornos e norma do erro de discretização. Para todas as variáveis desejadas deste experimento são conhecidos as suas respectivas posições. A equação de Poisson unidimensional é discretizada com malha uniforme onde as variáveis desejadas são temperatura máxima e sua posição. É definido nesta tese o erro de posição que associado à interpolação e extrapolação de Richardson resulta em resultados numéricos extremamente acurados. Mostra-se, portanto, que MER reduz significativamente o erro de discretização nos problemas numéricos de condução de calor, o estimador de erro de Richardson funciona para resultados numéricos obtidos com MER e os resultados mais efetivos com MER são obtidos usando precisão quádrupla nos cálculos, reduzindo o erro de posição por meio de interpolação, maior número de extrapolações, maior número de malhas e ordens corretas do erro.

Palavras-chave: Erro de discretização. Solução numérica. Múltipla extrapolação de Richardson. Estimador de erro. Erro de posição.

ABSTRACT

The discretization error is the biggest concern for a user of fluid mechanics and heat transfer in a computational numerical application. Error that occurs is the conversion of the equations governing the physical models in a continuous domain to discrete domain space. It is reduced and the accuracy of the calculations is increased when the mesh parameter h tends to continued due to the numerical solution is sensitive to spacing. However, this procedure of reducing the discretization error is inversely proportional to the computational cost. A tool to improve the accuracy of the numerical solution without increasing the computational cost is a repeated Richardson extrapolation (RRE). This tool to be used effectively in reducing the discretization error has to be evaluated, refined and generalized for use on problems in fluid mechanics and heat transfer due to present convergence problems in situations where the solutions have extreme local and / or global. To assess, improve and generalize the RRE tool we used two classic problems in computational heat transfer governed by the Laplace equation for two-dimensional and one-dimensional Poisson equation. For the Laplace equation calculation domain is discretized with square and uniform meshes. Results are obtained for primary and secondary variables with temperature in the center of the field, the average temperature field, and rate of heat transfer in two contours and standard error of discretization. For all desired variables of this experiment are known to their respective positions. For the one-dimensional Poisson equation is discretized with uniform mesh where the variables are desired maximum temperature and position. It is defined in this thesis that the position error associated with interpolation and Richardson extrapolation results in extremely accurate numerical results. It shows therefore that RRE significantly reduces the discretization error, the error estimator Richardson works for numerical results obtained with MER and the MER with more effective results are obtained using quadruple precision in the calculations, reducing the position error by interpolation, extrapolation of many, many orders of knitwear and correct the error.

Keywords: Discretization error. Numerical solution. Repeated Richardson Extrapolation. Error estimation. Error position.

SUMÁRIO

1	INT	TROI	DUÇÃO	21
	1.1.	O P	ROBLEMA	25
	1.2.	DE	FINIÇÕES	26
	1.2.	.1	Erros Numéricos	26
	1.2.	.2	A redução do Erro de Discretização com MER	28
	1.3.	DE	FINIÇÃO DOS OBJETIVOS	29
	1.3.	.1	Objetivo Geral	29
	1.3.	.2	Objetivos específicos	30
	1.4.	MO	TIVAÇÃO	30
	1.5.	LIN	MITES DA PESQUISA	31
2	FU	NDA	MENTOS	32
	2.1.	ASI	PECTOS PRELIMINARES	32
	2.1.	.1	Funções de uma Única Variável	33
	2.1.	.2	Funções de Várias Variáveis	33
	2.1.	.3	Funções Analíticas e Série de Taylor	34
	2.2.	DIN	NÂMICA DAS SOLUÇÕES E SUAS CARACTERÍSTICAS	37
	2.3.	AC	ELERAÇÃO DE CONVERGÊNCIA	44
	2.4.	TÓ	PICOS PARA EXTRAPOLAÇÃO DE RICHARDSON	46
	2.4.	.1	Aproximações Numéricas e Ordem do Erro a Priori	50
	2.4.	.2	Erro de Discretização e sua Estimativa	52
	2.5.	EX	TRAPOLAÇÃO DE RICHARDSON	54
	2.6.	GE	NERALIZAÇÃO DA EXTRAPOLAÇÃO DE RICHARDSON	57
3	ME	TOE	OOLOGIA PARA APLICAÇÃO DE MÚLTIPLAS EXTRAPOLAÇÕES	DE
R	ICHA	RDS	ON - MER	58
	3.1.	SO	LUÇÕES NUMÉRICAS COM MÚLTIPLAS EXTRAPOLAÇÕES	DE
	RICH		DSON	
	3.2.	OR	DEM DO ERRO	60
	3.3.	EST	TIMADOR DE ERRO DE RICHARDSON	64

	3.4. N	METODOLOGIA PARA CORREÇÃO DOS DADOS PARA MER	64
	3.4.1	Função Contínua e Função Discreta	65
	3.4.2	Discretização do Domínio de Cálculo	68
	3.4.3	Efeito da Discretização no Ponto de Máximo ou Mínimo	69
	3.4.4	Metodologia para Correção dos Dados de Entrada para MER	70
	3.4.5	Erro de Posição da Informação	70
	3.4.6	Minimização do Erro de Posição com Interpolação Polinomial	74
	3.4	.6.1 Interpolação Polinomial Quadrática	74
	3.4	.6.2 Interpolação Polinomial de Ordem n > 2	75
	3.5.	CONCLUSÃO	76
4	EQUA	AÇÃO DE LAPLACE BIDIMENSIONAL	78
	4.1. N	MODELO MATEMÁTICO	79
	4.2. N	MODELO NUMÉRICO	81
	4.3. N	METODOLOGIA	83
	4.4. F	RESULTADOS	84
	4.4.1	Efeito da Precisão dos Cálculos das Soluções Numérica	
	4.4.1		
	4.4.1 Aperf 4.4.2	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estim	as como
	4.4.1 Aperf 4.4.2	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER	as como
	4.4.1 Aperf 4.4.2 Richa 4.4.3	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimardson Verificação de Ordens de Erro e as Consequências para MER ao Adot	as como 87 ativa com 90 tar Ordens
	4.4.1 Aperf 4.4.2 Richa 4.4.3 Errad	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimardson Verificação de Ordens de Erro e as Consequências para MER ao Adotas do Erro	as como 87 ativa com 90 tar Ordens
	4.4.1 Aperf 4.4.2 Richa 4.4.3 Errad 4.5. (Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimardson Verificação de Ordens de Erro e as Consequências para MER ao Adotas do Erro CONCLUSÃO	as como 87 ativa com 90 tar Ordens 95
5	4.4.1 Aperf 4.4.2 Richa 4.4.3 Errad 4.5. C EQUA	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimadon Verificação de Ordens de Erro e as Consequências para MER ao Adotas do Erro CONCLUSÃO	as como 87 ativa com 90 tar Ordens 95 97
5	4.4.1 Aperf 4.4.2 Richa 4.4.3 Errad 4.5. C EQUA 5.1. M	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimadon Verificação de Ordens de Erro e as Consequências para MER ao Adot as do Erro CONCLUSÃO	as como 87 ativa com 90 tar Ordens 95 97 99
5	4.4.1 Aperf 4.4.2 Richa 4.4.3 Errad 4.5. C EQUA 5.1. M 5.2. M	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimardson Verificação de Ordens de Erro e as Consequências para MER ao Adot as do Erro CONCLUSÃO	as como 87 ativa com 90 tar Ordens 95 97 99 101
5	4.4.1 Aperf 4.4.2 Richa 4.4.3 Errad 4.5. C EQUA 5.1. M 5.2. M	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimardson Verificação de Ordens de Erro e as Consequências para MER ao Adot as do Erro CONCLUSÃO	as como 87 ativa com 90 tar Ordens 95 97 99 101 103
5	4.4.1 Aperf 4.4.2 Richa 4.4.3 Errad 4.5. C EQUA 5.1. M 5.2. M	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimardson Verificação de Ordens de Erro e as Consequências para MER ao Adot as do Erro CONCLUSÃO	as como 87 ativa com 90 tar Ordens 95 97 99 101 103
5	4.4.1 Aperf 4.4.2 Richa 4.4.3 Errad 4.5. C EQUA 5.1. M 5.2. M	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimardson Verificação de Ordens de Erro e as Consequências para MER ao Adot as do Erro CONCLUSÃO	as como 87 ativa com 90 tar Ordens 95 97 99 101 103
5	4.4.1 Aperf 4.4.2 Richa 4.4.3 Errad 4.5. C EQUA 5.1. M 5.2. M 5.3. M 5.4. F	Efeito da Precisão dos Cálculos das Soluções Numérica feiçoamento para MER A Efetiva Redução do Erro de Discretização com MER e sua Estimardson Verificação de Ordens de Erro e as Consequências para MER ao Adotas do Erro CONCLUSÃO AÇÃO DE POISSON UNIDIMENSIONAL MODELO MATEMÁTICO MODELO NUMÉRICO METODOLOGIA RESULTADOS OBTIDOS	as como 87 ativa com 90 tar Ordens 95 97 99 101 103

	5.4.4	Aplicação de MER por meio de interpolação de décima ordem	121
4	5.4.5	Comparação dos resultados	124
4	5.4.6	Redução e estimativa do erro	126
5.5	. CO	NCLUSÃO	127
6 (CONCL	USÕES E RECOMENDAÇÕES	128
6.1	. ME	TODOLOGIA DA PESQUISA	128
6.2	. CO	NCLUSÃO GERAL	128
6.3	. REC	COMENDAÇÕES PARA TRABALHOS FUTUROS	129
A PÊ	NDICE	A - Valores analíticos das variáveis de interesse	135
A PÊ	NDICE	B - Tabelas dos Gráficos do Cap. 4 e Cap. 5	138
A PÊ	NDICE	C - Teste de Polinômios	197
APÊ]	NDICE	D - Programa Diferenças Divididas	211

LISTA DE FIGURAS

Figura 1. 1 - Métodos de Solução de Problemas (Adaptado de Marchi, 2001)23
Figura 2.1 - Malha unidimensional com 3 nós e comprimento $L_x = 1$.
Figura 2.2 - Malha unidimensional com 5 nós e comprimento $L_x = 1$.
Figura 2.3 - Malha unidimensional com 9 nós e comprimento $L_x = 1$
Figura 2.4 - Efeito qualitativo da redução do tamanho da manha h em função do módulo do
erro para um determinado ponto no domínio de cálculo
Figura 2.5 - Malha com <i>25</i> nós
Figura 2.6 - Malha com 81 nós, $r = 2$.
Figura 2.7 - Malha com 36 nós e $r = 1,25$
Figura 2.8 - Efeito qualitativo da redução do tamanho da malha h em função da solução
numérica para um determinado ponto no domínio de cálculo
Figura 2.9 - Malha unidimensional com cinco nós
Figura 3.1 – Função contínua e pontos de inflexão.
Figura 3.2 - Discretização do domínio [0,L] em N elementos
Figura 3.3 - Diferenciação entre a função $f(x)$ e função $g(x)$.
Figura 3.4 - Diferenciação entre a função $f(x)$ e função $g(x)$
Figura 3.5 - Solução analítica em diversas malhas para geração da função $g(x)$
Figura 3.6 - Solução analítica, numérica e ponto crítico.
Figura 4.1 - Grade quadrada representando uma placa
Figura 4.2 - Malha bidimensional com 25 nós.
Figura 4.3 - Comparativo entre o módulo do erro em precisão dupla e quádrupla para as
variáveis T_C , T_2 , T_m , L , Q_e , Q_n versus tamanho da malha (h)
Figura 4.4 - Módulo do erro da solução numérica das variáveis T_C , T_2 , T_m , L , Q_e , Q_n 91
Figura 4.5 - Módulo do erro da solução numérica das variáveis T_C , T_2 , T_m , L , Q_e , Q_n 93
Figura 4.6 - Erro (E) das variáveis T_C , T_2 , T_m , L , Q_e , Q_n
Figura 4.7 - Ordem efetiva (p _E) para Te versus o tamanho da Malha (h) e número de
extrapolações (m).
Figura 4.8 - Ordem Aparente (p _u) <i>versus</i> o tamanho da Malha (h) e número de extrapolações
(m)96
Figura 4.9 - Efeito da utilização da ordem errada de p _m
Figura 5.1 – Malha 5x5 - Laplace 2D
Figura 5.2 - Malha 9x9 - Laplace 2D

Figura 5.3 -Temperatura T em função do x	103
Figura 5.4 - Malha unidimensional uniforme.	104
Figura 5.5 – Oscilação da localização do ponto de máximo discreto \mathbf{M}^{D} em diver	rsas malhas
em função do ponto de máximo contínuo M ^C	106
Figura 5.6 - Módulo do erro <i>versus h</i> para a variável T_{max} sem interpolação $p_L=2$ e	$dp_L=2.108$
Figura 5.7 - Módulo do erro <i>versus h</i> para a variável x_{max} sem interpolação $p_L=2$ e	$dp_L=2.108$
Figura 5.8 - Módulo do erro <i>versus h</i> para a variável T_{max} sem interpolação p_L =2 e	$dp_L=1.109$
Figura 5.9 – Planejamento para coincidir o ponto de máximo $M^{\rm C}$ com um nó $M^{\rm D}$ e	em diversas
malhas.	112
Figura 5.10 – Módulo do erro <i>versus h</i> para a variável T_{max} sem interpolação p_L	=2 e dp _L =2
sem erro de posição.	113
Figura 5.11 – Ordem efetiva (p_E) para T_{max} versus o tamanho da Malha (h) e	número de
extrapolações (m).	114
Figura 5.12 - Esquema de interpolação dos dados obtidos por Poisson_1Dp_3p1_6-	4BITS .115
Figura 5.13 - Módulo do erro $$ versus $$ h para a variável $$ $$ T $_{m\acute{a}x}$ para interpolação $$	de ordem 2
$com p_L=2 e dp_L=2.$	117
Figura 5.14 - Módulo do erro versus h para a variável $x_{máx}$ para interpolação de or p_L =2 e dp_L =2	
Figura 5.15 - Ordem efetiva (p_E) para T_{max} (a) e x_{max} (b) <i>versus</i> o tamanho da M	
número de extrapolações (m) para interpolação de ordem 2	
Figura 5.16 - Módulo do erro versus h para a variável T _{má} x para interpolação o	
$com p_L=2 e dp_L=2.$	
Figura 5.17 - Módulo do erro versus h para a variável $x_{máx}$ para interpolação de or	
p_L =2 e d p_L =2	
Figura 5.18 - Ordem efetiva (p_E) para T_{max} (a) e x_{max} (b) <i>versus</i> o tamanho da M	
número de extrapolações (m) para interpolação de ordem 4	
Figura 5.19 - Módulo do erro versus h para a variável T _{má} x para interpolação d	
$com p_L=2 e dp_L=2.$	
Figura 5.20 - Módulo do erro versus h para a variável $x_{máx}$ para interpolação de	
$com p_L=2 e dp_L=2$	
Figura 5.21 - Ordem efetiva (p_E) para T_{max} (a) e x_{max} (b) <i>versus</i> o tamanho da M	
número de extrapolações (m) para interpolação de ordem 10	
Figura 5.22 - Módulo do erro versus h para a variável $T_{máx}$ para diversas interpo	
p_L =2 e d p_L =2	
1 = 1 =	_

Figura 5.23 - Módulo do erro	versus h para a variável	x _{máx} para diversas	interpolações com
p _L =2 e dp _L =2			125

LISTA DE TABELAS

Tabela 2.1 – Valores previstos para as ordens verdadeiras e assintóticas (MARCHI, 2001)53
Tabela 3.1 – Índices das soluções numéricas sem extrapolação
Tabela 3.2 – Índices das Múltiplas Extrapolações de Richardson
Tabela 4.1 – Valores numéricos obtidos para as variáveis $T_{\rm C},~T_{\rm 2},~T_{\rm m},~Q_{\rm e}$ e $Q_{\rm n}$ com precisão
dupla84
Tabela 4.2 - Valores numéricos obtidos para as variáveis $T_{\rm C},~T_{\rm 2},~T_{\rm m},~Q_{\rm e}$ e $Q_{\rm n}$ com precisão
quádrupla85
Tabela 4.3 – Iterações Externas para atingir o erro de arredondamento de máquina
Tabela 4.4 – Cálculo da diferença entre precisão dupla e quádrupla para a variável $T_{\rm c}$ 88
Tabela 4.5 – Análise do Custo Computacional para a variável T_c
Tabela 4.6 - Comparação entre as ordens do erro de dicretização obtidas a priori e a
posteriori97
Tabela 5.1 – Tendência da Ordem efetiva (P_{E}) para a variável T_{max} de Eh e Emer em função
de h
Tabela 5.2 – Tendência da Ordem efetiva (P_{E}) para a variável x_{max} de Eh e Emer em função
de h
Tabela $5.4-$ Redução do erro para malhas fixas para a variável x_{max} com interpolação de $10^{\rm a}$
ordem
Tabela 5.5 – Redução de nós de malha para erros fixos para a variável x_{max} com interpolação
de 10 ^a ordem
Tabela B.1 – Ordem aparente (p _U) de T _C versus tamanho da malha (h) e número de
extrapolações (m) Fig. 4.3 (a)
Tabela B.2 – Ordem aparente (p_U) de T_2 versus tamanho da malha (h) e número de
extrapolações (m) Fig. 4.3 (b)
Tabela B.3 – Ordem aparente (p_U) de T_m versus tamanho da malha (h) e número de
extrapolações (m) Fig. 4.3 (c)
Tabela $B.4$ – Ordem aparente (p_U) de L versus tamanho da malha (h) e número de
extrapolações (m) Fig. 4.3 (d)
Tabela $B.5$ – Ordem aparente (p_U) de Q_e versus tamanho da malha (h) e número de
extrapolações (m) Fig. 4.3 (e)
Tabela B.6 - Ordem aparente (p _U) de Q _n versus tamanho da malha (h) e número de
extrapolações (m) Fig. 4.3 (f)

Tabela B.7 – Módulo do erro da solução numérica da Variável Tc versus tamanho da malha
h Fig. 4.4 (a)
Tabela B.8 – Módulo do erro da solução numérica da Variável T2 versus tamanho da malha
<i>h</i> Fig. 4.4 (b)
Tabela B.9 – Módulo do erro da solução numérica da Variável <i>T</i> m versus tamanho da malha
<i>h</i> Fig. 4.4 (c)
Tabela B.10 — Módulo do erro da solução numérica da Variável ${\cal L}$ versus tamanho da malha
<i>h</i> Fig. 4.4 (d)
Tabela B.11 – Módulo do erro da solução numérica da Variável <i>Qe</i> versus tamanho da
malha h Fig. 4.4 (e)
Tabela B.12 – Módulo do erro da solução numérica da Variável Qn versus tamanho da
malha h Fig. 4.4 (f)
Tabela B.13 – Módulo do erro da solução numérica da Variável T c versus tamanho da malha
<i>h</i> Fig. 4.5 (a)
Tabela B.14 – Módulo do erro da solução numérica da Variável <i>T</i> 2 versus tamanho da malha
<i>h</i> Fig. 4.4 (b)
Tabela B.15 - Módulo do erro da solução numérica da Variável Tm versus tamanho da
malha h Fig. 4.5 (c)
Tabela B.16 – Módulo do erro da solução numérica da Variável L versus tamanho da malha
h Fig. 4.5 (d)
Tabela B.17 - Módulo do erro da solução numérica da Variável Qe versus tamanho da
malha <i>h</i> Fig. 4.5 (e)
Tabela B.18 - Módulo do erro da solução numérica da Variável <i>Qn</i> versus tamanho da
malha h Fig. 4.5 (f)
Tabela B.19 – Erro (E) das variáveis T _C versus tamanho da malha (h) e número de
extrapolações (m) em Real*16 Fig. 4.6 (a)
Tabela B.20 - erro (e) das variáveis T2 versus tamanho da malha (h) e número de
extrapolações (m) em Real*16 Fig. 4.6 (b)
Tabela B.21 – erro (e) das variáveis T _m versus tamanho da malha (h) e número de
extrapolações (m) em Real*16 Fig. 4.6 (c)
Tabela B.22 - erro (e) das variáveis L versus tamanho da malha (h) e número de
extrapolações (m) em Real*16 Fig. 4.6 (d)
Tabela B.23 - erro (e) das variáveis Qe versus tamanho da malha (h) e número de
extrapolações (m) em Real*16 Fig. 4.6 (e)

Tabela B.24 – erro (e) das variáveis Q _n versus tamanho da malha (h) e número de
extrapolações (m) em Real*16 Fig. 4.6 (f)
Tabela B.25 - Ordem efetiva (p _e) versus o tamanho da Malha (h) e número de extrapolações
(m). Fig. 4.7
Tabela B.26 - Ordem Aparente (p _u) versus o tamanho da Malha (h) e número de
extrapolações (m). Fig. 4.8 (a)
Tabela B.27 – Ordem Aparente (p _u) versus o tamanho da Malha (h) e número de
extrapolações (m). Fig. 4.8 (b)
Tabela B.28 – Ordem Aparente (p _u) versus o tamanho da Malha (h) e número de
extrapolações (m). Fig. 4.8 (c)
Tabela B.29 – Ordem Aparente (p _u) versus o tamanho da Malha (h) e número de
extrapolações (m). Fig. 4.8 (d)
Tabela B.30 – Ordem Aparente (p _u) versus o tamanho da Malha (h) e número de
extrapolações (m). Fig. 4.8 (e)
Tabela B.31 – Ordem Aparente (p _u) versus o tamanho da Malha (h) e número de
extrapolações (m). Fig. 4.8 (f)
Tabela B.32 – EEfeito da utilização da ordem errada de p _m . Fig. 4.9
Tabela B.33 – Módulo do erro versus h para a variável T _{max} sem interpolação p _L 2 e dp _L =2,
Fig. 5.6
Tabela B.34 – Módulo do erro versus h para a variável x_{max} sem interpolação $p_L 2$ e $dp_L = 2$,
Fig. 5.7
Tabela B.35 − Módulo do erro versus h para a variável T _{max} sem interpolação p _L 2 e dp _L =1,
Fig. 5.8
Tabela B.36 – Módulo do erro versus h para a variável T _{max} sem interpolação p _L 2 e dp _L =2,
Fig. 5.10
Tabela B.37 – Ordem aparente (p_E) de T_{max} versus tamanho da malha (h) e número de
extrapolações (m) Fig. 5.11
Tabela B.38 – Ordem aparente (p_E) de T_{max} versus tamanho da malha (h) e número de
extrapolações (m) Fig. 5.11
Tabela B.39 – Módulo do erro versus h para a variável T _{max} com interpolação de segunda
ordem com $p_L 2$ e $dp_L = 2$, Fig. 5.13
Tabela B.33 – Módulo do erro versus h para a variável x _{max} com interpolação de segunda
ordem com $p_L 2$ e $dp_L = 2$, Fig. 5.14

Tabela B.41 – Ordem aparente (p _E) de T _{max} versus tamanho da malha (h) e número de
extrapolações (m) Fig. 5.15 (a)
Tabela B.42 – Ordem aparente (p _E) de x _{max} versus tamanho da malha (h) e número de
extrapolações (m) Fig. 5.15 (b)
Tabela B.43 – Módulo do erro versus h para a variável T _{max} com interpolação de quarta
ordem com p _L 2 e dp _L =2, Fig. 5.16
Tabela B.44 – Módulo do erro versus h para a variável x _{max} com interpolação de quarta
ordem com $p_L 2$ e $dp_L = 2$, Fig. 5.17
Tabela B.45 – Ordem aparente (p _E) de T _{max} versus tamanho da malha (h) e número de
extrapolações (m) Fig. 5.18 (a)
Tabela B.46 – Ordem aparente (p_E) de x_{max} versus tamanho da malha (h) e número de
extrapolações (m) Fig. 5.18 (b)
Tabela B.47 – Módulo do erro versus h para a variável T _{max} com interpolação de décima
ordem com $p_L 2$ e $dp_L = 2$, Fig. 5.19
Tabela B.48 – Módulo do erro versus h para a variável x _{max} com interpolação de décima
ordem com $p_L 2$ e $dp_L = 2$, Fig. 5.20
Tabela B.49 - Ordem aparente (p _E) de T _{max} versus tamanho da malha (h) e número de
extrapolações (m) Fig. 5.21 (a)
Tabela B.50 – Ordem aparente (p _E) de x _{max} versus tamanho da malha (h) e número de
extrapolações (m) Fig. 5.21 (b)

LISTA DE ABREVIATURAS E SIGLAS

BEM	Método dos Elementos de Contorno ou Boundary Element Method.
CDS	Central Differencing Scheme.
CHT	Computational Heat Transfer.
CFD	Computational Fluid Dynamics.
CPU	Unidade Central de Processamento ou Central Processing Unit.
DDS	Downstream Differencing Scheme.
EDO	Equação Diferencial Ordinária.
RE	Extrapolação de Richardson ou Richardson Extrapolation.
FDM	Método das Diferenças Finitas ou Finite Difference Method.
FEM	Método dos Elementos Finitos ou Finite Element Method.
FVM	Método dos Volumes Finitos ou Finite Volume Method.
MER	Múltiplas Extrapolações de Richardson ou Repeated Richardson Extrapolation
	(RRE).
TDMA	Tri-Diagonal Matrix Algorithm.
UDS	Upwind Differencing Scheme.

LISTA DE SÍMBOLOS

Romanos

a ponto qualquer do domínio em x.

A conjunto de números, matriz dos coeficientes.

B conjunto de números, matriz dos termos fontes.

 b, b_0, b_1, b_2 ponto qualquer na definido pela associação f (a), coeficientes da

equação quadrática.

 c_n coeficientes do erro, n assume valores 1,2,3,4,5...

E erro da solução numérica.

e, E expoente.

f função.

f derivada primeira da função f.

 f^{ii} derivada segunda da função f.

g malha, função.

h distância entre dois nós consecutivos (m).

k condutividade térmica, W/m.K

L comprimento do domínio na coordenada x (m). Média da norma.

M malha.

M^C máximo da função discreta.
 M^D máximo da função contínua.

m número de extrapolações de Richardson.

N número de nós. \mathbb{R} números reais.

 \mathbb{R}_{+}^{*} números reais positivos e diferente de zero.

 \mathbb{R}^n números reais no espaço *n*-dimensional.

raio de convergência da série, razão de refino.

 $p_{\rm E}$ ordem efetiva.

 p_m ordens verdadeiras, m assume valores 0,1,2,3,4...

 p_L ordem assintótica.

 p_n ordem do erro, n assume valores 2,3,4,5...

P ponto qualquer no espaço.

Pn polinômio de Taylor.

 Q_e , Q_n taxa de transferência de calor (W).

T temperatura (K).

Tc temperatura no centro do domínio.

 T_{max} temperatura máxima no intervalo [0,L] (K).

 T_2 temperatura em $x = \frac{1}{2}$ e $y = \frac{3}{4}$ (K).

Tm temperatura média.

 U_{RI} estimador de Richardson.

x direção coordenada x (m).

y direção coordenada y (m).

z elemento, direção coordenada z (m).

w função de interpolação.

Símbolos

O Ômicron. Relação binária de equivalência.

 \rightarrow tende.

Gregos

ε erro de truncamento.

ε D erro de posição.

 ϕ solução numérica de qualquer variável de interesse.

Φ solução analítica exata para qualquer variável de interesse.

 ∞ infinito.

 λ matriz de soluções.

 $\lambda^i_{\,j}$ solução numérica da variável dependente, $\,i\,$ é a ordem da derivada, j

assume valores ..., j-2, j-1, j, j+1, j+2,...

Índices

n iteração, número de mínimos ou máximos relativos, 1,2,3,.., n.

 π arredondamento.

prog programação.

h discretização.

min mínimo.

máx máximo.

1 INTRODUÇÃO

A condução de calor é um fenômeno natural que ocorre no meio ambiente e nos organismos vivos o que está constantemente sendo investigado por cientistas e engenheiros para novas descobertas. A condução de calor é um modo de transferência de calor em que a troca de energia tem lugar da região de alta temperatura para a de baixa temperatura pelo movimento cinético ou pelo impacto direto de moléculas, no caso de fluidos em repouso, e pelo movimento de elétrons, no caso dos metais. (HOLMAN, 1998). A aplicação da condução de calor é extremamente importante no projeto e desenvolvimento de aeronaves, foguetes, turbinas, motores para automóveis, entre outros.

O desenvolvimento de teorias de condução de calor nasce da necessidade de resolver problemas físicos reais. Em princípio, as teorias já existem e são consagradas no meio científico e podem ser utilizadas para resolver a maioria dos problemas práticos existentes no nosso cotidiano.

Entretanto, há necessidade de desenvolver ferramentas capazes de utilizar as teorias da física com ferramentas matemáticas e numéricas capazes de promover respostas mais acuradas com menor custo e em menor tempo.

Existem, basicamente, três técnicas aplicadas na resolução de problemas em engenharia sendo eles o método experimental, o método analítico e numérico (TANNEHILL *et al.*, 1997). Marchi (2001) apresenta um fluxograma que exemplifica as formas de solução de problemas e seus métodos. Este fluxograma está detalhado na Fig. 1.1 e mostra que todas as formas de solução de problemas iniciam-se com o fenômeno físico tal como se observa na natureza. Após a definição do problema, opta-se por resolvê-lo de forma experimental ou de forma teórica dependendo dos recursos financeiros, equipamentos e mão de obra disponível.

Neste trabalho, não há interesse em definir quais as melhores formas de solução de problemas. O interesse é citá-los, organizá-los de maneira objetiva para situar as múltiplas extrapolações de Richardson (MER) neste universo.

Quando o problema é resolvido da forma experimental, desenvolvem-se modelos em tamanho real ou reduzido ou ampliado, a fim de que seja possível instalar a instrumentação para definir os parâmetros necessários, com vistas ao entendimento e à geração da solução do problema e assim criar os parâmetros de projeto. Os resultados experimentais podem auxiliar os métodos teóricos na geração de constantes e promover a verificação e validação do

equacionamento. É por esse motivo que na Fig. 1.1 existe uma seta de ligação entre os resultados experimentais e a seta que liga o fenômeno, que se quer estudar, aos métodos teóricos.

Quando resolvido da forma teórica, é utilizado o equacionamento matemático por meio de conjunto de funções para associar os dados de entrada às soluções. Em algumas situações, os métodos analíticos possuem ferramentas para resolver esses conjuntos de funções geradas com o equacionamento matemático. Em outras situações, onde as dificuldades matemáticas aumentam com o número de equações e incógnitas, a saída mais adequada são as soluções aproximadas por meio dos métodos numéricos.

Na Fig. 1.1 é demonstrado que os modelos matemáticos podem ser resolvidos por métodos analíticos e numéricos. Para um mesmo problema e mesmo modelo matemático, as soluções obtidas por solução analítica são mais precisas que as resolvidas pelos métodos numéricos. A diferença está no fato de haver erros da solução numérica em virtude do truncamento dos termos da equação diferencial ao aproximá-la por série de Taylor chamado de erro de truncamento, do erro de iteração causado pelo processo iterativo das soluções numéricas, do erro de arredondamento devido à representação finita dos números e o erro de programação gerado pela falha no desenvolvimento do programa computacional. Para corrigir estes erros dos programas numéricos utilizam-se as soluções analíticas ou soluções chamadas "Benchmark" como parâmetro para efetivar as correções nos programas numéricos, caso houver. A busca por erros em um programa não é uma tarefa simples. Quando o erro da solução numérica é composto somente por erros de truncamento este passa a se chamar erro de discretização.

Um engenheiro ou um cientista possui diversas formas de solucionar seus problemas de forma numérica utilizando:

- Softwares comerciais: são programas prontos como PHOENICS®, FLUENT®, FLOW3D® e STAR-CD® que possibilitam solucionar os problemas desejados. Neste caso, os softwares são comerciais e possuem pouca flexibilidade quanto à melhora da precisão numérica e o tempo de processamento. A estimativa de erro normalmente não é informada. As exigências de memória e de armazenamento são impostas sendo imprescindível a configuração mínima de hardware.
- Softwares matemáticos: são programas prontos como MATLAB® e MAPLE® sendo ferramentas matemáticas que possuem rotinas estabelecidas e auxiliam no cálculo

numérico. Embora não dedicadas exclusivamente a problemas de condução de calor, essas ferramentas possibilitam, em alguns casos, solucioná-los. São flexíveis, entretanto, demandam tempo para inserção dos dados e devida execução.

Códigos particulares e não comerciais: são programas gerados em FORTRAN®, C® ou C++ ® ou outro *software* de desenvolvimento. Os programas são exclusivamente para um determinado problema desejado. É possível torná-los flexíveis, rápidos e precisos exigindo baixa capacidade de memória e armazenamento com base no método numérico empregado para o desenvolvimento.

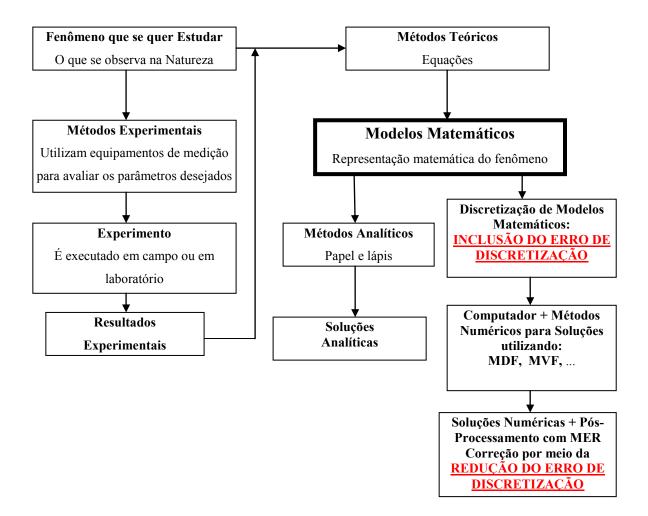


Figura 1. 1 - Métodos de Solução de Problemas (Adaptado de Marchi, 2001).

Os métodos numéricos são utilizados para problemas independentemente da complexidade, da geometria e dos parâmetros físicos. Não há restrição de linearidade e pode

ser considerada a evolução temporal durante o processo. As principais desvantagens dos métodos numéricos estão na determinação do erro computacional, na prescrição apropriada das condições de contornos e nos custos computacionais (TANNEHILL *et al.*,1997). São empregados diversos métodos como:

- Método diferenças finitas ou Finite Difference Method (FDM).
- Método elementos finitos ou Finite Element Method (FEM).
- Método volumes finitos ou Finite Volume Method (FVM).
- Método elementos de contorno ou Boundary Element Method (BEM).

As respostas obtidas pelos métodos analíticos são adequadas quando os problemas possuem geometrias simples. As idealizações em problemas complexos tornam os resultados analíticos distantes dos reais, em virtude da dificuldade de o analista resolver as equações sem fazer considerações simplificadoras do modelo estudado. Uma alternativa para problemas complexos é utilizar as técnicas numéricas. Essas técnicas têm avançado bastante com o avanço da tecnologia cibernética. Os custos computacionais têm diminuído em função dos avanços das tecnologias *de hardware* e *software* (DEITEL *et al.*, 2001)

Às crescentes inovações tecnológicas de *hardware* alavanca também o avanço dos *softwares*, pois *hardwares* necessitam de novos programas que são gerados por métodos e algoritmos computacionais melhorados em termos de utilização de memória e precisão e velocidade de execução possibilitando que o usuário resolva seus de forma mais eficiente. Acompanhando esta tendência dos avanços tecnológicos, a transferência de calor computacional também é beneficiada devido à acessibilidade destas tecnologias para desenvolvimentos de novas técnicas computacionais que possibilitam resolver problemas nesta área mais rapidamente, com menor esforço computacional e com maior precisão.

Os métodos numéricos são capazes de resolver problemas complexos nas áreas das engenharias mecânica, elétrica, naval, eletrônica, espacial e outras. São aplicados para promover soluções aproximadas dos equacionamentos matemáticos. Para este assunto, há uma vasta literatura sobre as técnicas numéricas como por exemplos os livros de Grégoire (2007), Collins (2003), Chapra e Canale (1998) e Isaacson e Keller (1994) que discutem métodos capazes de resolver numericamente problemas em engenharia. A escolha do método numérico a utilizar deve levar em consideração a exigência do problema estudado. Segundo

Broadie e Detemple (1996), a escolha do método numérico deve partir do equilíbrio das seguintes questões:

- Precisão numérica.
- Tempo necessário para a execução.
- Estimativa de erro.
- Flexibilidade.
- Exigências de memória e de armazenamento.

No entanto, mesmos com os avanços tecnológicos em *hardwares* e dos métodos numéricos, para equilibrar todos os quesitos citados por Broadie e Detemple (1996), há a necessidade de simplificações em muitos modelos matemáticos de problemas físicos da atualidade. O motivo pelo qual isto ocorre é de haver limitações de *hardware* em termos de memória, armazenamento de dados, velocidade de processamento e dos métodos numéricos relativos principalmente à acurácia e ao erro do método. Além disso, o custo computacional está diretamente relacionado com o aumento de precisão dos cálculos em função da tecnologia atual de *hardwares*. Consequentemente, para a obtenção de respostas mais acuradas é necessário executar mais cálculos em computadores e, com isso, gastar mais tempo de processamento para alcançar os objetivos desejados.

Novos algoritmos e métodos numéricos mais avançados estão em constante desenvolvimento (MELO Jr., 2005), visando melhorar a precisão numérica e a redução do tempo de processamento. Os métodos numéricos em engenharia, mais especificamente em engenharia mecânica nas áreas numéricas, têm uma grande importância devido estas ferramentas serem a única forma de resolver as equações que regem muitos dos fenômenos físicos estudados.

1.1. O PROBLEMA

Esta pesquisa concentra-se no estudo da técnica de múltiplas extrapolações de Richardson (MER) utilizada para reduzir e estimar o erro de discretização. Verifica-se que a aplicação de MER para reduzir o erro de discretização das variáveis de interesse tem demonstrado ineficiência em problemas em que há máximos e/ou mínimos e não tem produzido resultados mais acurados como esperado. A aplicabilidade do método exige soluções suaves. Para soluções com descontinuidades ou singularidades a eficácia do método é reduzida (ROY, 2005). Alterações oscilatórias na solução não podem ser detectadas (FRANKE *et al.*, 2007).

Para compreender, aperfeiçoar e generalizar o uso de MER nestas situações e torná-lo eficiente na redução do erro de discretização foram utilizados dois modelos que são representados pelas seguintes equações:

- Equação de Laplace bidimensional.
- Equação de Poisson unidimensional.

Estas equações diferenciais parciais de segunda ordem modelam nesta tese a condução de calor e geram, após discretizadas, em seu domínio de cálculo, resultados das variáveis de interesse com características diferentes entre si. Por meio destes resultados será possível compreender os efeitos de MER e assim sistematizar o seu aperfeiçoamento e sua generalização para este tipo de problema.

1.2. DEFINIÇÕES

1.2.1 Erros Numéricos

Para resolver um problema computacionalmente, o domínio matemático, antes contínuo, é transformado em um domínio discreto no mesmo intervalo. O conjunto de pontos do domínio discreto é chamado de malha, e seus pontos, de nós. A distância entre dois nós consecutivos é definida como tamanho da malha h, e o seu refino significa que o número de nós aumenta e o tamanho h diminui. Quando h é constante em todo o domínio, chama-se de malhas uniformes; quando tal não ocorre, têm-se as chamadas malhas não-uniformes.

As diferenciais das equações dos modelos matemáticos são aproximadas por série de Taylor originando uma equação discretizada. Os números de pontos da malha considerados nestas aproximações por série de Taylor definem o esquema de aproximação podendo este ser de primeira, segunda, terceira ou de mais alta ordem. Espera-se que quanto maior a ordem do esquema de aproximação, maior será o custo computacional e mais acurado será o resultado.

As equações discretizadas são aproximações das equações diferenciais do modelo matemático. À medida que o tamanho da malha é reduzido, o erro de discretização diminui. As equações discretizadas com as devidas condições de contorno formam um sistema de equações diferenciais que são resolvidas por métodos iterativos.

Portanto, pode-se afirmar que é intrínseco dos métodos numéricos gerarem respostas aproximadas dos problemas resolvidos devido à introdução de erros motivados pela sua concepção básica. A diferença entre a solução analítica exata (Φ) e a sua solução numérica (ϕ) é definido como erro numérico E(ϕ) (FERZIGER e PERIC, 1999) definido por

$$E(\phi) = \Phi - \phi . \tag{1.1}$$

Os principais tipos de erros numéricos caracterizados pela Eq. (1.1) são erros de discretização, iteração, arredondamento e programação que podem ser representados por

$$E(\phi) = f(E_h, E_n, E_{\pi}, E_{prog}, \dots).$$
(1. 2)

O erro de discretização (E_h) é devido ao truncamento das derivadas durante a aproximação pela série de Taylor. Este erro é reduzido aumentando a ordem da discretização ou diminuindo o parâmetro h da malha. Isso tem um custo que está relacionado com o aumento do tempo de processamento. Para Oberkampf e Blottner (1997) o erro de discretização é aquele causado pela discretização das equações diferenciais parciais dos modelos físicos. Para Marchi e Silva (1999) o erro de discretização é a diferença entre a solução analítica do modelo matemático e a solução numérica exata das equações discretizadas, sem qualquer outro tipo de erro nesta solução definida por

$$E_h(\phi) = \Phi - \phi_h . \tag{1.3}$$

Quando o erro é causado por apenas erro de discretização tem-se

$$E_h(\phi) = C_0 h^{p_0} + C_1 h^{p_1} + C_2 h^{p_2} + C_3 h^{p_3} \dots,$$
 (1.4)

onde C_0 , C_1 , C_2 , C_3 ... são coeficientes que dependem de Φ e suas derivadas, bem como das variáveis independentes, mas independem de h. Os índices p_0 , p_1 , p_2 , p_3 ,... são as ordens verdadeiras de $E_h(\phi)$ cujo conjunto é representado por p_m (MARCHI *et al.*, 2008). Maiores detalhes sobre as equações do erro serão apresentados nos próximos capítulos.

O erro de iteração (E_n) é a diferença entre a solução exata das equações discretizadas e a solução numérica em uma determinada iteração (FERZIGER e PERIC, 2001) numa mesma malha, sem erros de arredondamento e enganos nestas soluções.

O erro de arredondamento (E_{π}) é o erro que ocorre principalmente devido à representação finita dos números reais. Este erro aumenta com a redução do tamanho da malha (MARCHI, 2001).

O erro de programação ($E_{\rm prog}$) consiste principalmente em erro de lógica. É difícil de diagnosticar, pois contém códigos sintática e semanticamente corretos. Este erro faz com que a resposta numérica do programa se distancie da resposta analítica. É uma falha do programador e pode ser corrigida com o trabalho de análise do código computacional.

1.2.2 A redução do Erro de Discretização com MER

Os métodos numéricos aplicados à transferência de calor computacional foram desenvolvidos predominantemente por cientistas interessados na física do fenômeno. Os erros demonstrados na Eq. (1.2) eram frequentemente identificados como fenômenos particulares. Entretanto, têm o efeito forte de alterar o resultado desejado. Para Stern *et al.* (2001), com a complexidade e a responsabilidade das simulações numéricas em engenharia nas últimas décadas é necessário que as simulações efetuadas possuam credibilidade. Porém, como aumentar a credibilidade das simulações numéricas?

Aumentar a credibilidade da simulação numérica não é uma atividade simples. A finalidade básica dos métodos numéricos é resolver problemas físicos de maneira iterativa. Devido à evolução das tecnologias de *hardwares* e *softwares*, estes métodos em engenharia tornaram-se ferramentas fundamentais no desenvolvimento de projetos e motivaram as implementações de modelos mais completos onde a complexidade dos problemas tornou-se crescente. Além disso, a sociedade pressiona a mudança de filosofia da engenharia em busca da sustentabilidade do planeta. O foco não é mais resolver um problema de engenharia, é executar cálculos confiáveis e acurados, com modelos físicos mais completos que promovam a redução efetiva do consumo de energia, mais específicamente em sistemas mecânicos.

Uma maneira capaz de aumentar a credibilidade das simulações numéricas é reduzindo o erro $E(\phi)$ com a redução exclusiva do erro de discretização $E_h(\phi)$.

Em geral, os erros de discretização não podem ser evitados e sim reduzidos. A forma mais comum de reduzir o erro de discretização é selecionar cuidadosamente os esquemas de discretização que possuem um menor erro conhecido *a priori*. Entretanto, com o aumento da ordem do esquema de discretização o custo computacional também é elevado.

Outra forma de obter a redução do erro de discretização é por meio do método de extrapolação de Richardson. Essa técnica utiliza uma série de soluções numéricas geradas a partir dos métodos *FDM*, *FEM*, *FVM* e *BEM*. As soluções obtidas numericamente sem MER são em diversos pontos do domínio discreto e em diversas malhas. Estas soluções são utilizadas *a posteriori* como dados de entrada para o método da extrapolação. MER é um pós-processamento que deve produzir resultados melhores do que os obtidos sem extrapolação quando estabilizado.

Para Burden e Faires (2003), a extrapolação de Richardson ou Richardson *Extrapolation (RE)* é utilizada para gerar resultados com alta precisão, ainda que com aproximações de baixa ordem. Para Grasselli e Pelinovsky (2008) é um algoritmo capaz de melhorar a precisão numérica por meio do cancelamento do erro de truncamento. Moin (2010) define a extrapolação de Richardson com uma técnica poderosa capaz de obter soluções mais precisas. A eficácia desta técnica pode ser melhorada. Ela deve ser aplicada repetidas vezes, isto é, extrapolação sobre extrapolação. Isto é conhecido como múltiplas extrapolações de Richardson (MER) ou *repeated* Richardson *extrapolation (RRE)*. Alguns trabalhos como de Benjamin e Denny (1979), Schreiber e Keller (1983) e Erturk *et al.*, (2005) obtiveram resultados melhorados ao empregar MER.

1.3. DEFINIÇÃO DOS OBJETIVOS

A Transferência de Calor Computacional *ou Computational Heat Transfer* (CHT) é a análise de sistemas, envolvendo transferência de calor, por meio de simulação em computador (VERSTEEG e MALALASEKRA, 2007). É a área da computação científica que estuda métodos computacionais para simulação de fenômenos de transferência de calor reduzindo o número de experimentos práticos, explorando fenômenos que não podem ser estudados em laboratório de forma prática (FORTUNA, 2000). *CHT* possui importância inquestionável no desenvolvimento industrial e na pesquisa, pela possibilidade de reduzir o tempo e o custo dos projetos.

1.3.1 Objetivo Geral

O objetivo geral desta tese é avaliar, aperfeiçoar e generalizar o uso de múltiplas extrapolações de Richardson (MER), capaz de reduzir e estimar o erro de discretização em condução de calor. Busca-se, com isso, diminuir a memória computacional, tempo da

unidade central de processamento ou *Central Processing Unit (CPU)* necessários para solução de problemas de *CHT* e fazer estimativas do erro numérico.

1.6.1 1.3.2 Objetivos específicos

Os objetivos específicos deste trabalho são:

- Desenvolver a teoria de MER, melhorando o desempenho de MER em variáveis de campo como temperatura média e temperatura no centro do domínio.
- Desenvolver a teoria de MER, melhorando o desempenho de MER em variáveis de campo que têm extremos locais ou globais e suas coordenadas.
- Desenvolver um estimador de erro para soluções obtidas com MER. Apesar de ser uma necessidade estimar o erro, poucos problemas na literatura comprovam seus resultados com estimadores de erros.

1.4. MOTIVAÇÃO

MER é uma grande alternativa para acelerar a convergência da solução numérica para variáveis de interesse. O custo computacional ainda é um fator de avaliação nos processos em *CFD* e consequentemente em *CHT*. MER é uma alternativa para redução de custos computacionais. Além disso, é um método robusto que se traduz numa ótima ferramenta para a redução de erro de discretização.

Para variável local, tipo máximo e mínimo, MER é ineficiente. Segundo Marchi *et al.* (2009), o uso de extrapolações múltiplas Richardson não foi tão eficaz para as variáveis velocidade máxima e velocidade mínima para o problema de fluxo no interior de uma cavidade quadrada.

Como qualquer ferramenta, ela não pode ser utilizada para toda tarefa desejada. Nesta tese, utilizam-se problemas que possuem as características de máximo e/ou mínimo e se desenvolve o equacionamento físico e matemático para MER tornar-se eficiente.

MER é uma ferramenta muito poderosa, capaz de gerar resultados mais acurados. É uma ferramenta apropriada para gerar soluções *benchmarks*.

Justifica-se, portanto, a importância e relevância do trato deste tema, tendo-se em vista que os resultados deste trabalho poderão trazer benefícios à sociedade, à indústria e à ciência.

Com relação à sociedade, diz-se que os resultados do trabalho poderão servir como ferramenta para outras áreas como *Computational Fluid Dynamics (CFD)* e financeira.

Já com relação à indústria, pode-se afirmar que os resultados do trabalho poderão impactar positivamente na relação custo x benefício dos projetos em mecânica dos fluidos e

transferência de calor em decorrência de se obter respostas mais rápidas e mais precisas com menor tempo de processamento.

Por fim, quanto à ciência, diz-se que os resultados deste trabalho acrescentarão ao estágio atual do conhecimento um acréscimo no desenvolvimento dos métodos numéricos aplicados à engenharia.

1.5. LIMITES DA PESQUISA

O presente trabalho está delimitado a área da ciência da Engenharia, mais especificamente no ramo da Engenharia Mecânica. Esta tese aborda questões de transferência de calor computacional. Os conceitos aplicados neste texto estão definidos dentro de fontes de pesquisa que vão desde 1910 até 2010. São abordados os temas principais como funções, dinâmica das soluções, aceleração de convergência e extrapolação de Richardson. O tema é de trato universal visto que os resultados são válidos para todo o universo da ciência.

Este trabalho limita-se no objetivo geral e nos objetivos específicos. Para tanto, aborda as equações de Laplace 2D e Poisson 1D, sendo que os fenômenos físicos traduzidos por estas equações simulam todos os efeitos necessários e suficientes para avaliar, aperfeiçoar e generalizar o uso de múltiplas extrapolações de Richardson.

2 FUNDAMENTOS

As técnicas de aceleração de convergência por meio de extrapolações como o minimum polynomial extrapolation e a extrapolação de Richardson são importantes ferramentas em transferência de calor computacional. Possibilitam reduzir o tempo computacional por meio da convergência rápida ao resultado. Drikakis et al. (1998), Shimano et al. (2006), Marchi et al. (2008) e Bertele et al. (2010) são exemplos de sucesso na utilização de diferentes métodos de aceleração de convergência aplicados nas equações de Navier-Stokes, Laplace 2D e fluido incompressível com transferência de calor.

Este capítulo desenvolve os fundamentos gerais de extrapolação como acelerador de convergência das soluções numéricas (também chamadas soluções numéricas sem extrapolação) e, no contexto, introduzir e definir o conceito específico de extrapolação de Richardson. Outros métodos de extrapolação, não menos interessantes, como E-Algoritmo, e-Algoritmo, G-Transformação, extrapolação rotacional, transformações de Levin, processo de Overholt e outros, não serão tratados nesta revisão, por não serem o foco do trabalho. Assinale-se, porém, que esses assuntos estão disponíveis em Sidi (2003) e Brezinski e Zaglia (1991).

2.1. ASPECTOS PRELIMINARES

A ferramenta básica do cálculo é o conceito de função. Kline (1972) apresenta um histórico do conceito de função, introduzido inicialmente com os trabalhos de Newton¹ de 1665, que usou o termo *fluent* para representar alguma relação entre variáveis. No manuscrito Methodus tangentium inversa seu de functionibus de Leibniz² 1673, é utilizada a palavra function para representar qualquer quantidade variando de ponto para outro da curva. Leibniz, em Mathematische Schriften, Abth. 2 Band I (1858, p. 266-269), apresenta uma coleção de termos técnicos com as palavras ordinata, evolutio, differentiare, parameter, differentiabilis, functio, ordinata e abscissa (apud STRUIK, 1969). Em 1697, trabalhando com funções, Bernoulli³ falou sobre quantidade formada de variáveis e constantes. Ele adaptou a expressão de Leibniz function of x para essa quantidade, em 1698. Em 1714,

¹ Isaac Newton (1643-1727). ² Gottfried Wilhelm Leibniz (1646-1716).

³ John Bernoulli (1667-1748).

Leibniz usou a palavra *function* significando quantidades que dependem de uma variável. A respeito da notação, Bernoulli escreveu X ou ξ para uma função geral de x, embora em 1718 mudasse para ϕ x. A notação f(x) foi introduzida por Euler⁴ em 1734.

2.1.1 Funções de uma Única Variável

Sejam A e B subconjuntos de \mathbb{R} . Uma função $f:A \to B$ é uma regra ou correspondência que a cada elemento de A faz corresponder um único elemento de B. O conjunto A é chamado de domínio e B, de imagem (FLEMMING e GONÇALVES, 2006)

Munem e Foulis (1982) definem a função de uma única variável f com uma regra ou uma correspondência que faz associar um e somente um valor da variável f para cada valor de variável f sendo f a variável independente e tornando-se qualquer valor num certo conjunto de números f denominados **domínio** de f. Para cada valor de f no domínio de f o valor correspondente de f é denotado por f(f) tal que,

$$y = f(x) \quad . \tag{2.1}$$

A variável y é denominada variável dependente, visto que seu valor depende de x. O conjunto de números B assumidos por y, à medida que x varia no domínio, é denominado imagem de f. Uma equação que fornece y em termo de x determina uma função f, e diz-se que a função f é definida pela equação.

Se a função é definida por uma equação, compreende-se que o domínio de f consiste naqueles valores de x para os quais a equação faz correponder um e somente um y. A imagem de f é automaticamente determinada pelos valores de y que correspondem, pela Eq. (2.1), aos valores de x no domínio.

Uma função de uma única variável é um conjunto de pares ordenados definido por (x,y). Se (x,y) é um par ordenado neste conjunto, diz que y corresponde a x pela função.

O gráfico de uma função f é o conjunto de todos os pontos (x,y) no plano xy tal que x pertence ao domínio de f e y à imagem de f.

2.1.2 Funções de Várias Variáveis

Gonçalves e Flemming (2007) definem a função de várias variáveis como sendo um conjunto A de n-uplas ordenadas ($x_1, x_2,..., x_n$) de números reais do espaço n-dimensional

_

⁴ Leonhard Paul Euler (1707-1783).

 $(A \subseteq \mathbb{R}^n)$. Se a cada ponto P do conjunto A se associar um único elemento de $z \in \mathbb{R}$, temos uma função $f: A \subseteq \mathbb{R}^n \to \mathbb{R}$

$$z = f(P) \tag{2.2}$$

2.1.3 Funções Analíticas e Série de Taylor

As funções analíticas são aquelas que podem ser localmente expandidas em séries de potências. O estudo de séries teve início em 1668 com Mercator⁵ que publicou um livro intitulado Logarithmotechnia que detalha a série para o logaritmo pela primeira vez. No mesmo ano Brouncker⁶ mostrou que a área abaixo da hipérbole (x + 1)y = 1, em que x = 0 e y = 1 é dada por uma série infinita $1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \dots$ (MAOR, 2006). Em 1671, Gregory descobriu as séries de expansão, porém a mesma série foi publicada por Taylor em 1715 (TANTON, 2005).

Seja a série definida (RUDIN, 1980)

$$\sum_{n=0}^{\infty} a_n x^n, \tag{2.3}$$

 $a_n \in \mathbb{R}$, que converge para x < R, função $f: (-r, r) \to \mathbb{R}$ definida por

$$f(x) = \sum_{n=0}^{\infty} a_n x^n, \quad x < r,$$
 (2.4)

onde a_n , com $n=1,2,3,\dots$ são os coeficientes da série, r>0 é o raio de convergência da série. Então a Eq. (2.3) converge uniformemente [-r+a,r-a] para qualquer a. A função da Eq. (2.4) é contínua e derivável em (-r,r)

$$f^{i}(x) = \sum_{n=1}^{\infty} n a_n x^{n-1}, \quad -r < x < r.$$
 (2.5)

⁵ Nicholaus Mercator (1620-1687). ⁶ William Brouncker (1620-1684).

⁷ James Gregory (1638-1675).

Definindo \mathbb{C} como o conjunto de x para os quais a série converge tal que $\mathbb{C} \subset \mathbb{R}$, a Eq. (2.5) estabelece que a função analítica da Eq. (2.4) é derivável e sua derivada também é analítica em (-r,r). A conclusão disso é que toda a função analítica f é de classe \mathbb{C}^{∞} e todas as suas derivadas são obtidas derivando sucessivamente termo a termo à série de potências de f. A demonstração das Eqs. (2.3), (2.4) e (2.5) está disponível em Rudin (1980).

As funções analíticas possuem uma única e convergente série de potência que as representa sobre um intervalo de raio positivo. Seja a Eq. (2.6) uma função analítica definida (TÁBOAS, 2008)

$$f(x) = \sum_{n=0}^{\infty} a_n (x - a)^n, \quad a - r < x < a + r;$$
 (2.6)

pode-se obter uma expressão para cada coeficiente de a_n da expressão em termos das derivadas de f no ponto a com o mesmo procedimento aplicado na Eq. (2.5), resultando em

$$f^{i}(x) = \sum_{n=1}^{\infty} n a_{n} (x - a)^{n-1},$$
 (2.7)

 $f^{(k)}(x) = \sum_{n=k}^{\infty} n(n-1) \dots (n-k+1) a_n (x-a)^{n-k},$ (2.8)

para $x \in (a - r, a + r)$ tem-se

$$f(a) = a_0, \ f^i(a) = a_1, ..., f^{(k)}(a) = k! a_k,$$
 (2.9)

se $f \in \mathbb{C}^{\infty}$ em um intervalo (-r+a,r-a), a série é definida como série de Taylor de f em torno de a

$$\sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x-a)^n.$$
 (2. 10)

Portanto, a função f é analítica em um intervalo (a + r, a - r) se f é a soma de sua série de Taylor em torno de a, ou seja,

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x - a)^n, |x - a| < r.$$
 (2.11)

Para escrever a série de Taylor de uma função f em torno de um ponto a é preciso que existam as suas derivadas de todas as ordens em a. A ordem n da série de Taylor de uma função f, em torno de um ponto a, resulta no polinômio de Taylor

$$P_n(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x - a)^n,$$
 (2.12)

considerando que f seja da classe \mathbb{C}^{∞} em(a-r,a+r), sendo r>0 um número fixo; se aproximar f em (a-r,a+r) pelo polinômio de Taylor P_n , o erro absoluto é

$$E_n(x) = \sum_{n=0}^{\infty} \frac{1}{(n+1)!} f^{(n+1)}(\sigma) (x-a)^{n+1},$$
 (2.13)

onde $\sigma = a + \xi(x - a)$, ou seja,

$$\sum_{n=0}^{\infty} \frac{1}{(n+1)!} f^{(n+1)}(\sigma)(x-a)^{n+1} = f(x) - E_n(x), \qquad |x-a| < r.$$
 (2.14)

A demonstração da Eq. (2.14) está disponível em Táboas (2008). A série de Taylor é uma expansão para f(x) quando

$$\lim_{x \to 0} \frac{1}{x^n} \left[f(x) - \sum_{n=0}^k a_n x^n \right] = 0.$$
 (2. 15)

A respeito da Eq. (2.15) pode-se observar:

- (i) Para cada n, $\sum_{n=0}^{k} a_n x^n$ torna-se uma melhor aproximação quando $x \to 0$. Quando $x \to 0$, $\sum_{n=0}^{k} a_n x^n$, aproxima-se de f(x) mais rápido do que $x^n \to 0$.
- (ii) A definição não diz nada a respeito do que acontece quando $n \to \infty$. Não é garantido que, para cada x fixado, $\sum_{n=0}^{k} a_n x^n$ tende para f(x) quando $n \to \infty$.
- (iii) Toda função derivável infinitesimal possui uma assintótica expansão, embora essa expansão possa convergir ou não.

A respeito da série de Taylor pode-se afirmar:

Condição 1: se o raio de convergência r > 0 e $\lim_{n \to \infty} E_n(x) = 0$ para todo -r < x < r: a série de Taylor converge para f(x) para todo $x \in (-r, r)$. Portanto, a f(x) é uma função analítica sobre (-r, r).

Condição 2: se o raio de convergência r > 0: a série de Taylor converge para algum outro f(x).

2.2. DINÂMICA DAS SOLUÇÕES E SUAS CARACTERÍSTICAS

As funções expressam as características essenciais dos sistemas físicos em termos de constantes e variáveis que os descrevem (REDDY, 2006). O conjunto dessas equações modela o problema físico. A resolução dessas equações pode ser feita de várias formas, que dependem da complexidade do problema e da afinidade do analista quanto ao uso das ferramentas matemáticas disponíveis no universo do conhecimento. Neste trabalho, utiliza-se métodos numéricos para a resolução dos sistemas de equações, com enfoque em extrapolação de Richardson.

Seja a Eq. (2.16) a representação matricial do sistema de equações do modelo matemático que define o problema:

$$[A][\lambda] = [B], \tag{2.16}$$

onde as matrizes A, λ e B representam o sistema original de m equações em n incógnitas (ANTON E RORRES, 2001), sendo A a matriz de coeficientes, λ a matriz de soluções e B a

matriz dos termos fontes. As soluções numéricas a serem obtidas com base na Eq. (2.16), em mecânica dos fluidos, transferência de calor e outros processos associados, começam quando as leis governantes desses processos estão definidas na forma matemática, em um determinado sistema de coordenadas, geralmente em termos de equações diferenciais (PATANKAR, 1980).

As equações diferenciais são exclusivamente aproximadas por métodos numéricos que podem ser diferenças finitas, elementos finitos ou volumes finitos e discretizados ao longo de um domínio discreto chamado de malha; representam um sistema de equações lineares aproximadas (DICKSTEIN, 1995)

$$[A_{h_g}][\lambda_{h_g}] = [B_{h_g}],$$
 (2.17)

onde h > 0 e $h \in \mathbb{R}_+^*$, sendo h um parâmetro de tamanho de malha. A_{h_g} é a transformação linear onde $g = 0, 1, 2, 3,..., \infty$ representando as diferentes malhas. λ_{h_g} são as soluções numéricas definidas matematicamente por

$$\lambda_{h_g} = \{\lambda_{h_0}, \lambda_{h_1}, \lambda_{h_2}, \lambda_{h_3}, \dots, \lambda_{h_{\infty}}\}$$
 para $h_0 > h_1 > h_2 > \dots > h_{\infty}$, (2.18)

e B_{h_g} é a matriz dos termos fontes. A razão de refino, para malhas uniformes, é a relação entre a malha mais grossa h_g e fina h_{g+1} , ou seja,

$$r = \frac{h_g}{h_{g+1}} \ . {(2.19)}$$

Quando os termos do sistema formado pela Eq. (2.17) são totalmente independentes, a solução é da forma direta e, por exemplo, pode ser resolvida pelo método TDMA (*tri-diagonal matrix algorithm*). Quando A é função da própria variável, a solução é iterativa e, portanto, contém erros numéricos que estão relacionados principalmente pela discretização das equações. Esse tipo de erro é reduzido lentamente, porém proporcionalmente, com a diminuição do tamanho da malha até atingir o erro de máquina.

Nas Figs. 2.1, 2.2 e 2.3 a seguir são representados o domínio unidimensional $L_x = 1$ com diferentes tamanhos de malhas h definidas por h_0 , h_1 e h_2 .

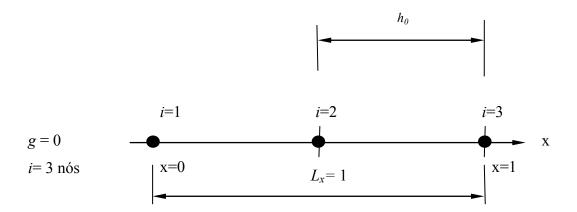


Figura 2.1 - Malha unidimensional com 3 nós e comprimento $L_x = 1$.

Para a Fig. 2.1, o domínio L_x possui uma malha com três nós i e o tamanho h da malha igual a $\frac{L_x}{2}$ ou $\frac{1}{2}$.

A representação x=0 indica o índice de λ_{h_g} que define a malha de estudo. λ_{h_g} é o conjunto formado pelas soluções numéricas em cada nó:

$$\lambda_{h_0} = \{ \phi_1, \ \phi_2, \ \phi_3 \}, \tag{2.20}$$

onde ϕ_i são as soluções numéricas em cada nó i com i= 1, 2 e 3.

Para a Fig. 2.2, o domínio L_x possui uma malha de cinco nós i e o tamanho h da malha igual a $\frac{L_x}{4}$ ou $\frac{1}{4}$:

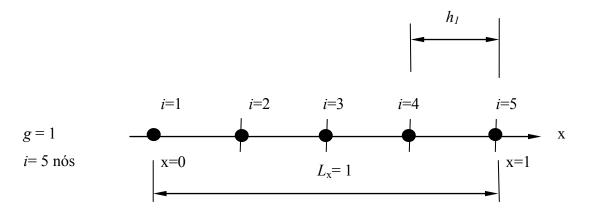


Figura 2.2 - Malha unidimensional com 5 nós e comprimento $L_x = 1$.

A Fig. 2.2 representa a segunda malha de estudo com g = 1. O conjunto formado pelas soluções numéricas em cada nó é dado por

$$\lambda_{h_1} = \{ \phi_1, \ \phi_2, \ \phi_3, \ \phi_4, \phi_5 \},$$
 (2.21)

onde ϕ_i são as soluções numéricas em cada nó i com i= 1, 2, 3, 4 e 5. A razão de refino para os exemplos das Fig. 2.1 e 2.2 é calculada a partir da Eq. (2.19) resultando em

$$r = \frac{h_0}{h_1} = 2 \quad . {(2.22)}$$

Para a Fig. 2.3, o domínio L_x possui nove nós i com tamanho h da malha igual a $\frac{L_x}{8}$ ou $\frac{1}{8}$:

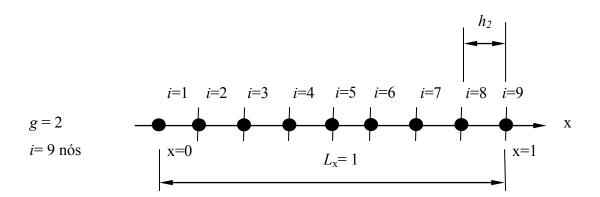


Figura 2.3 - Malha unidimensional com 9 nós e comprimento $L_x = 1$.

A Fig. 2.3 representa a terceira malha de estudo com g = 2 formado pelo conjunto de soluções numéricas em cada nó definido por

$$\lambda_{h_2} = \{ \phi_1, \phi_2, \phi_3, \phi_4, \phi_5, \phi_6, \phi_7, \phi_8, \phi_9 \},$$
 (2. 23)

onde ϕ_i são as soluções numéricas em cada nó i com i= 1, 2, 3, 4, 5, 6, 7, 8 e 9. A razão de refino para os exemplos das Fig. 2.2 e 2.3 é calculada por

$$r = \frac{h_1}{h_2} = 2. {(2.24)}$$

As Eqs. (2.20), (2.21) e (2.23) podem ser representadas, de forma geral, para o caso unidimensional

$$\lambda_{h_g} = \{ \phi_i \} \quad i = 1, 2, 3, \dots$$
 (2. 25)

i independe de g = 1, 2 e 3.

A expansão da Eq. (2.25) para o sistema bidimensional resulta

$$\lambda_{h_g} = \{ \phi_{i,j} \}$$
 $i=1,2,3,...$ $j=1,2,3,...$ (2. 26) $i \in j$ independe de $g=0,1,2,...$

 $\phi_{i,j}$ representa as soluções numéricas das variáveis de interesse do problema em função de uma determinada malha. É incomum obter as soluções numéricas em todos os nós da malha conforme descrito na Eq. (2.18); busca-se obter a solução para um determinado ponto específico conforme a Eq. (2.26) que traduz a solução em um determinado ponto no espaço desejado.

Um efeito qualitativo que ocorre com o refino da malha é a redução do erro de discretização quando o problema é observado para um ponto específico do domínio. Similar à série de Taylor, as soluções numéricas obtidas para um determinado ponto convergem para o resultado analítico, se houver, com o refino da malha, como demonstrado na Fig. 2.4.

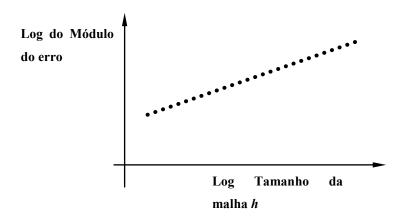


Figura 2.4 - Efeito qualitativo da redução do tamanho da manha h em função do módulo do erro para um determinado ponto no domínio de cálculo.

É importante destacar neste momento que, para obter soluções numéricas em diversas malhas e em um determinado ponto específico do domínio, a razão de refino deve ser precisamente escolhida. Como exemplo, um sistema bidimensional com malha 5x5, formado pelos domínios L_x e L_y , foi representado na Fig. 2.5, onde será avaliada a solução numérica no centro do domínio, isto é, em $\frac{L_x}{2}$ e $\frac{L_y}{2}$:

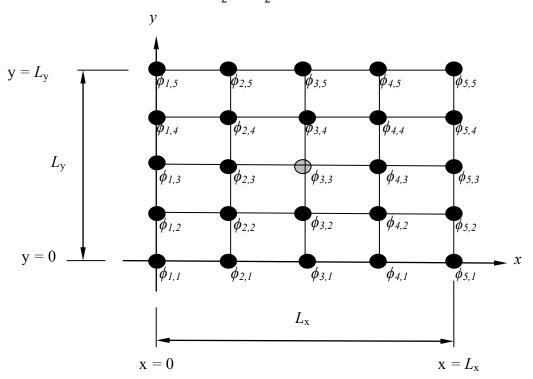


Figura 2.5 - Malha com 25 nós.

A razão de refino deve ser compatível com o ponto de interesse. A escolha da razão errada acarretará na falta de um nó central para avaliação da informação. As malhas 5x5 e 9x9 representadas nas Figs. 2.5 e 2.6 são compatíveis, pois nos dois exemplos existem nós em $\phi_{3,3}$ e $\phi_{5,5}$ que carregam as informações da posição espacial adequada do interesse, isto é, em $\frac{L_x}{2}$ e $\frac{L_y}{2}$. Nesses exemplos, a razão de refino é igual a 2 e verifica-se que haverá, em qualquer malha 5x5, 9x9, 17x17, 33x33, ..., um nó central .

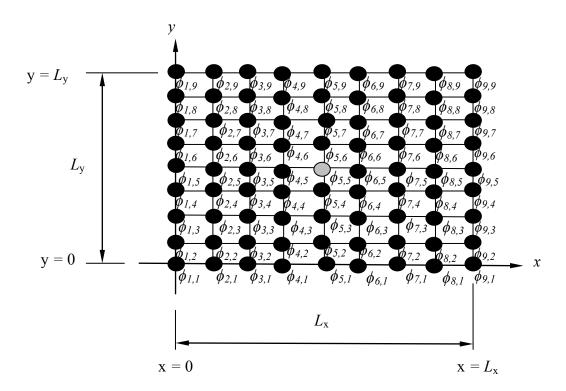


Figura 2.6 - Malha com 81 nós, r = 2.

Em um segundo exemplo, representado na Fig. 2.7, a malha é 6x6, com razão de refino 1,25 em relação à malha 5x5, representada na Fig. 2.5. Neste caso, a taxa de refino não é compatível com o ponto de interesse localizado em $\frac{L_x}{2}$ e $\frac{L_y}{2}$ e não haverá solução numérica no centro do domínio devido à inexistência de um nó central.

Como um dos interesses, nesta tese, é obter as soluções numéricas em uma posição específica do domínio de cálculo e não em toda a malha, define-se como ϕ_g a solução numérica da variável de interesse em um determinado domínio de cálculo, sendo g o índice de malha da solução numérica definida em g=1, 2, 3, 4,..., ∞ . O conjunto das soluções numéricas em uma posição definida no domínio de cálculo é representado por

$$\phi_g = \{ \phi_1, \phi_2, \phi_3, ..., \phi_M \} \quad g = 1, 2, 3, ..., M$$
 (2.27)

onde M é a malha mais fina que se quer alcançar.

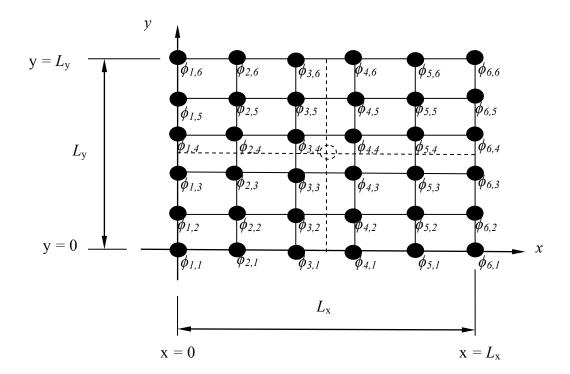


Figura 2.7 - Malha com 36 nós e r = 1,25.

2.3. ACELERAÇÃO DE CONVERGÊNCIA

Na Fig. 2.4, observa-se que a taxa de convergência é constante. Existem métodos que podem fazer acelerar a taxa de convergência. É exemplo disto os métodos de extrapolação, entre eles a extrapolação de Richardson.

A extrapolação é um processo de fazer uma estimativa de um valor f(x) que está fora do intervalo dos pontos bases conhecidos (CHAPRA e CANALE, 1998). Em matemática, a extrapolação é o processo de construção de novos pontos de dados fora de um conjunto discreto de pontos de dados conhecidos. A extrapolação, neste caso, proporciona bons resultados quando usada para ampliar os dados não muito distantes dos dados conhecidos. Em cálculo numérico, o conjunto discreto é formado pela solução numérica da variável de interesse em diversas malhas com $h_1 > h_2 > h_3 > ... > h_k$ em um ponto específico no domínio. A função de extrapolação tem como objetivo obter um conjunto de resultados melhores do que os obtidos pela solução numérica nas diversas malhas. A qualidade do método de extrapolação é limitada pelas hipóteses sobre a função de extrapolação.

As vantagens do método de extrapolação são:

- (i) Baixa complexidade na implementação computacional, se comparado com outros métodos.
- (ii) Não interfere na programação original, sendo este um pós-processamento.
- (iii) Os requisitos computacionais e financeiros são baixos.

As desvantagens do método de extrapolação são:

- (i) Utiliza apenas dados agregados.
- (ii) Assume que as tendências observadas no conjunto de dados de base irão prever os dados futuros.
- (iii) Deve possuir uma quantidade adequada de dados que permita verificar adequadamente o local de convergência da solução numérica.
- (iv) A aplicação da extrapolação não é direta. É necessário calibrar os dados de entrada.

Matematicamente a extrapolação é definida pela seguinte conotação

$$T \colon \phi_g o \ \phi_{g,m}$$
 ,

onde:

$$g = 1, 2, 3, ..., M$$
 (2.28) $m = 1,$

e ϕ_g é a solução numérica sem extrapolação e $\phi_{g,m}$ é a solução numérica extrapolada. A sequência ϕ_g converge muito lentamente e custosamente para o limite (SIDI, 2003). A nova sequência formada por $\phi_{g,m}$ obtida por transformação, com base em ϕ_g , converge mais rapidamente para ∞ do que a sequência formada por ϕ_g .

2.4. TÓPICOS PARA EXTRAPOLAÇÃO DE RICHARDSON

A extrapolação de Richardson (ER) (RICHARDSON e GAUNT, 1927) é um pósprocessamento e pode ser aplicada a posteriori das soluções obtidas por *FDM*, *FEM* e *FVM*. Do ponto de vista da análise da complexidade de algoritmos é importante concentrar-se nessa observação, pois a taxa de crescimento do tempo de execução como uma função do tamanho de entrada n é praticamente nula. Com a utilização da extrapolação de Richardson, há uma redução do erro de discretização, do custo computacional, quando levada em consideração a mesma acurácia das respostas sem acréscimo de complexibilidade na programação ou custo.

O erro de discretização é função das aproximações do modelo matemático e das aproximações das condições de contorno. O parâmetro de malha h está inserido nas equações discretizadas que são resolvidas por meio de métodos iterativos. Na Fig. 2.8 é mostrado qualitativamente que, quando h tende a zero, a solução numérica tende à solução exata. Entretanto, existe um limite de refinamento de h, seja ele econômico ou de limitação tecnológica de hardware e/ou software. Na mesma figura, observa-se que existe a possibilidade de ajustar uma curva com base nos pontos existentes e obter outros pontos extrapolados fora desse domínio. Esses pontos extrapolados podem tender à solução exata. A extrapolação de Richardson está baseada nesse conceito. É um processo de extrapolação que utiliza os resultados obtidos por meio de FDM, FEM e FVM. Estes pontos são denominados pontos-base existentes para obter soluções mais acuradas.

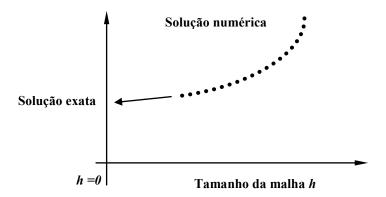


Figura 2.8 - Efeito qualitativo da redução do tamanho da malha h em função da solução numérica para um determinado ponto no domínio de cálculo.

Há necessidade de conhecer os resultados aproximados para o mesmo ponto de interesse, variando-se o parâmetro h e o tipo da função do erro (linear ou quadrática) quando h tendendo a zero. Então, podem-se interpolar os resultados obtidos com a função apropriada e avaliar a interpolação para h=0, isto é, extrapolar o limite para obter uma aproximação de ordem com mais elevada precisão. Portanto, a extrapolação de Richardson é utilizada para gerar resultados com alta acurácia utilizando equações de baixa ordem (BURDEN e FAIRES, 2003).

Uma das bases deste trabalho é a série de Taylor, cuja finalidade é representar a função do erro de discretização. A série de Taylor é uma expansão assintótica que, fazendo a expansão da Eq. (2.12), resulta em

$$P_n(x) = f_{(n)} + f^i(a)(x - a)^1 + \frac{f^{ii}(a)}{2!} (x - a)^2 + \frac{f^{iii}(a)}{3!} (x - a)^3 + \dots + \frac{f^M(a)}{M!} (x - a)^M,$$
(2. 29)

onde $P_n(x)$ representa o valor analítico exato obtido em qualquer coordenada x. A notação de Landau ou assintótica é uma notação matemática utilizada para analisar o comportamento assintótico das funções. O comportamento assintótico é observado na função P(x), quando M tende ao infinito. Para uma dada função P(x), denota-se O(P(x)) tal que,

$$O(P_n(x)) = \begin{cases} f(x) : \text{ existem constantes positivas } c e x_0 \\ tais \ que \\ 0 \le f(x) \le c P(x) \\ para \ todo \\ x \ge x_0 \end{cases}.$$
 (2. 30)

Uma função f(x) pertence ao conjunto O(P(x)) se existe uma constante positiva \mathbf{c} que a limita por c.P(x) para um valor de x grande. Se $P_n(x)$ da Eq. (2.29) é um polinômio de grau M, então $|P_n(x)|$ é $O(x^M)$.

Demonstração:

$$|P_{n}(x)| \leq |f_{(n)}| + |f^{i}(a)|(x-a)^{1} + \left|\frac{f^{ii}(a)}{2!}\right|(x-a)^{2} + \left|\frac{f^{iii}(a)}{3!}\right|(x-a)^{3} + \dots + \left|\frac{f^{M}(a)}{M!}\right|(x-a)^{M}$$

$$(2.31)$$

$$|P_{n}(x)| \leq \left(\frac{|f_{(n)}|}{(x-a)^{M}} + \frac{|f^{i}(a)|}{(x-a)^{M-1}} + \frac{\left|\frac{f^{ii}(a)}{2!}\right|}{(x-a)^{M-2}} + \frac{\left|\frac{f^{iii}(a)}{3!}\right|}{(x-a)^{M-3}} + \dots + \left|\frac{f^{M}(a)}{M!}\right|\right) (x-a)^{M}$$

$$(2.32)$$

$$|P_n(x)| \le \left(\left| f_{(n)} \right| + \left| f^i(a) \right| + \left| \frac{f^{ii}(a)}{2!} \right| + \left| \frac{f^{iii}(a)}{3!} \right| + \dots + \left| \frac{f^M(a)}{M!} \right| \right) (x - a)^M \quad (2.33)$$

é aplicado para $(x - a) \ge 1$. Fazendo

$$c = \left(\left| f^{(n)} \right| + \left| f^{i}(a) \right| + \left| \frac{f^{ii}(a)}{2!} \right| + \left| \frac{f^{iii}(a)}{3!} \right| + \dots + \left| \frac{f^{M}(a)}{M!} \right| \right), \tag{2.34}$$

e $(x_0 - a) = 1$, tem-se

$$|P_n(x)| \le c (x-a)^M,$$
 (2.35)

aplicado para todo $(x - a) \ge (x_0 - a)$.

Portanto, sejam $\phi \subset R^d$, aberto, limitado, e $u, f : \phi \to R^d$ funções reais. Define-se a relação binária de equivalência, indicada por O, no espaço das funções reais F por (PINTO e SOARES, 2006)

$$u \circ f, \forall u, f \in \mathcal{F} \iff \exists C_1, C_2 > 0 : C_1 f(x) \le u(x) \le C_2 f(x), \forall x \in \emptyset.$$
 (2. 36)

Diz-se que as funções u e f são de ordens de grandeza equivalentes. Escreve-se

$$u = 0 (f)$$
 (2.37)

e diz-se que a notação O as seleciona em classes de equivalências de funções.

Demonstração (PINTO e SOARES, 2006):

Assumindo que existe C_1 , $C_2 > 0$,

$$C_1 h^p \le \text{erro}_{\mathbf{n}}(x) \le C_2 h^p, \quad \forall x \in \phi,$$
 (2.38)

escreve-se o erro da Eq. (2.13) da seguinte forma:

$$erro_{n}(x) = O(h^{p}),$$
 (2.39)

onde p é a ordem de convergência, isto é, equivalente a $h \to 0$, e a função do erro convergindo a zero da mesma forma que h^p . A aplicação da notação de Landau na série de Taylor se dá quando p é fixado, uma sequência de **refinamento de** h da discretização do conjunto ϕ_{h1} , ϕ_{h2} , ... com $h \to 0$ é gerada e diz-se que é uma **convergência-h** da solução

aproximada. Quando h é fixado e uma sequência de **refinamentos de** p da discretização ϕ^{p1} , ϕ^{p2} , ... com $p_i \to \infty$ é construída, diz-se consistir de uma **convergência-p** da solução aproximada. Se as variações na discretização são simultaneamente em h e p, diz-se de uma **convergência-** h^p .

A notação de Landau é útil, pois permite comparar aproximações diferentes para um mesmo problema e dá indícios de como a nova aproximação se comporta em relação a problemas conhecidos. A notação *O* (ômicron) é usada para expressar o limite superior do tempo de execução de um algoritmo para resolver um dado problema.

A estratégia de refino segundo Novatny e Fancello (1998) para malhas de elementos finitos são geralmente baseadas na mudança de localização dos nós mediante a redução do parâmetro h, sem alteração dos limites do modelo, aumento da ordem polinomial p dos resultados obtidos ou a combinação dos dois métodos h^p . Os métodos descritos implicam em melhorar localmente o erro de discretização, fornecendo uma solução mais acurada.

No refino *h* da malha, o tamanho da malha reduz cadenciadamente. A ordem polinomial das funções de interpolação dos elementos é constante. Esta modalidade é adequada para os casos em que as soluções possuem um comportamento estável em todo o domínio.

No refino *p* procura-se criar uma sequência de malhas aumentando sucessivamente a ordem polinomial das funções de interpolação em cada elemento. A técnica é adequada para problemas em que a solução analítica não possui pontos de singularidades.

A estratégia h^p é o método mais eficiente no controle do erro. Existem várias formas de desencadear um refino h^p , uma delas é fazer o refino h, identificar as singularidades adaptando-as ao método p.

2.4.1 Aproximações Numéricas e Ordem do Erro a Priori

As aproximações numéricas são um método de resolução das equações diferenciais. Podem basear-se em diversas filosofias, como descrito no capítulo 1. A aproximação numérica será obtida a partir da Série de Taylor definido pela Eq. (2.12). Caracterizando-se adequadamente a série de Taylor, temos:

$$f(x) = f(a) + f^{ii}(a)(x - a)^{1} + \frac{f^{ii}(a)}{2!}(x - a)^{2} + \frac{f^{iii}(a)}{3!}(x - a)^{3} + \dots + \frac{f^{M}(a)}{M!}(x - a)^{M},$$
(2. 40)

onde f é a variável de interesse no modelo matemático e f(x) é o valor analítico exato da variável de interesse, obtido por meio da expansão da série de Taylor a partir do nó a. São conhecidos os valores exatos f(a) e suas respectivas derivadas representadas por f^n , onde n é o índice da derivada. A Eq. (2.40) é válida para uma função contínua em x em um intervalo fechado, bem como suas derivadas. A Fig. 2.9 mostra os respectivos nós da malha para a Eq. (2.40).

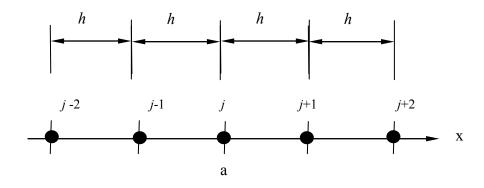


Figura 2.9 - Malha unidimensional com cinco nós.

A aproximação numérica *UDS* por meio da série de Taylor para a derivada primeira utiliza um ponto a montante. Como (x - a) = h e x < a, tem-se

$$f(x)_{j-1} = f(a)_j - f^i(a)_j h + f^{ii}(a)_j \frac{h^2}{2} - f^{iii}(a)_j \frac{h^3}{6} + f^{iv}(a)_j \frac{h^4}{24} - f^v(a)_j \frac{h^5}{120} + \dots$$
(2.41)

Isolando-se a derivada primeira da Eq.(2,41) e dividindo-se por h resulta

$$f'(a)_{j} = \frac{f(a)_{j} - f(x)_{j-1}}{h} + f''(a)_{j} \frac{h}{2} - f'''(a)_{j} \frac{h^{2}}{6} + f''(a)_{j} \frac{h^{3}}{24} - f'(a)_{j} \frac{h^{4}}{120} + \cdots$$

$$(2.42)$$

Como

$$f_{(a)_{j}}^{n} = \left[(\lambda^{n})_{\text{tipo de aproximação}} \right]_{j} + \varepsilon \left[(\lambda^{n})_{\text{tipo de aproximação}} \right]_{j}, \tag{2.43}$$

onde n é a ordem da derivada, λ é a aproximação numérica e ϵ é o erro de truncamento. Para UDS, a Eq. (2.42) é reduzida a

$$\lambda_{j}^{i} = \frac{f(a)_{j} - f(x)_{j-1}}{h},$$
(2.44)

e o seu erro de truncamento

$$\varepsilon(\lambda_{j}^{i}) = f^{ii}(a)_{j} \frac{h}{2} - f^{iii}(a)_{j} \frac{h^{2}}{6} + f^{iv}(a)_{j} \frac{h^{3}}{24} - f^{v}(a)_{j} \frac{h^{4}}{120} \dots$$
 (2. 45)

Portanto, na aproximação *UDS*, o erro de truncamento é de primeira ordem. A generalização da Eq. (2.45) é

$$\varepsilon_{n}(\Phi) = c_{0}h^{p_{L}} + C_{1}h^{p_{1}} + C_{2}h^{p_{2}} + C_{3}h^{p_{3}} + \cdots$$
(2. 46)

O conjunto formado por $p_L = p_0, p_1, p_2, p_3, ...$ é denominado por ordens verdadeiras p_m que são as respectivas ordens M definidas na Eq. (2.29) dos termos da série de Taylor. Em especial p_L recebe o nome de ordem assintótica, e os coeficientes são representados por $c_0 = f^{ii}(a)_j \frac{1}{2}, c_1 = -f^{iii}(a)_j \frac{1}{6}, c_2 = f^{iv}(a)_j \frac{1}{24}, c_3 = -f^v(a)_j \frac{1}{120}, ...$ para o UDS.

2.4.2 Erro de Discretização e sua Estimativa

Quando o erro de discretização é composto por somente erro de truncamento, tem-se $\varepsilon(\phi) = E(\phi)$ e a Eq. (2.46) torna-se a equação geral do erro de truncamento

$$E_{\rm n}(\phi) = C_0 h^{p_L} + C_1 h^{p_1} + C_2 h^{p_2} + C_3 h^{p_3} + \cdots,$$
 (2.47)

onde C_0 , C_1 , C_2 , C_3 , C_4 , ... são constantes originárias da Série de Taylor que representam soma ou subtração da razão das derivadas f', f'', f''', ... pelos seus respectivos fatoriais das aproximações numéricas, que podem ser iguais a c_0 , c_1 , c_2 , c_3 , c_4 , ... O termo (x - a) da série de Taylor é representado por h.

Marchi (2001) demonstra, para as soluções numéricas UDS, CDS, DDS, DDS-2 e regra do trapézio, as ordens verdadeiras e assintóticas do erro de truncamento, como segue na Tab. 2.1.

Tabela 2.1 – Valores previstos para as ordens verdadeiras e assintóticas (MARCHI, 2001).

Tipo de Variável	Solução Numérica Φ	Tipo de Aproximação Numérica	Ordens Verdadeiras p _m	Ordens Assintóticas PL
Derivada de l ^a ordem da variável dependente	UDS	um ponto a montante	1,2,3,	1
	CDS	diferença central	2,4,6,	2
	DDS	um ponto a jusante	1,2,3,	1
	DDS-2	dois pontos a jusante	2,3,4,	2
Derivada de 2ª ordem da variável dependente	CDS	diferença central	2,4,6,	2
Média da variável dependente	m	regra do trapézio	2,4,6,	2

Pode-se observar na Tab. 2.1 que existem ordens assintóticas $p_L = 1$ e $p_L = 2$; para o erro da derivada de la ordem da variável dependente escreve-se

$$E_{\rm n}(\phi)_{\rm UDS,DDS} = O(h^1),$$
 (2.48)

e

$$E_{\rm n}(\phi)_{\rm CDS,DDS-2} = O(h^2).$$
 (2.49)

A característica desta notação é o tempo de execução e o consumo de memória em função da quantidade de dados de uma maneira mais intuitiva e menos trabalhosa. A função que representa o algoritmo da Eq. (2.48) cresce de forma assintótica mais lentamente que o algoritmo da Eq. (2.49).

A Eq. (1.1) e a Eq. (2.47) podem ser calculadas quando se conhece a solução analítica do modelo matemático. Quando a solução analítica é desconhecida, estima-se o seu valor. A estimativa é chamada de incerteza da solução numérica e é assim definida (MARCHI e SILVA, 2005):

$$U(\phi) = \phi_{\infty} - \phi, \qquad (2.50)$$

onde ϕ_{∞} é a solução estimada de Φ .

2.5. EXTRAPOLAÇÃO DE RICHARDSON

A solução exata é composta pela solução numérica, com base em um tipo da aproximação numérica, e o erro de truncamento, associado ao tipo de aproximação feita, definida pela Eq. (2.51)

$$\Phi = \phi_{UDS, DDS, CDS, ...} + E(\phi)_{UDS, DDS, CDS, ...}$$
 (2.51)

sendo o erro de truncamento dependente da ordem da aproximação

$$E(\phi)_{UDS, DDS, CDS, ...} = O(h^{pL}).$$
 (2.52)

A extrapolação de Richardson é uma ferramenta matemática que possibilita obter uma solução mais próxima da solução exata, demonstrada por

$$\phi_{\infty} \cong Extrapolação de \phi_{UDS, DDS, CDS, ...}$$
 (2.53)

onde ϕ_{∞} é a extrapolação de Richardson ao limite.

A extrapolação de Richardson é também conhecida por extrapolação h^2 . A equação de Richardson (1927) na forma original é representada por

$$\phi_{\infty} = \frac{h_2^2 \,\phi_1 - h_1^2 \,\phi_2}{h_2^2 - h_1^2} + E(\phi_{\infty}), \tag{2.54}$$

onde ϕ_1 e ϕ_2 significam as soluções numéricas nas malhas fina e grossa. Sabendo-se que $r = \frac{h_2}{h_1}$ e simplificando a Eq. (2.51) conforme Roache (1994), tem-se a aproximação:

$$\phi_{\infty} = \phi_1 + \frac{\phi_1 - \phi_2}{r^2 - 1}, \qquad (2.55)$$

e

$$\phi_{\infty} \cong \Phi, \tag{2.56}$$

sendo Φ a solução analítica. Ou ainda,

$$\phi_{\infty} = \Phi - E(\phi_{\infty}), \qquad (2.57)$$

e

$$O^{pL} \in O^{p\phi_{\infty}}. \tag{2.58}$$

Portanto a ordem do erro é caracterizada por $p_L > p\phi_{\infty}$ e a convergência da extrapolação de Richardson é do tipo h^p , isto é, depende do refinamento da malha h e da aceleração p.

Demonstração: Extrapolação de Richardson (CHAPRA e CANALE, 1998)

A incerteza da solução foi representada na Eq. (2.50) e é transformada para malhas $\phi_1 \, e \, \phi_2$, definindo

$$U_{\phi_1} = \phi_{\infty} - \phi_1$$
, (2.59)

$$U_{\phi_2} = \phi_{\infty} - \phi_2. {(2.60)}$$

Considerando que a incerteza U de uma determinada solução numérica ϕ seja dada por

$$U_{\phi} = K_u h^2, \tag{2.61}$$

e que K_u é composto pela aproximação numérica e sua derivada, sendo ele constante e independente em todo o refinamento de h. Fazendo a Eq.(2.61) em duas malhas diferentes e identificadas por fina e grossa, tem-se

$$U_{\phi_1} = K_u h_1^2, (2.62)$$

e

$$U_{\phi_2} = K_u h_2^2. {(2.63)}$$

Fazendo-se a relação entre as Eqs. (2.62) e (2.63), resulta em

$$\frac{U_{\phi_2}}{U_{\phi_1}} \cong \frac{h_2^2}{h_1^2},\tag{2.64}$$

Reorganizando a Eq. (2.64), tem-se

$$U_{\phi_2} \cong U_{\phi_1} \left(\frac{h_2}{h_1}\right)^2$$
. (2. 65)

Reorganizando as Eqs. (2.59) e (2.60)

$$\phi_{\infty} = \phi_1 + U_{\phi_1}, \qquad (2.66)$$

$$\phi_{\infty} = \phi_2 + U_{\phi_2}, \qquad (2.67)$$

e igualando-as, tem-se:

$$\phi_2 + U_{\phi_2} = \phi_1 + U_{\phi_1}. \tag{2.68}$$

Substituindo a Eq.(2.65) na Eq. (2.68)

$$\phi_2 + U_{\phi_1} \left(\frac{h_2}{h_1}\right)^2 = \phi_1 + U_{\phi_1}.$$
 (2.69)

Isolando U_{ϕ_1} , tem-se

$$U_{\phi_1} = \frac{\phi_1 - \phi_2}{(r^2 - 1)},\tag{2.70}$$

e substituindo-o na Eq. (2.66) resulta

$$\phi_{\infty} = \phi_1 + \frac{\phi_1 - \phi_2}{(r^2 - 1)}. \tag{2.71}$$

2.6. GENERALIZAÇÃO DA EXTRAPOLAÇÃO DE RICHARDSON

Roache (1994) generaliza a extrapolação de Richardson para qualquer aproximação numérica substituindo o expoente da razão de refino por pL, resultando **a equação generalizada de Richardson** (RE) definida por

$$\phi_{\infty} = \phi_1 + \frac{\phi_1 - \phi_2}{(r^{pL} - 1)}, \tag{2.72}$$

sendo o estimador de erro de Richardson definido por

$$U_{RI} = \frac{\phi_1 - \phi_2}{(r^{pL} - 1)}. ag{2.73}$$

3 METODOLOGIA PARA APLICAÇÃO DE MÚLTIPLAS EXTRAPOLAÇÕES DE RICHARDSON - MER

São complexos os problemas relacionados à transferência de calor que exigem simulações numéricas. As necessidades dos clientes e os custos envolvidos no desenvolvimento de produtos nesta área também são grandes, bem como as incertezas dos modelos utilizados. Frequentemente tem que se trabalhar com resultados aproximados, obtidos por métodos iterativos ou procedimentos computacionais dependentes de alguns parâmetros como tempo, temperatura, pressão e velocidade. A convergência dos sistemas numéricos é lenta, principalmente quando se trabalha com malhas finas que resultam em milhões de nós. A simplificação dos modelos, a redução dos números de nós das malhas, as simplificações unidimensionais para problemas bi ou tridimensionais podem acarretar um grave entrave para a sua utilização na prática. Por esse motivo, técnicas de aceleração, como a extrapolação de Richardson, têm sido estudadas e aplicadas.

No capítulo 2 foi abordado o conceito de extrapolação de Richardson conforme se apresenta na literatura. No presente capítulo serão apresentadas as equações e fundamentações da metodologia de múltiplas extrapolações de Richardson (MER) da forma generalizada e o desenvolvimento da metodologia que torna aplicável MER em problemas que possuem máximos e mínimos.

A aplicabilidade para MER neste capítulo é dividida em duas partes:

- (i) aplicação direta de MER por meio da Eq. (3.1) e suas correlações equação de Laplace 2D a aplicação desta metodologia está detalhada no capítulo 4;
- (ii) aplicação de MER após a aplicação da metodologia para correção dos dados para MER - equação de Poisson 1D - a aplicação desta metodologia está detalhada no capítulo 5.

Para ambos os casos (i) e (ii) houve sucesso na aplicação de MER por meio da metodologia desenvolvida neste capítulo.

3.1. SOLUÇÕES NUMÉRICAS COM MÚLTIPLAS EXTRAPOLAÇÕES DE RICHARDSON

No Cap. 2 foram definidos os pontos base para a aplicação da extrapolação de Richardson. Estes pontos são definidos como as soluções numéricas sem extrapolação, e sua

obtenção é feita diretamente por meio dos resultados das simulações com diferentes *h*. Na Tab. 3.1 é feita a representação esquemática dessas soluções, onde g é o conjunto das malhas representado em um intervalo discreto de passo 1, definido por [1, *G*]. Essa sequência é crescente como 1, 2, 3,..., G e está representada como a malha mais grossa pelo índice 1 e a malha mais fina pelo índice G.

	Solução Numérica sem
Malhas	Qualquer Extrapolação
g=l	φ ₁
g=2	φ ₂
g=3	ф з
g=	
g=G	ϕ_G

Tabela 3.1 – Índices das soluções numéricas sem extrapolação.

A Eq. (3.1) representa a equação para a extrapolação repetida de Richardson, onde $\phi_{g,m}$ é a solução numérica em uma determinada malha g com m extrapolações:

$$\phi_{g,m} = \phi_{g, m-1} + \frac{\phi_{g, m-1} - \phi_{g-1, m-1}}{r^{p_{m-1}} - 1}.$$
(3.1)

A Eq. (3.1) é válida para g=[2,g] e m=[1, G-1]. O diagrama da Tab. 3.2 exemplifica os índices da Eq. (3.1) e indica seus limites de operação. As setas na Tab. 3.2 representam os resultados das extrapolações; por exemplo, o cálculo extrapolado de $\phi_{2,1}$ é obtido a partir das soluções ϕ_1 e ϕ_2 ; o cálculo extrapolado de $\phi_{3,2}$ é obtido a partir das soluções $\phi_{2,1}$ e $\phi_{3,1}$; e assim o método se repete. As extrapolações são possíveis até m extrapolações, que estão relacionadas com o número de soluções numéricas $\phi_{g,m=0}$. Com MER, a correção do resultado é diagonal com ϕ_1 , $\phi_{2,1}$, $\phi_{3,2}$, $\phi_{4,3}$, ... até $\phi_{G,G-1}$.

A representação m=0 indica a representação de solução numérica sem qualquer extrapolação. As ordens verdadeiras representadas por p_m constituem-se de uma progressão aritmética e dependem da ordem assintótica da equação discretizada e são definidas por

$$p_m = p_L + m (p_1 - p_L) (3.2)$$

sendo a Eq. (3.2) válida para g=[1,G] e m=[0,g-1]

m=0m=2m=G-1m=1m=...Solução Numérica Primeira Segunda Malhas sem Qualquer Extrapolação Extrapolação Extrapolação g=1 $\phi_{(g=1,m=0)}$ g=2 $\phi_{(g=2,m=1)}$ g=3 $\phi_{(g=3,m=1)}$ - $\phi_{(g=3,m=2)}$ $\phi_{(g=3,m=0)}$ g=... g=G $\phi_{(g=G, m=0)}$ $\phi_{(g=G,m=1)}$ $\phi_{(g=G,m=2)}$ $\phi_{(g=G,m=G-1)}$ p_L p_I p_2 p_{G-1}

Tabela 3.2 – Índices das Múltiplas Extrapolações de Richardson.

Na Tab. 3.2 dois mecanismos para melhorar a precisão são aplicados conforme descrito no item 3.3. O primeiro é a redução do tamanho da malha h, do que resulta um maior número de malhas g. O outro mecanismo é extrapolar m vezes. Entretanto, os dois mecanismos estão relacionados devido à dependência de m em função de g. Ao utilizar MER há a possibilidade de controlar a ordem da solução numérica extrapolada. As sequências comumente empregadas para o refino da malha devem ser cuidadosamente escolhidas em razão das oscilações de frequência dos resultados numéricos, as quais podem comprometer o resultado extrapolado.

3.2. ORDEM DO ERRO

Segundo Marchi (2001), a incerteza calculada a partir da Eq. (2.71) somente fornece o valor correto do erro de discretização se

$$C_E h^{p_E} = E(\phi) \,, \tag{3.3}$$

onde C_E é dependente de h e definido por

$$C_{\rm E} = C_0 + C_1 h^{p_1 - p_L} + C_2 h^{p_2 - p_L} + C_3 h^{p_3 - p_L} + \cdots$$
(3.4)

A Eq. (3.3) é válida se o tamanho da malha h é muito pequeno, isto é, quando $h \to 0$. Mesmo que a Eq. (3.4) tenha vários termos, o valor do erro tende ao valor do primeiro termo desta equação, isto é

$$C_{\rm E} = C_0$$
 se, e semonete se, $h \to 0$. (3.5)

A ordem efetiva (p_E) é definida, portanto, como a inclinação local da curva do erro de discretização (E) da solução numérica (ϕ) versus o tamanho da malha (h) dos elementos da malha em um gráfico logaritmo (MARCHI, 2001). Esse cálculo a posteriori das soluções numéricas permite confirmar a ordem assintótica do erro de discretização. À medida que h é reduzido, os valores representados em gráfico logaritmo tendem à ordem assintótica do erro de truncamento.

Demonstração: Ordem efetiva baseada em duas soluções numéricas (MARCHI, 2001)

Seja aplicada a Eq. (3.3) em duas malhas $\phi_1 e \phi_2$ definidas por

$$C_E h_1^{pE} = E(\phi_1) \tag{3.6}$$

e

$$C_E h_2^{pE} = E(\phi_2) \tag{3.7}$$

onde o erro de discretização $E(\phi)$ e o tamanho da malha h são conhecidos nas duas malhas. Resolvendo o sistema de equações formado pelas Eqs. (3.6) e (3.7), obtém-se

$$p_E = \frac{\log\left[\frac{E(\phi_2)}{E(\phi_1)}\right]}{\log(r)} , \qquad (3.8)$$

sendo r a razão de refino. A Eq. (3.8) generalizada para MER resulta em

$$(p_E)_{g,m} = \frac{\log\left[\frac{E(\phi_{g-1,m})}{E(\phi_{g,m})}\right]}{\log(r)} ;$$
(3.9)

a Eq. (3.9) é válida para g = [2,G] e m=[0, g-2]. De acordo com as Eqs. (3.8) e (3.9), é necessário conhecer a solução analítica exata da variável para que seja possível o seu cálculo. Nos casos em que não há esta opção utiliza-se o conceito de ordem aparente.

A ordem aparente (p_U) é definida com a inclinação da curva de incerteza (U) da solução numérica (ϕ) *versus* o tamanho da malha (h) dos elementos da malha em um gráfico logaritmo (MARCHI, 2001). Esse cálculo *a posteriori* das soluções numéricas permite confirmar a ordem assintótica do erro de discretização. À medida que h é reduzido, os valores representados em gráfico logaritmo tendem à ordem assintótica do erro de truncamento.

$$K_u h^{p_u} = U(\phi) , \qquad (3.10)$$

onde $K_{\rm u}$ é dependente de h e definido por

$$K_{\mathbf{u}} = K_0 + K_1 h^{p_1 - p_u} + K_2 h^{p_2 - p_u} + K_3 h^{p_3 - p_u} + \cdots$$
(3.11)

Admite-se que K_u é um coeficiente constante e independente de h, no entanto não o é. Esse fato é equivalente ao discutido no item 4.3.1 sendo que o valor de K_u tende ao valor do primeiro termo quando $h \to 0$. Isso implica que, se h não é suficientemente pequeno, a estimativa do erro U_{RI} da Eq. (2.71) será diferente do erro de discretização.

Demonstração: Ordem aparente baseada em duas soluções numéricas (MARCHI, 2001): Substituindo na Eq. (2.50) a Eq. (3.10) obtém-se

$$K_u h^{p_u} = \phi_{\infty} - \phi, \tag{3.12}$$

e escrevendo a Eq. (3.12) para três malhas ϕ_1 , ϕ_2 e ϕ_3

$$K_u h_1^{p_u} = \phi_{\infty} - \phi_1, \tag{3.13}$$

$$K_u h_2^{p_u} = \phi_{\infty} - \phi_2,$$
 (3. 14)

$$K_u h_3^{p_u} = \phi_{\infty} - \phi_3, {3.15}$$

Resolvendo o sistema de equações formado pelas Eqs. (3.13), (3.14) e (3.15), onde as soluções numéricas das variáveis de interesse ϕ_1 , ϕ_2 e ϕ_3 e o tamanho da malha h_1 , h_2 e h_3 são conhecidos, resulta em

$$p_u = \frac{\log\left[\frac{\phi_2 - \phi_3}{\phi_1 - \phi_2}\right]}{\log(r)} \ . \tag{3.16}$$

A Eq. (3.16) generalizada para MER resulta em

$$(p_u)_{g,m} = \frac{\log\left[\frac{\theta_{g-1,m} - \theta_{g-2,m}}{\theta_{g,m} - \theta_{g-1,m}}\right]}{\log(r)},$$
(3.17)

sendo a Eq. (3.17) válida para g = [3,G], m = [0,Int((g-3)/2)] e a razão de refino constante. A representação **Int** () significa a que o valor a ser considerado é a parte inteira do valor calculado no interior do colchete. À medida que $h \to 0$, os valores de $(p_u)_{g,m}$ devem tender à ordem verdadeira p_m do respectivo nível de extrapolação (m). A Eq. (3.1) admite que os valore de p_m são conhecidos. Para se obter p_m a posteriori, sem interferências, em vez de usar ϕ usa-se a nova variável θ .

$$\theta_{g,m} = \theta_{g,m-1} + \frac{\theta_{g,m-1} - \theta_{g-1,m-1}}{r^{(p_u)_{g,m-1}} - 1}.$$
(3.18)

A Eq. (3.18) é válida para g = [3,G], m = [1,Int.((g-3)/2)]. Para m=0 a Eq. (3.18) não se aplica, pois tem-se $\phi_{g,m=0} = \theta_{g,m=0}$. Para estimar adequadamente p_m é necessário conhecer pelo menos as três primeiras ordens: p_0 , p_1 e p_2 . Caso isto não seja possível, pode-se estimar

as outras ordens a partir de $p_0 e p_1$ devido os valores de p_m serem uma progressão aritmética. Quando os valores de p_m são determinados arbitrariamente, o impacto desta escolha pode afetar negativamente o resultado esperado (MARCHI *et al.*, 2008). Os valores de p_m são obtidos de $(p_u)_{g,m}$. Eles são números inteiros e positivos que são extraídos para cada nível de extrapolação (m). Quando $h \rightarrow 0$ a tendência vale : $p_u \rightarrow p_m$.

3.3. ESTIMADOR DE ERRO DE RICHARDSON

A estimativa do erro de discretização da solução numérica (U_{RI}) da Eq. (2.73) pode ser empregada para qualquer variável de interesse. Para g malhas e m extrapolações de Richardson o estimador de Richardson é dado por

$$U(\phi_{g,m}) = \frac{\phi_{g,m} - \phi_{g-1,m}}{r^{p_m} - 1},$$
(3.19)

onde a equação é válida para g=[2,G] e m=[0, g-2].

3.4. METODOLOGIA PARA CORREÇÃO DOS DADOS PARA MER

Esta parte do presente capítulo se concentra na fundamentação teórica e pretende demonstrar a metodologia utilizada para a correção dos dados para a aplicação de MER em problemas de máximos e mínimos em que estes pontos são desconhecidos ao longo do domínio.

A efetividade da múltipla extrapolação de Richardson está associada diretamente à qualidade dos dados de entrada no método. Partindo-se do princípio que a variável de interesse está localizada em ponto indefinido do domínio, conclui-se que não há uma sequência de nós em malhas diferentes (h₁, h₂, h₃,...) que carregue a informação necessária sem haver outros tipos de erros conforme descrito em 3.4.5.

Chapra e Canale (1998) descrevem que, para a técnica de extrapolação de Richardson funcionar, os dados de entrada devem:

- (i) ser uniformemente espaçados e
- (ii) ser uniformemente gerados em forma sucessiva.

A afirmação (a) está correta e possui relação direta com a razão de refino (r) da malha na qual deve ser constante. Entretanto, quanto à afirmação (b), verifica-se que está

incompleta. Os dados devem ser uniformemente gerados em forma sucessiva e devem ser exclusivamente pontuais com a mesma posição ao longo das diversas malhas a serem extrapoladas.

A dificuldade de problemas com MER que possuem máximos e/ou mínimos está no ponto de inflexão. Os dados de entrada para MER oscilam em torno deste ponto devido a função numérica ser discreta.

3.4.1 Função Contínua e Função Discreta

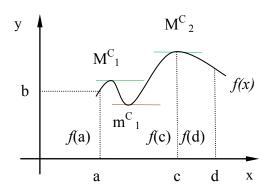


Figura 3.1 – Função contínua e pontos de inflexão.

A função f é contínua em um ponto a qualquer do domínio de x se, e somente se, as seguintes condições forem válidas:

(i)
$$f(a)$$
 é definido, (3.20)

(ii)
$$\lim_{x\to a} f(x)$$
 existe, e (3.21)

(iii)
$$\lim_{x \to a} f(x) = f(a)$$
. (3.22)

Analogamente, uma função é contínua em um intervalo fechado [a,d] se for contínua em todos os números do intervalo (MUNEM e FOULIS, 1982).

A função de A em B é um subconjunto de A X B. As associações funcionais serão relacionadas por uma equação que descreve as associações. Tecnicamente é comum a utilização de parte do domínio de A para representar, em pares ordenados, B. A utilização da função f, que relaciona A em B em apenas alguns pontos do domínio de A, acaba por gerar outra função, denominada g, que não representa fielmente a função f. A função g consiste em pontos discretos e possui o domínio diferente da função f, porém representa f ao longo do intervalo fechado [a,d]. A função g(x) é definida como uma função discreta que representa a função f(x).

A função f(x), representada na Fig. 3.1, possui pontos de picos definidos por M_1^C e M_2^C e depressão definida por m_1^C . A letra M maiúscula significa ponto de máximo, o superíndice C significa que a função é contínua, e os subíndices 1,2,3,... significam o número do pico. A letra m minúscula significa ponto de mínimo. Os super e subíndices possuem o mesmo significado.

Uma determinada função pode conter vários pontos de máximo e mínimo. Por exemplo, a função f(x), definida no intervalo fechado [a,d] na Fig. 3.1, possui dois máximos locais definidos por M^C1 e M^C2 e um mínimo local definido por M^C1 , que demonstram os picos e a depressão dessa função. Entretanto, observa-se na Fig. 3.1 que um pico é maior que o outro, isto é, $M^C1 < M^C2$; neste caso diz-se que M^C2 é o máximo absoluto ou global da função f(x) definida no intervalo [a,d]. Sendo o valor c a coordenada da variável independente x que faz a associação y = f(x) assumir o maior valor y, tal que $M^C = y_{max} = f(c)$, podese definir que

$$M^{c} > f(x), (3.23)$$

onde M^C é o maior valor da função f(x), isto é, o máximo absoluto ou global da função f(x). Analogamente, pode-se definir que

$$m^{\mathcal{C}} < f(x) \,, \tag{3.24}$$

onde m^C é o menor valor da função f(x), isto é, o mínimo absoluto ou global da função f(x). Observa-se que os subíndices 1,2,3... nas Eqs. (3.23) e (3.24) deixaram de existir, caracterizando as letras M e m como máximo ou mínimo absoluto ou global. O superíndice C nas Eqs. (3.23) e (3.24) caracteriza que é o máximo ou mínimo de uma função contínua.

O ponto crítico M^C ou m^C é determinado para uma função f, fazendo

$$\frac{df(x)}{dx} = 0, ag{3.25}$$

podendo assumir a derivada primeira um ponto de máximo ou mínimo no intervalo de interesse. O ponto crítico é máximo ou mínimo quando (JACQUES, 2006):

(i)
$$M^{C}: \frac{f^{2}(c)}{dx} > 0$$
, (3.26)

(ii)
$$m^{C} : \frac{f^{2}(c)}{dx} < 0.$$
 (3.27)

O ponto crítico discreto é máximo ou mínimo quando:

(i)
$$M^D$$
: c é o maior valor da função $g(x)$ no domínio, (3.28)

sendo que os valores anterior e posteriores a \mathbf{c} são menores que \mathbf{c} e $\mathbf{c} \neq \mathbf{a}$ e $\mathbf{c} \neq \mathbf{d}$.

(ii)
$$m^D : c \in o \text{ menor valor da função } g(x) \text{ no domínio,}$$
 (3.29)

sendo que os valores anterior e posteriores a \mathbf{c} são maiores que \mathbf{c} e $\mathbf{c} \neq \mathbf{a}$ e $\mathbf{c} \neq \mathbf{d}$.

3.4.2 Discretização do Domínio de Cálculo

Uma variável é discreta quando possui um número finito de valores dentro de um intervalo fechado (MONTGOMERY e RUNGER, 2003). No item anterior, a função contínua foi definida por f(x) e a função discreta, por g(x).

As técnicas numéricas em *CHT* trabalham com domínio discreto e obtendo os correspondentes pontos na imagem. A associação é feita a partir de uma função g(x) discreta. A função g(x) representa a função f(x) contínua. Neste momento não está sendo considerado nenhum tipo de erro numérico, isto é, g(P) = f(P), onde P é um ponto qualquer do domínio de [0,L]. O processo de dividir o domínio contínuo [0,L] em finitos pontos discretos é chamado de discretização, que consiste em definir em quais pontos se deseja conhecer a variável dependente y, construindo sobre o domínio x uma malha que conterá um conjunto de pontos, denominados nós (SUERO, 2006). O tamanho de cada elemento de malha h para um sistema unidimensional e uniforme é definido por (TANNEHIILL *et al.*, 1997):

$$h = \frac{L}{N-1} , \qquad (3.30)$$

onde h é a distância entre dois nós consecutivos, L é o tamanho do domínio e N é o número de elementos da malha.

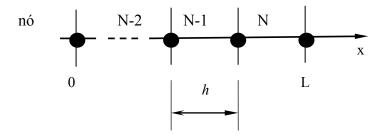


Figura 3.2 - Discretização do domínio [0,L] em N elementos.

O domínio discretizado não é contínuo, conforme é detalhado na Fig. 3.2, e a informação da variável x limita-se aos pontos estabelecidos no domínio. Observa-se que, quanto maior for o número de nós do domínio, maior será a quantidade de informação e maior será o tempo computacional para processar as informações.

3.4.3 Efeito da Discretização no Ponto de Máximo ou Mínimo

O domínio discretizado possui um número finito de nós e as informações das variáveis de interesse estão contidas nestes pontos. A Fig. 3.3 compara a função discreta g(x) e a função contínua f(x). Observa-se que os pontos de máximo da função f(x) e g(x) são diferentes. O domínio da função g(x) não contempla o ponto onde a função f(x) é máxima, recaindo na desigualdade

$$M^{C} \neq M^{D}$$
, (3.31)

onde M^D é o máximo da função discreta g(x).

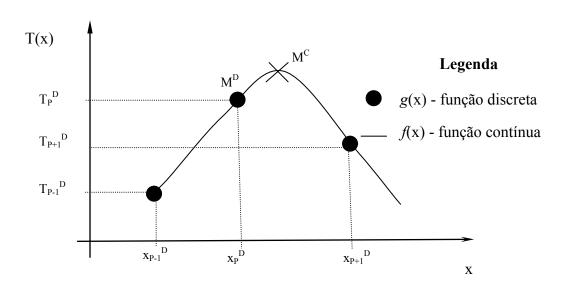


Figura 3.3 - Diferenciação entre a função f(x) e função g(x).

Quando a malha $h \to 0$, o módulo da diferença entre os máximos de f(x) e g(x) se aproxima de zero:

$$\left| \mathbf{M}^{\mathsf{C}} - \mathbf{M}^{\mathsf{D}} \right| \to \mathbf{0} ; \tag{3.32}$$

quando a malha $h \to 0$, o módulo da diferença entre os mínimos de f(x) e g(x) se aproxima de zero:

$$\left| \mathbf{m}^{\mathbf{C}} - \mathbf{m}^{\mathbf{D}} \right| \to \mathbf{0} . \tag{3.33}$$

3.4.4 Metodologia para Correção dos Dados de Entrada para MER

No item 3.4.1, foi definido que a imagem e o domínio da função de associação g(x) estão contidos na imagem e domínio da função de associação f(x). Já no item 3.4.3, foi demonstrado que o ponto de máximo (M^D) de uma função discreta g(x) é diferente do ponto de máximo (M^C) de uma função contínua f(x). A mesma afirmação vale para o ponto de mínimo. O ponto de máximo da função g(x) representada pela Eq. (3.28) que é o maior valor de todos os valores representados pela associação g(x). Foi considerado que havia uma g(x) discreta geradora de pontos nodais com h qualquer sobre a curva contínua de f(x) e a solução discreta era igual à solução analítica nodal nesses pontos, não havendo, portanto, nenhum tipo de erro, inclusive erro de discretização. Neste trabalho, o que se procura minimizar é o erro de discretização; no entanto, ao se utilizar do valor de máximo da função g(x), está incluindo-se na solução numérica um novo tipo de erro, que será definido a seguir.

3.4.5 Erro de Posição da Informação

O erro de posição εD é definido matematicamente por

$$\varepsilon D = M^{C} - M^{D}. \tag{3.34}$$

É a diferença entre o ponto de máximo da função contínua e o ponto de máximo da função discreta. Vale para o ponto de mínimo, no caso deve substituir M por m. Os valores de M é a forma geral de representar todas variáveis de interesse do problema como, por exemplo, temperatura máxima/mínima e posição máxima/mínima. Na Fig. 3.4 é mostrado este tipo de erro, onde $\varepsilon D_{\rm M}$ é o erro de posição na coordenada M e $\varepsilon D_{\rm X}$ é o erro de posição na coordenada x (de forma geral, podendo ser y,z).

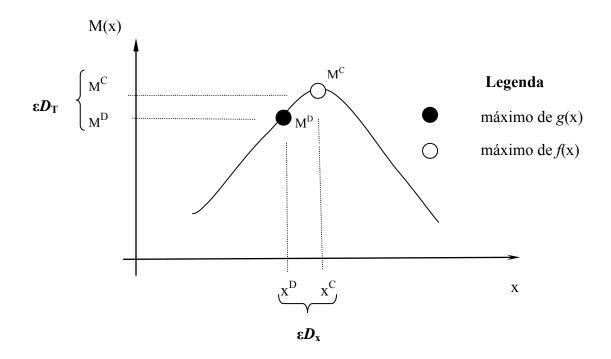


Figura 3.4 - Diferenciação entre a função f(x) e função g(x).

O erro de posição é reduzido quando é aumentado o número de nós da função g(x) ou reduzido o tamanho da malha h, para o mesmo intervalo [0,L]. Na Fig. 3.5 são mostradas, para o mesmo intervalo [0,L], quatro curvas para o mesmo ponto de máximo da função f(x). Sobre a função f(x) representou-se, sem erro de discretização, a função discreta g(x) que, nos pontos discretos, possui o mesmo valor da função f(x). Quando a função g(x) possui menos nós, definiu-se como malha mais grossa; já quando a função g(x) possui mais nós, como malha mais fina. Observa-se que, à medida que a malha se torna mais fina, o ponto de máximo da função g(x) se aproxima do máximo da função f(x). Portanto, conclui-se que o erro de posição tende a zero quando a malha é refinada. O erro de posição é determinado pela Eq. (3.35)

$$\varepsilon D_{\rm G} = M_{\rm G}^{\rm C} - M_{\rm G}^{\rm D} \,, \tag{3.35}$$

onde εD_G é o erro de posição da malha mais grossa, M_G^C é valor do ponto de máximo da variável de interesse na malha grossa da função f(x) e M_G^D é valor do ponto de máximo da variável de interesse na malha grossa da função discreta g(x). Para a malha mais fina tem-se

$$\varepsilon D_{\rm G} = M_{\rm F}^{\rm C} - M_{\rm F}^{\rm D}, \tag{3.36}$$

onde εD_F é o erro de posição da malha mais fina, M_F^C é valor do ponto de máximo da variável de interesse na malha fina da função f(x) e M_F^D é valor do ponto de máximo da variável de interesse na malha fina da função discreta g(x). Comparando-se as Eqs. (3.35) e (3.36) com a Fig. 3.5 resulta em

$$\varepsilon D_{\rm F} < \varepsilon D_{\rm G}.$$
 (3.37)

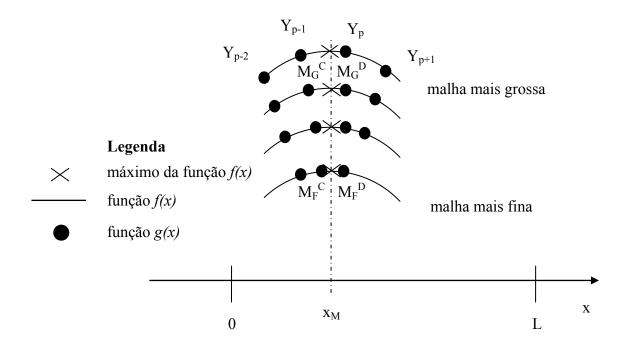
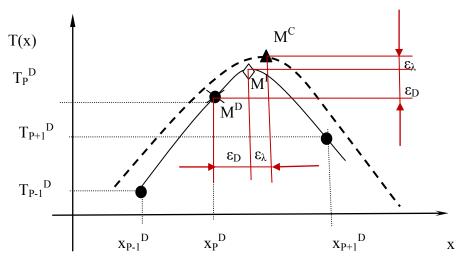


Figura 3.5 - Solução analítica em diversas malhas para geração da função g(x).

Em problemas práticos, normalmente a solução analítica é desconhecida, portanto, a função f(x) é desconhecida e a função g(x) é a solução numérica. Observa-se que a função g(x) não é baseada na função f(x) e sim na solução numérica do problema. Admitindo-se a premissa de que para a função g(x) as fontes principais de erros são (1) erro de posição (εD) e

(2) o erro de discretização ($\epsilon\lambda$). A Fig. 3.6 define esses erros graficamente. A curva tracejada é a função f(x) desconhecida que possui um ponto de máximo local. O triângulo preto, definido por M^C , isto é, o ponto de máximo e suas coordenadas são o objetivo central deste capítulo. Os círculos pretos são a solução numérica conhecida. O losango M^I é o ponto de máximo formado pela curva contínua w(x).



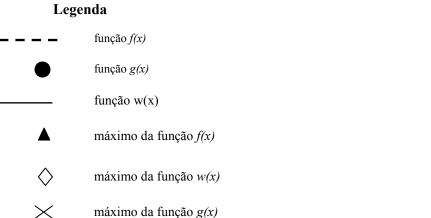


Figura 3.6 - Solução analítica, numérica e ponto crítico.

Sabe-se que MER é efetivo para redução dos erros de discretização, no entanto para problemas de máximo e mínimo demonstra ser ineficiente. A eficiência de MER está relacionada com os tipos de erros envolvidos nas soluções numéricas. MER é eficiente se o erro predominante for o de discretização. Na Fig. 3.6, observa-se que existe um erro de posição relativo ao deslocamento do nó da malha em relação ao ponto de interesse. Portanto,

MER é ineficiente quando há outros erros que contaminam o resultado da solução numérica. Para minimizar o erro de posição, é obtida a curva contínua interpolada w(x) em função dos pontos discretos g(x). O erro numérico é representado conforme a Eq. (1.1).

A função w(x) é uma função inexistente, entretanto pode ser gerada a partir de interpolada polinomial utilizando os dados da função discreta g(x) em torno do ponto M^D da solução numérica. A função w(x) é contínua e derivável; a partir da derivada primeira da função w(x), posiciona-se o ponto de máximo M^I mais próximo do máximo da função f(x), definido por M^C . O resultado da interpolação é a redução do erro de posição.

3.4.6 Minimização do Erro de Posição com Interpolação Polinomial

A interpolação polinomial é a alternativa adotada para minimizar o erro de posição definido na Eq. (3.34). Consiste em determinar o único polinômio de grau n que passa pelos n+1 pontos dados (CHAPRA e CANALE, 1998). Há diversas formas de se obter e expressar esse polinômio. Optou-se por adotar dois métodos, isto é, a interpolação quadrática para polinômio de segundo grau e o polinômio de Newton para grau 4 e grau 10. A equação geral do polinômio para n+1 pontos é

$$w(X) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \dots + a_n x^n$$
 (3.38)

Considera-se a hipótese de que, quanto maior o grau do polinômio, menor é o erro de posição em virtude de a curva do polinômio estar ligada a mais nós.

3.4.6.1 Interpolação Polinomial Quadrática

A Eq. (3.39) é a forma conveniente para o polinômio quadrático, que é definido por (CHAPRA e CANALE, 1998)

$$w(x) = b_{P-1} + b_P(x - x_{P-1}) + b_{P+1}(x - x_{P-1})(x - x_P), \tag{3.39}$$

onde os pares ordenados conhecidos de g(x) são (x_{P-1}, T_{P-1}) , (x_P, T_P) e (x_{P+1}, T_{P+1}) ; a Eq. (3.39) para $x = x_{P-1}$ resulta

$$b_{P-1} = T_{P-1}; (3.40)$$

substituindo na Eq. (3.39) o parâmetro $x = x_P$ e fazendo $b_{P-1} = T_{P-1}$, obtém-se

$$b_P = \frac{T_P - T_{P-1}}{T_P - T_{P-1}}. (3.41)$$

Substituindo-se as Eqs. (3.40) e (3.41) em (3.39),

$$b_{P+1} = \frac{\frac{T_{P+1} - T_P}{x_{P+1} - x_P} - \frac{T_P - T_{P-1}}{x_P - x_{P-1}}}{x_{P+1} - x_{P-1}}.$$
(3.42)

Quando h→ 0 e a precisão dos cálculos computacionais é quádrupla a forma da interpolação apresentada por Chapra e Canale (1998) é a que produz melhores resultados na interpolação. A solução trivial que resolve três equações e três incógnitas produz resultados divergentes do esperado devido haver muitas divisões e multiplicações durante os cálculos dos coeficientes para geração da equação de interpolação.

3.4.6.2 Interpolação Polinomial de Ordem n > 2

A forma geral de Newton para um polinômio w(x) que interpola $x_{P-(n/2)}$ até $x_{P-[(n-2)/2]}$ para os pares ordenados conhecidos de g(x) são $(x_{P-(n/2)}, T_{P-(n/2)}), ..., (x_P, T_P)$ e $(x_{P+[(n+2)/2]}, T_{P+[(n+2)/2]})$ dados por

$$w(x) = b_{P - \left(\frac{n}{2}\right)} + b_{P - \left(\frac{n+2}{2}\right)} \left(x - x_{P - \left(\frac{n}{2}\right)}\right) + \cdots$$

$$+ b_{P + \left(\frac{n-2}{2}\right)} \left(x - x_{P - \left(\frac{n}{2}\right)}\right) \left(x - x_{P - \left(\frac{n+2}{2}\right)}\right) \dots \left(x - x_{P + \left(\frac{n-2}{2}\right)}\right)$$
(3.43)

sendo

$$b_{P-\left(\frac{n}{2}\right)} = T_{P-\left(\frac{n}{2}\right)} \tag{3.44}$$

$$b_{P-\left(\frac{n+2}{2}\right)} = \frac{T_{P-\left(\frac{n}{2}\right)} - T_{P-\left(\frac{n+2}{2}\right)}}{x_{P-\left(\frac{n}{2}\right)} - x_{P-\left(\frac{n+2}{2}\right)}}$$
(3.45)

$$b_{P-\left(\frac{n+4}{2}\right)} = \frac{\frac{T_{P-\left(\frac{n+4}{2}\right)} - T_{P-\left(\frac{n+2}{2}\right)}}{x_{P-\left(\frac{n+4}{2}\right)} - x_{P-\left(\frac{n+2}{2}\right)}} - \frac{T_{P-\left(\frac{n+2}{2}\right)} - T_{P-\left(\frac{n}{2}\right)}}{x_{P-\left(\frac{n+2}{2}\right)} - x_{P-\left(\frac{n}{2}\right)}}$$

$$x_{P-\left(\frac{n+4}{2}\right)} - x_{P-\left(\frac{n}{2}\right)}$$
(3.46)

até

$$b_{P+\left(\frac{n-2}{2}\right)} = f\left[x_{P-\left(\frac{n}{2}\right)}, x_{P-\left(\frac{n+2}{2}\right)}, x_{P-\left(\frac{n+4}{2}\right)}, \dots, x_{P}, \dots, x_{P+\left(\frac{n-4}{2}\right)}, x_{P+\left(\frac{n-2}{2}\right)}\right], \tag{3.47}$$

onde a função com colchetes corresponde às diferenças divididas de Newton.

A escolha pela interpolação polinomial consiste no fato que o polinômio interpolado gera uma função contínua, isto é, em $\mathbb{R} = (-\infty, +\infty)$ (MUNEM e FOULIS, 1982). A função w(x), por ser contínua em todo o domínio de x, é derivável e possibilita obter o resultado de máximo de M^I .

3.5. CONCLUSÃO

A múltipla extrapolação de Richardson é o aperfeiçoamento da técnica da extrapolação de Richardson (RE) em que é aplicada, repetidas vezes, a equação da extrapolação. É uma ferramenta matemática usada para melhorar a precisão dos valores estimados por qualquer método numérico baseado em discretização (CAMARGO, 2007).

MER pode ser aplicada à solução numérica de um sistema de equações diferenciais parcial, mas é vulnerável a certas anomalias que fazem a metodologia falhar e divergir do objetivo citado por Camargo (2007). Segundo Burg e Erwin (2008), essas anomalias incluem o efeito da base de dados pobres causados pelo refinamento da malha, resultando em uma sequência de dados que não são geometricamente semelhantes entre si.

Outra vulnerabilidade está associada às oscilações das soluções numéricas em pontos de descontinuidades em razão das oscilações das frequências das velocidades das ondas na resolução da malha. A localização das descontinuidades, a frequência da obtenção dos dados

deve ser avaliada, pois a extrapolação de Richardson ou as múltiplas extrapolações de Richardson aplicadas diretamente sobre soluções numéricas impróprias não melhoram o resultado e não permitem o sucesso da metodologia.

Portanto, a extrapolação de Richardson e, consequentemente MER, deve ser aplicada cautelosamente para atingir o objetivo desejado.

4 EQUAÇÃO DE LAPLACE BIDIMENSIONAL

Neste capítulo, é testada a metodologia de MER em um problema de condução de calor, modelado pela equação de Laplace bidimensional com condições de contorno de Dirichlet, conforme Fig. 4.1. Será utilizado um domínio Lx=1 e Ly=1, isto é, uma placa plana quadrada de condutividade térmica do material k=1 (W/m.K). As variáveis de interesse são a temperatura T=T(x, y) em uma região do domínio, a média do campo de temperaturas, a taxa de transferência de calor em dois contornos e a norma do erro de discretização.

Resolver a equação de Laplace bidimensional tanto analítica quanto numericamente é um processo trivial. Mas o objetivo é este: utilizar um equacionamento estável e confiável para testar outra ferramenta, mais especificamente MER. Introduziu-se na programação numérica o *multigrid* como meio de acelerar a convergência das soluções numéricas e possibilitar a obtenção de resultados em malhas muito finas, para alcançar o objetivo: testar MER em maior número de malhas e observar e apreender o seu comportamento. As soluções numéricas obtidas com *multigrid* são expressas no texto como soluções numéricas sem extrapolação; as extrapoladas com MER, como soluções numéricas extrapoladas.

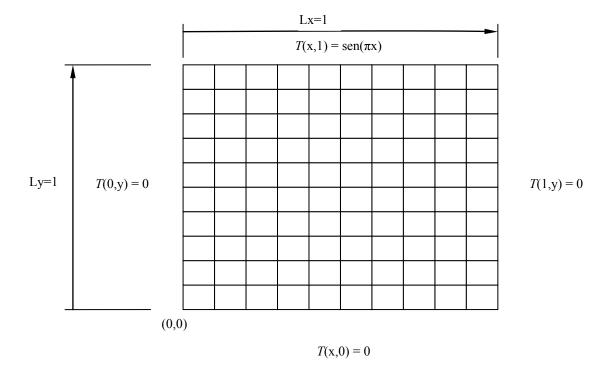


Figura 4.1 - Grade quadrada representando uma placa.

São consideradas propositalmente neste trabalho variáveis primárias e secundárias. ER é sensível à qualidade dos dados dos pontos base; a verificação em diferentes tipos de variáveis possibilita verificar a ordem do erro por meio dos resultados com MER. Além disso, é utilizada precisão dupla e quádrupla nos cálculos com MER para analisar o comportamento delas com a variação de precisão. Serão apresentados resultados mostrando que MER, para a equação de Laplace bidimensional, possui um comportamento estupendo na redução do erro de discretização, alcançando até o nível do erro de máquina. Serão definidos pontos importantes a serem considerados na aplicação da ER e, consequentemente, MER.

4.1. MODELO MATEMÁTICO

Os processos físicos relacionados à difusão referem-se às atividades atômicas e moleculares do meio em que está ocorrendo o fenômeno. A difusão, no caso específico da condução de calor, é vista como a transferência de energia de partículas mais energéticas para partículas de menor energia devido às interações entre elas (INCROPERA e DeWITT, 1999). O modelo matemático empregado é a equação da difusão bidimensional, com condições de contorno de Dirichlet, como detalhado na Fig. 4.1, e assim definido:

$$T(x,0) = T(0,y) = T(1,y) = 0,$$
 (4.1)

e

$$T(x, 1) = sen(\pi x).$$
 (4.2)

A equação de Laplace 2D (LOMAX et al., 2003) é definida pela equação

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0, (4.3)$$

para $0 \le x \le 1$ e $0 \le y \le 1$, onde x e y são as coordenadas e T é a temperatura. Para a Eq. (4.3) existem diversos métodos matemáticos e numéricos que possibilitam a solução do problema. Dentre os mais utilizados encontram-se o método de separação de variáveis (OZISIK, 1994 e BOYCE, 1995) e a análise por diferenças finitas.

As variáveis de interesse para este problema são:

(a) Variável primária local: temperatura T_C definida pela solução da Eq. (4.3) no ponto

$$x = y = \frac{1}{2}. (4.4)$$

(b) Variável primária local: temperatura T_2 definida pela solução da Eq. (5.3) no ponto

$$x = \frac{1}{2} \quad e \quad y = \frac{3}{4}. \tag{4.5}$$

(c) Variável secundária global: temperatura média T_m definida por

$$T_{m} = \frac{1}{L_{x}L_{y}} \int_{0}^{Lx} \int_{0}^{Ly} T(x,y)dx dy,$$
 (4.6)

sendo Lx = Ly = 1.

(d) Variável secundária local: taxa de transferência de calor Q_e definida por

$$Q_e = -k \int_0^{Ly} \left(\frac{\partial T}{\partial x}\right)_{x=1} dy.$$
 (4.7)

(e) Variável secundária local: taxa de transferência de calor Q_n definida por

$$Q_n = -k \int_0^{Lx} \left(\frac{\partial T}{\partial y}\right)_{y=1} dx.$$
 (4.8)

Para verificação das soluções numéricas sem extrapolação e a evolução da redução do erro de discretização, por meio da utilização de MER, será utilizada a solução analítica da Eq. (4.3) que é

$$T(x,y) = sen(\pi x) \frac{senh(\pi y)}{senh(\pi)}.$$
 (4.9)

4.2. MODELO NUMÉRICO

O modelamento numérico proposto é resolvido por meio do método de diferenças finitas. Os elementos da malha são quadrados e uniformemente distribuídos ao longo do domínio. A forma quadrada dos elementos possibilita gerar uma série de resultados com ordem de precisão e eficiência computacional adequadas para resolver este problema com extrapolação.

Para a discretização da Eq. (4.3) utilizou-se para as derivadas de segunda ordem o esquema *CDS* que é uma aproximação de segunda ordem de acurácia, conforme descrito a seguir.

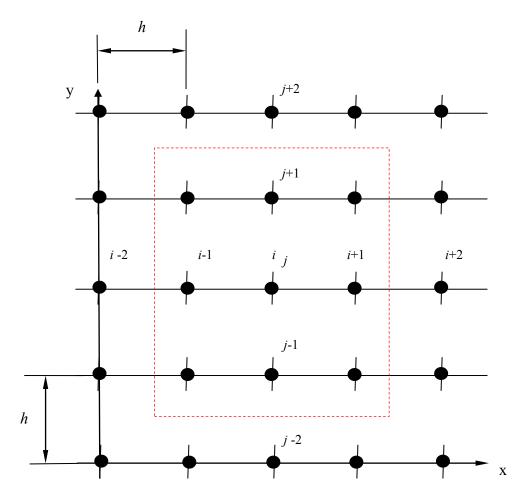


Figura 4.2 - Malha bidimensional com 25 nós.

A aproximação numérica da derivada segunda por *CDS* utilizando a série de Taylor, para o ponto a montante:

$$T_{i-1} = T_i - T_i^1 h + T_i^2 \frac{h^2}{2} - T_i^3 \frac{h^3}{6} + T_i^4 \frac{h^4}{24} - T_i^5 \frac{h^5}{120} + \cdots,$$
 (4.10)

e outro ponto a jusante:

$$T_{i+1} = T_i + T_i^1 h + T_i^2 \frac{h^2}{2} + T_i^3 \frac{h^3}{6} + T_i^4 \frac{h^4}{24} + T_i^5 \frac{h^5}{120} + \cdots;$$
 (4.11)

fazendo a soma das Eqs. (4.10) e (4.11), resulta

$$T_{i+1} + T_{i-1} = 2 T_i + T_i^2 h^2 + T_i^4 \frac{h^4}{12} + \cdots;$$
 (4.12)

isolando a derivada de segunda ordem da Eq. (4.12):

$$T_i^2 = \frac{T_{i-1} - 2T_i + T_{i+1}}{h^2} - T_i^4 \frac{h^2}{12} - T_i^6 \frac{h^4}{360} - T_i^8 \frac{h^4}{20160} - \cdots.$$
 (4.13)

Utilizando o procedimento para a coordenada y,

$$T_j^2 = \frac{T_{j-1} - 2T_j + T_{j+1}}{h^2} - T_j^4 \frac{h^2}{12} - T_j^6 \frac{h^4}{360} - T_j^8 \frac{h^4}{20160} - \cdots.$$
 (4.14)

Substituindo as Eqs. (4.13) e (4.14) na Eq. (4.13) e fazendo

$$\frac{\partial^2 T}{\partial x^2} = T_i^2 \quad e \quad \frac{\partial^2 T}{\partial y^2} T_j^2 ,$$

resulta a equação discretizada

$$\frac{T_{i-1} - 2T_i + T_{i+1}}{h^2} + \frac{T_{j-1} - 2T_j + T_{j+1}}{h^2} = 0.$$
 (4.15)

A Eq. (4.15) tem acurácia de segunda ordem. As ordens verdadeiras do erro de discretização, obtidas *a priori*, têm valor 2,4,6,8.... sendo a ordem assintótica 2 observada nas Eqs. (4.14) e (4.15). As variáveis locais *TC* e *T2* são obtidas por meio da Eq. (4.15) em diferentes pontos da malha. Apesar de o procedimento descrito nas Eqs. (4.10) até (4.15) ser conhecido na literatura especializada, neste momento é importante detalhá-lo para evidenciar

as ordens verdadeiras e assintóticas do erro de discretização. Por meio da metodologia de MER, será possível demonstrar *a posteriori* esses mesmos valores obtidos *a priori*.

A temperatura média ($T_{\rm m}$) é obtida por meio da integração numérica pela regra do trapézio (KRAYSZIG, 1999). As ordens do erro para $T_{\rm m}$ a priori são 2,4,6,...

As soluções para a taxa de calor Q_e e Q_n são obtidas por meio de integração numérica pela regra do trapézio antecedida pelo uso do esquema UDS-2 (TANNEHILL et al., 1997). As ordens do erro são 2,3,4,..

A média da norma l_1 do erro de discretização é definida pela Eq. (4.16)

$$L = \frac{\sum_{P=1}^{N_{ij}} \left| T_P^{analítico} - T_P^{numérico} \right|}{N}, \tag{4.16}$$

onde p representa cada um dos N_{ij} nós da malha e N representa todos os nós da malha. As ordens do erro são 2,4,6,...

4.3. METODOLOGIA

Para as soluções do modelamento matemático e numérico foi utilizado um microcomputador com um processador Intel® Xeon Quad Core X5355 com 2,66 GHz e com 16GB memória RAM e sistema operacional Windows® XP 64 bits. O *software* utilizado para a solução analítica foi o Maple® 7.00. Para os cálculos efetuados com esse *software* utilizaram-se precisões de 50 casas decimais; os valores analíticos de cada variável de interesse estão apresentados no apêndice A.

Para as soluções numéricas os programas foram elaborados em Fortran® 95, versão 9.1 da Intel®, usando precisão dupla (Real*8) e precisão quádrupla (Real*16).

As soluções numéricas sem extrapolação do sistema de equações resultante da Eq. (4.15) foram obtidas com MSI (Modified Strongly Implicit Method) com tolerância de -1,000000000e-01 e multigrid, esquema FAS (Full Aproximation Scheme), ciclo V, restrição por injeção, prolongação por interpolação bilinear e razão de engrossamento dois como acelerador de convergência. O número de vezes que o ciclo V do método multigrid é repetido é denominado iterações externas. O número de iterações externas utilizadas foi 50. O número de iterações internas utilizado foi, no máximo, 8. Como estimativa inicial foi adotado o valor zero para cada problema. O processo iterativo foi levado até atingir o erro de arredondamento de máquina para solução das variáveis de interesse. O programa desenvolvido para as soluções numéricas sem extrapolação é denominado LP 2008 64 BITS. As soluções

numéricas com extrapolação foram obtidas por meio do pós-processamento utilizando MER. O programa atualizado que calcula os resultados numéricos com MER denomina-se Richardson_3p0.

4.4. RESULTADOS

Os resultados das soluções numéricas em precisão dupla e quádrupla, sem extrapolação, das variáveis de interesse estão representados nas Tabs. 4.1 e 4.2. Para precisão dupla, as malhas utilizadas foram 3x3, 5x5, 9x9, ... até 8193x8193.

Tabela 4.1 – Valores numéricos obtidos para as variáveis $T_{\rm C}$, $T_{\rm 2}$, $T_{\rm m}$, $Q_{\rm e}$ e $Q_{\rm n}$ com precisão dupla.

g	Malha	<i>T_c</i> (°C)	<i>T</i> ₂ (°C)	T_m (°C)
1	3x3	2,5000000000000000e-01	NA	1,875000000000000e-01
2	5x5	2,133883476483184e-01	NA	1,908986628323782e-01
3	9x9	2,029152235218276e-01	NA	1,874672742674199e-01
4	17x17	2,001880229640515e-01	4,537806753127554e -01	1,862806579876910e-01
5	33x33	1,994988165854051e-01	4,529617145016377e-01	1,859620913975416e-01
6	65x65	1,993260416376187e-01	4,527562318260793e-01	1,858810564917647e-01
7	129x129	1,992828181477035e-01	4,527048144582445e-01	1,858607103053255e-01
8	257x257	1,992720104130047e-01	4,526919571929576e-01	1,858556182864602e-01
9	513x513	1,992693083629209e-01	4,526887426938874e-01	1,858543449396471e-01
10	1025x1025	1,992686328430914e-01	4,526879390576574e-01	1,858540265815529e-01
11	2049x2049	1,992684639627249E-01	4,526877381479156E-01	1,858539469907025E-01
12	4097x4097	1,992684217426638E-01	4,526876879205233E-01	1,858539270929320E-01
13	8193x8193	1,992684111877090E-01	4,526876753639588E-01	1,858539221185398E-01

Malha	Q_{e}	Qn
3x3	7,500000000000000e-01	-7,500000000000000e-01
5x5	8,946067811865477e-01	-1,281336321356054e+00
9x9	9,132861795885330e-01	-1,623963668394075e+00
17x17	9,163086360410141e-01	-1,812452925714116e+00
33x33	9,169492978576036e-01	-1,909498675897294e+00
65x65	9,171020725172452e-01	-1,958425170640899e+00
129x129	9,171398009514844e-01	-1,982944860186459e+00
257x257	9,171492039311143e-01	-1,995212797918688e+00
513x513	9,171515528546949e-01	-2,001348009462359e+00
1025x1025	9,171521399717018e-01	-2,004415826577430e+00
2049x2049	9,171522867439658E-01	-2,005949775445733E+00
4097x4097	9,171523234369036E-01	-2,006716758385445E+00
8193x8193	9,171523326099151E-01	-2,007100251783917E+00

NA: valor inexistente.

Para precisão quádrupla atingiu-se até a malha 4097x4097, portanto g= 12 e m=11.

Tabela 4.2 - Valores numéricos obtidos para as variáveis $T_{\rm C}$, $T_{\rm 2}$, $T_{\rm m}$, ${\rm Q_e\,e\,Q_n}$ com precisão quádrupla.

g	Malha	T _c (°C)	T ₂ (°C)
1	3x3	2,5000000000000000000000000000000000000	NA
2	5x5	2,13388347648318440550105545263106e-01	NC
3	9x9	2,02915223521827643544500372420873e-01	NC
4	17x17	2,00188022964051459940078588838138e-01	4,53780675312755246196160896311431e-01
5	33x33	1,99498816585405126953842022419956e-01	4,52961714501638078682713059299108e-01
6	65x65	1,99326041637617006794169496115016e-01	4,52756231826077278275747459203198e-01
7	129x129	1,99282818147703110992801706628851e-01	4,52704814458246236256701526388922e-01
8	257x257	1,99272010413013183225536686816414e-01	4,52691957192968525175571247902661e-01
9	513x513	1,99269308362910985036797119708079e-01	4,52688742693869017749432531678484e-01
10	1025x1025	1,99268632843107929474872530584075e-01	4,52687939057669251249462870091549e-01
11	2049x2049	1,99268463962702311320566995441332e-01	4,52687738147905236290429055258675e-01
12	4097x4097	1,99268421742572478232092181079543e-01	4,52687687920419602690232490198047e-01

Malha	T _m (°C)	^
	1	Qe
3x3	1,875000000000000000000000000000000000000	7,5000000000000000000000000000000000000
5x5	1,90898662832378148153250334999939e-01	8,94606781186547524400844362104849e-01
9x9	1,87467274267419889821235462351880e-01	9,13286179588532776020437417774354e-01
17x17	1,86280657987690843810642980680824e-01	9,16308636041013664380905599321289e-01
33x33	1,85962091397541372037734685245441e-01	9,16949297857603415749161632959661e-01
65x65	1,85881056491764510355759359929910e-01	9,17102072517242345869368807882440e-01
129x129	1,85860710305325364361040548078565e-01	9,17139800951482541796456812464513e-01
257x257	1,85855618286464271474330886831590e-01	9,17149203931135244626681255091502e-01
513x513	1,85854344939639347055099527439316e-01	9,17151552854678044615446901217790e-01
1025x1025	1,85854026581549794204613010736512e-01	9,17152139971715377443053209152339e-01
2049x2049	1,85853946990690925434269521543532e-01	9,17152286743858988952254447864673e-01
4097x4097	1,85853927092892677848243781297467e-01	9,17152323436450156117907385210727e-01

Malha	$Q_{\rm n}$
3x3	-7,5000000000000000000000000000000000000
5x5	-1,28133632135605431360096498526268e+00
9x9	-1,62396366839407444925242031102055e+00
17x17	-1,81245292571411632714934172813516e+00
33x33	-1,90949867589729247765051116870470e+00
65x65	-1,95842517064090358783909263092239e+00
129x129	-1,98294486018645635101438039271584e+00
257x257	-1,99521279791865055226098400665065e+00
513x513	-2,00134800946238532541068546185166e+00
1025x1025	-2,00441582657743037259630213000425e+00
2049x2049	-2,00594977544582935162245432141130e+00
4097x4097	-2,00671675838533232578612532810146e+00

Para atingir o erro de arredondamento de máquina, o tempo máximo de CPU, para a malha mais fina 8193x8193, em precisão dupla e 50 iterações externas, foi de 3h23min22s; já para a malha mais fina 4097x4097, em precisão quádrupla e 20 iterações externas, o tempo foi de 1h57min54s. As iterações externas foram suficientes para atingir o erro de arredondamento de máquina. Na Tab. 5.3 é mostrado que os erros de arredondamento acontecem antes das 20 iterações para precisão dupla, isto é, acontecem em 5 iterações externas e, para precisão quádrupla, acontecem com 11 iterações externas.

Tabela 4.3 – Iterações Externas para atingir o erro de arredondamento de máquina.

Iterações	T _c Numérico	T _C Numérico
Externas	Real*8 -> 13ª Malha	r _c Numerico Real*16 → 12ª Malha
1		
1	0,0000000000000000e+00	4,02924217877668953826603464619858e-336
2	1,990890852271615e-01	1,99089097172425334487598593159582e-01
3	1,992682292048621e-01	1,99268239762687582418254845059863e-01
4	1,992684110249286e-01	1,99268421580109260231643005665022e-01
5	1,99268411187 <u>7541</u> e-01	1,99268421742429006172913394218150e-01
6	1,99268411187 5611 e-01	1,99268421742572353197178902152227e-01
7	1,99268411187 5612 e-01	1,99268421742572478123356337765286e-01
8	1,99268411187 3360 e-01	1,99268421742572478231997710708793e-01
9	1,99268411187 6094 e-01	1,99268421742572478232092098973490e-01
10	1,99268411187 7078 e-01	1,99268421742572478232092181008299e-01
11	1,99268411187 <u>7144</u> e-01	1,99268421742572478232092181079 552 e-01
12	1,99268411187 7820 e-01	1,99268421742572478232092181079 544 e-01
13	1,99268411187 <u>6669</u> e-01	1,99268421742572478232092181079 <u>645</u> e-01
14	1,99268411187 <u>3125</u> e-01	1,99268421742572478232092181079 <u>602</u> e-01
15	1,99268411187 2915 e-01	1,99268421742572478232092181079 <u>604</u> e-01
16	1,99268411187 <u>3638</u> e-01	1,99268421742572478232092181079 <u>671</u> e-01
17	1,99268411187 <u>7645</u> e-01	1,99268421742572478232092181079 <u>533</u> e-01
18	1,99268411187 4294 e-01	1,99268421742572478232092181079 698 e-01
19	1,992684111875337e-01	1,99268421742572478232092181079644e-01
20	1,992684111875969e-01	1,99268421742572478232092181079543e-01
21	1,992684111875833e-01	NA NA
22	1,992684111874422e-01	NA
23	1,992684111873862e-01	NA
24	1,992684111878018e-01	NA
25	1,992684111876600e-01	NA
26	1,992684111876467e-01	NA
27	1,992684111876663e-01	NA
28	1,992684111874714e-01	NA
29	1,992684111874343e-01	NA
30 31	1,992684111876510e-01 1,992684111878230e-01	NA NA
32	1,992684111875913e-01	NA NA
33	1,992684111874938e-01	NA
34	1,992684111877033e-01	NA
35	1,992684111875016e-01	NA
36	1,992684111875349e-01	NA
37	1,992684111875784e-01	NA
38	1,992684111874276e-01	NA
39	1,992684111875217e-01	NA
40	1,992684111876448e-01	NA
41	1,992684111876495e-01	NA
42	1,992684111875841e-01	NA
43	1,992684111875619e-01	NA NA
44	1,992684111873080e-01	NA NA
45 46	1,992684111875115e-01 1,992684111876102e-01	NA NA
47	1,992684111876102e-01 1,992684111876973e-01	NA NA
48	1,992684111873085e-01	NA NA
49	1,992684111873302e-01	NA
50	1,99268411187 7090 e-01	NA

NA: valor inexistente.

Para medir o erro com a Eq. (1.1), a solução analítica (Φ) das variáveis $T_{\rm C}$, $T_{\rm 2}$, $T_{\rm m}$, $Q_{\rm e}$ e $Q_{\rm n}$ foi obtida com 12 e 30 algarismos para precisão dupla e quádrupla por meio do *software* Maple.

Os resultados deste capítulo mostram, com base nas variáveis de interesse, para MER e para o estimador de Richardson:

- (a) O efeito da precisão dos cálculos das soluções numéricas como aperfeiçoamento para utilizar MER.
- (b) A efetiva redução do erro de discretização com MER e sua estimativa com Richardson.
- (c) Verificação de ordens de erro e as consequências para MER ao adotar as ordens erradas do erro.

Observação: para os gráficos mostrados neste capítulo estão disponíveis no apêndice B as suas respectivas tabelas de dados.

4.4.1 Efeito da Precisão dos Cálculos das Soluções Numéricas como Aperfeiçoamento para MER

A precisão dos cálculos para variáveis $T_{\rm C}$, $T_{\rm 2}$, $T_{\rm m}$, L, $Q_{\rm e}$ e $Q_{\rm n}$ afeta significativamente MER. A Tab. 4.4 mostra para $T_{\rm C}$ as diferenças entre os resultados numéricos sem extrapolação obtidos em precisão dupla e quádrupla. Os valores foram extraídos da Tabs. 4.1 e 4.2 da variável $T_{\rm C}$ especificamente na malha 513x513. Na Tab. 4.4, os números em negrito e grifados representam os algarismos onde se inicia-se o erro de arredondamento. O efeito desta precisão é mostrado na Fig. 4.3. Este gráfico mostra, para as variáveis de interesse, o tamanho da malha h versus o módulo do erro numérico com e sem extrapolação e com diferentes precisões. O erro de arredondamento acumulativo contido nos resultados numéricos antecipa a dissipação dos resultados com MER e promove o aumento do erro numérico à medida que h é reduzido. Neste ponto, o erro de arredondamento se torna a principal fonte de erro da solução apresentada.

Portanto, a precisão simples, dupla e quádrupla dos cálculos para obtenção da solução numérica promove a dissipação dos resultados da solução numérica com MER.

Tabela 4.4 – Cálculo da diferença entre precisão dupla e quádrupla para a variável T_c .

(a) $T_{\rm C}$ em precisão dupla: 1,992693083629**209**e-01

(b) $T_{\rm C}$ em precisão quádrupla: 1,99269308362910985036797119708**079**e-01

Modulo da diferença (a) - (b) : +0,9914963202880291921e-14

O erro de arredondamento de máquina ocorre na 13ª casa decimal para precisão dupla e 30ª casa decimal para precisão quádrupla. O módulo da diferença entre os resultados é da ordem de e-14.

A Fig. 4.3 mostra para as variáveis de interesse, $T_{\rm C}$, $T_{\rm 2}$, $T_{\rm m}$, L, $Q_{\rm e}$ e $Q_{\rm n}$, o módulo do erro numérico. Eh é o erro da solução numérica sem extrapolação calculada diretamente pela Eq. (1.1); Emer é o erro da solução numérica extrapolada, calculada pela Eq. (3.1) e Eq.(1.1) em precisão dupla (real*8) e quádrupla (real*16) para os limites g=[2,13] e m=[1,12].

O erro da solução numérica Eh (real*8) obtido para Laplace 2D é da ordem de 10^{-9} a 10^{-8} para o tamanho de malha de aproximadamente 10^{-4} . Observa-se que os erros de arredondamento não interferem nestes resultados e este erro decai monotonicamente quando $h \rightarrow 0$.

Para obtenção do erro numérico com extrapolação Emer (real*8) o resultado com MER é mais eficiente do que Eh (real*8). O Emer é da ordem de 10⁻¹⁴ a 10⁻¹³ para um tamanho de malha de aproximadamente 10⁻². Representa uma redução no erro numérico na ordem (Eh(real*8) /Emer(real*8)) de aproximadamente 100.000 vezes com uma malha 100 vezes maior. O esforço computacional é menor. Observa-se também que a solução numérica, para as diversas variáveis de interesse, em certo ponto nos gráficos mostrados na Fig. 4.3, é contaminada por erros de arredondamento. O efeito é caracterizado quando o módulo do erro numérico muda de direção aumentando o seu valor mesmo com h→0.

No caso de Emer (real*16) o resultado é mais eficiente do que Emer (real*8) e Eh (real*8). O Emer (real*16) é da ordem de 10^{-32} a 10^{-28} para um tamanho de malha de aproximadamente 10^{-3} . Representa uma redução no erro numérico na ordem (Eh(real*8)/Emer(real*16)) de aproximadamente $1x10^{20}$ vezes com uma malha em torno de 10 vezes maior. O esforço computacional é menor que Eh(real*8) e maior que Emer(real*8). O resultado numérico é igualmente contaminado como Emer (real*8), porém neste caso é alcançado o erro de máquina.

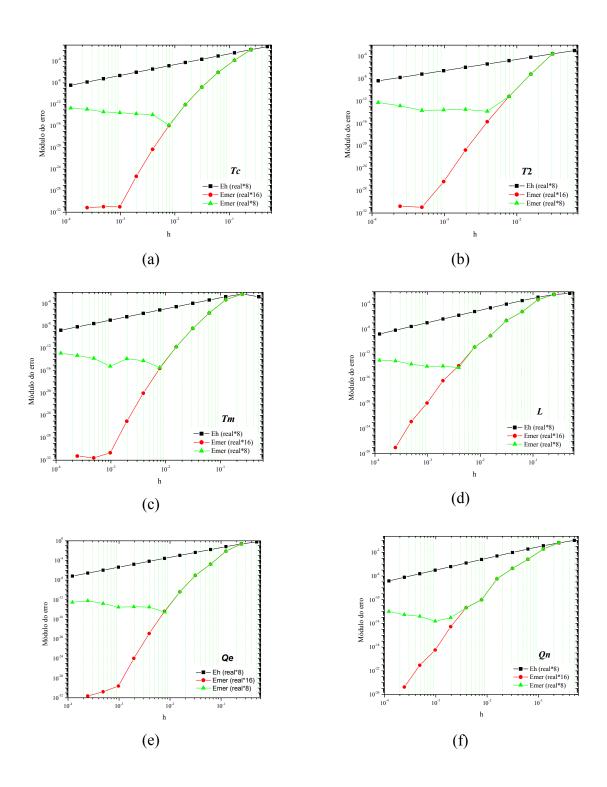


Figura 4.3 - Comparativo entre o módulo do erro em precisão dupla e quádrupla para as variáveis T_C , T_2 , T_m , L, Q_e , Q_n versus tamanho da malha (h) .

A perda de acurácia devido ao erro de arredondamento como mostrado na Tab. 4.4 representa em média (Emer (real*8) /Emer(real*16)) $1x10^{15}$ vezes. No entanto, há um custo computacional adicional ao se usar real*16.

4.4.2 A Efetiva Redução do Erro de Discretização com MER e sua Estimativa com Richardson

A Figs. 4.4 e 4.5 mostram o módulo do erro numérico com ou sem extrapolação em função de h em precisão dupla (real*8) e quádrupla (real*16) para as variáveis de interesse. Uh é a estimativa do erro numérico Eh, calculado pela Eq. (3.19) para os limites g=[2,13] e m=[0, 11]. Umer é a estimativa do erro de Emer calculado também por meio da Eq. (3.19).

O numerador da Eq. (3.19) é definido por $d\phi$ calculado por

$$d\phi = \phi_{g,m} - \phi_{g-1,m}, \tag{4.17}$$

para os seguintes limites g=[2,13] e m=[0,11].

É possível verificar nas seis variáveis de interesse que o erro de discretização é reduzido efetivamente com o uso de MER. O erro de discretização de Eh é reduzido monotonicamente quando $h \rightarrow 0$. Entretanto, este erro é reduzido mais rapidamente com o uso de MER. É possível verificar o fato apresentado nas curvas das Fig. 4.4 e 4.5, onde Emer reduz drasticamente o erro de discretização à medida que $h \rightarrow 0$. O erro de discretização somente deixa de ser reduzido quando o erro de arredondamento contamina o resultado fazendo com que Emer aumente com a diminuição de h.

Para precisão dupla e quádrupla, MER reduz o esforço computacional devido este método atingir, com malhas mais grosseiras, acurácia mais elevada do que soluções numéricas sem extrapolação.

A associação de solução numérica sem extrapolação com MER torna-se interessante devido ao tempo de processamento. Para atingir erro numérico desejado sem MER é necessário gastar muito mais tempo na simulação numérica utilizando uma malha refinada. Entretanto, MER exige várias soluções numéricas em diferentes malhas diferentemente das soluções sem MER, porém estas malhas são muito mais grosseiras comparadas com necessidade sem MER.

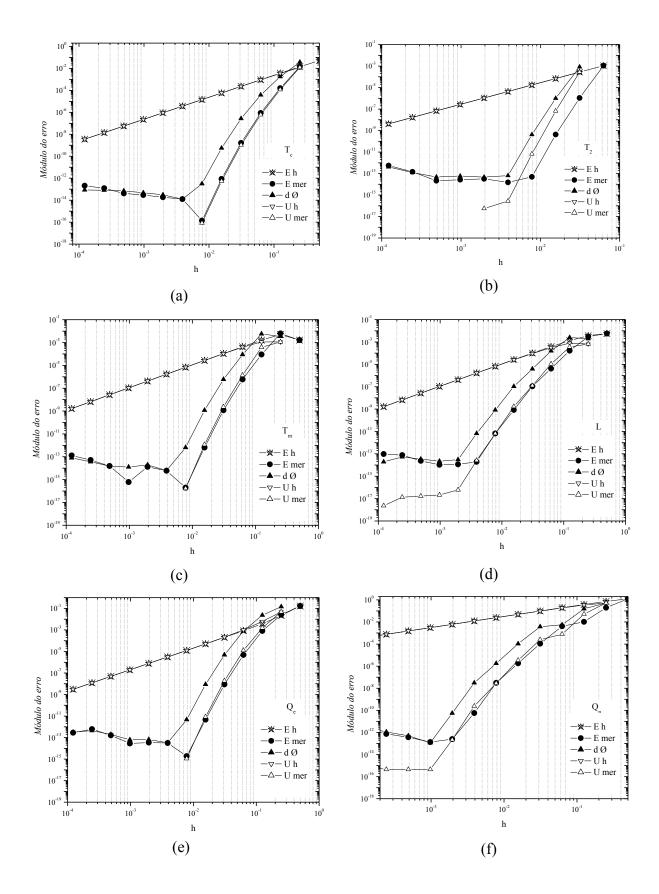


Figura 4.4 - Módulo do erro da solução numérica das variáveis T_C , T_2 , T_m , L, Q_e , Q_n versus tamanho da malha h Real*8.

O estimador definido pela Eq. (3.19) é extremamente eficiente. Nos gráficos apresentados nas Figs. 4.4 e 4.5 são observados, de forma geral, que as curvas de Uh estão praticamente sobre a Eh e Umer estão sobre Emer. A estimativa do erro numérico, tanto para Eh quanto para Emer, funciona bem. O estimador deixa de ser eficiente quando os erros de arredondamento também contaminam Umer. O efeito disso é o aumento do erro numérico estimado quando h é reduzido.

A redução do custo computacional para a variável *T*c com e sem MER é mostrada na Tab. 4.5. Para esta análise é então fixada a ordem do erro numérico em e-9. Para atingir esta ordem de erro por meio de solução numérica sem extrapolação é necessária uma malha de 4097x4097 que corresponde a um tempo de processamento de 7134s ou 1h58min6s.

Entretanto, para atingir a mesma ordem de erro numérico por meio de solução numérica extrapolada foram necessárias 5 malhas: 3x3, 5x5, 9x9, 17x17, 33x33 e MER. Isto correspondeu a gastar um tempo de CPU de 0,45 s. Comparativamente, o tempo percentual com e sem MER é da ordem de e-5 %, isto é, insignificante comparado com a necessidade de tempo para a solução sem MER.

Fica evidenciado que o uso de MER reduz o tempo de processamento mesmo que haja tempos improdutivos de *input* de dados. Para o caso sem MER, a memória RAM foi extremamente exigida devido a trabalhar com os cálculos em uma malha 4097x4907.

Tabela 4.5 – Análise do Custo Computacional para a variável T_c .

Método	Ordem do	Malha	Tempo de CPU
	erro numérico		
Sem MER	e-9	4097x4097	7134
		3x3	
		5x5	
Com MER	e-9	9x9	0,45
		17x17	(Σ dos tempos parciais sem inputs de
		33x33	dados)
		MER	

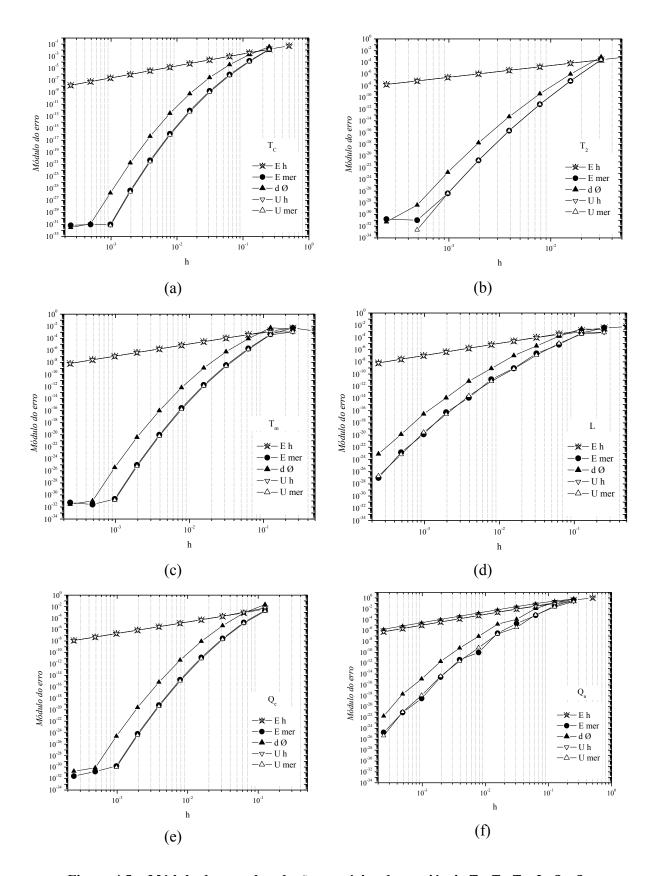


Figura 4.5 - Módulo do erro da solução numérica das variáveis T_C , T_2 , T_m , L, Q_e , Q_n versus tamanho da malha h Real*16.

As curvas de Emer mostradas nas Fig. 4.5 (a), (b), (c), (d), (e) e (f) são obtidas por meio da união dos primeiros pontos de cada extrapolação m representados na Fig. 4.6. Esta figura mostra que, a cada extrapolação, a ordem da solução numérica aumenta e assim reduz o erro de discretização. O mesmo raciocínio equivale para a Fig. 4.4

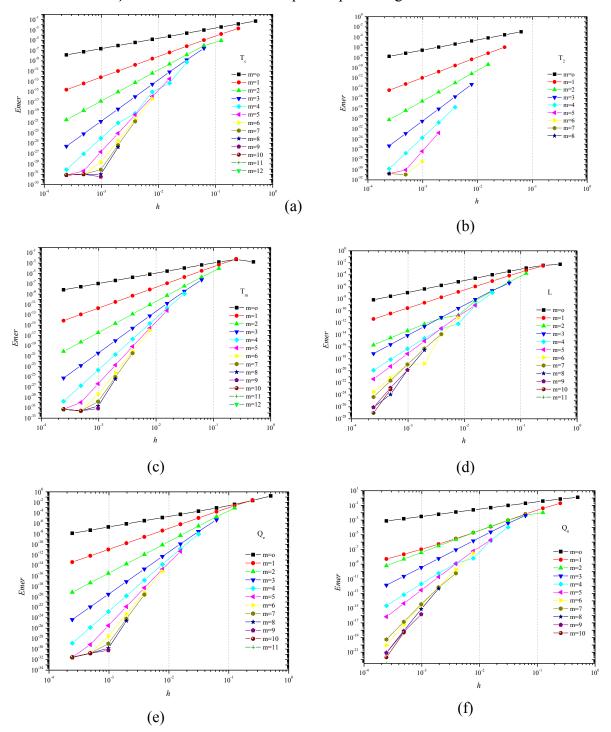


Figura 4.6 - Erro (E) das variáveis T_C , T_2 , T_m , L, Q_e , Q_n versus tamanho da malha (h) e número de extrapolações (m) em Real*16.

4.4.3 Verificação de Ordens de Erro e as Consequências para MER ao Adotar Ordens Erradas do Erro

A Fig. 4.7 mostra as ordens efetivas para a variável Tc (P_E) em função do tamanho da malha h calculada a partir da Eq. (3.9). À medida que $h \rightarrow 0$, as ordens efetivas devem tender às ordens verdadeiras. Portanto para a variável Tc, as ordens verdadeiras observadas no gráfico são 2,4,6,8,10,12,14,16 onde:

p0=2; p1=4; p2=6; p3=8; p4=10; p5=12; p6=14 e p7=16.

É confirmado que as ordens verdadeiras formam uma progressão aritmética conforme Eq.(3.2).

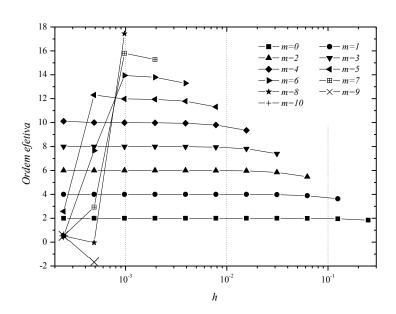


Figura 4.7 - Ordem efetiva (p_E) para Tc *versus* o tamanho da Malha (h) e número de extrapolações (m).

A Fig. 4.8 mostra em função de h a ordem aparente $(pu)_{g,m}$ sendo este calculado por meio da Eq. (3.17) para m=[0, 2] e para g=[3,13]. Para as variáveis de interesse são confirmadas a *posteriori* as ordens dos erros obtidas *a priori*. Teoricamente, quando a malha é refinada, a ordem aparente (p_u) tende à ordem verdadeira (p_m) . Este efeito é claramente verificado nos gráficos da Fig. 4.8 onde à medida que $h \rightarrow 0$ a ordem aparente tente à ordem verdadeira.

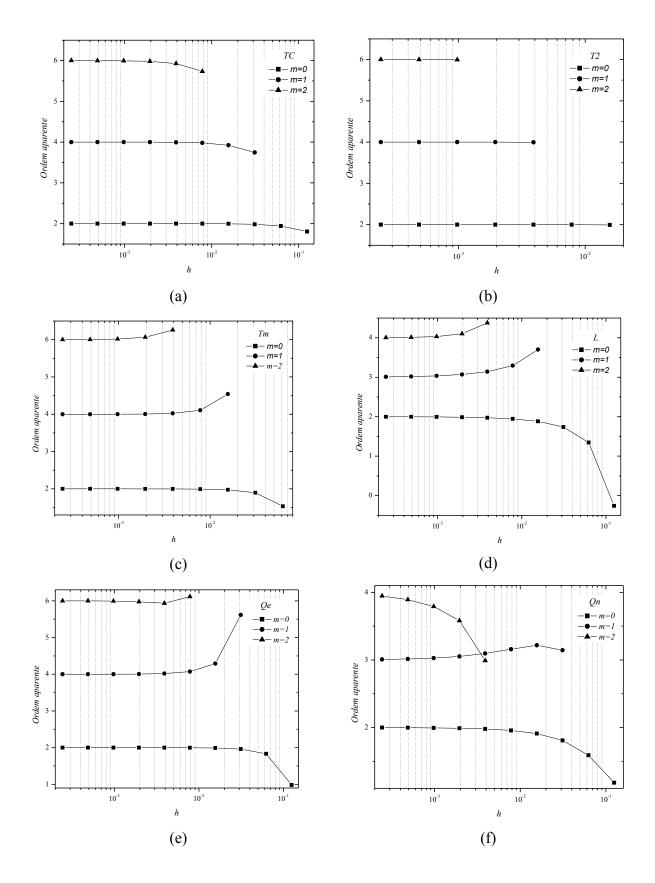


Figura 4.8 - Ordem Aparente (p_u) versus o tamanho da Malha (h) e número de extrapolações (m).

No entanto, é verificado que há diferenças entre os resultados *a priori* e *a posteriori*. As ordens obtidas *a posteriori* estão destacadas na Tab. 4.5 bem como sua comparação com as ordens obtidas *a priori*.

TC 1 1 4 C C	~ , 1	1 1 10 40	~ 1.4.1	
Tabela 4.6 – Compara	acão entre as ordens	do erro de dicretiz	acan obtidas <i>a</i>	nriori e a nosteriori
Tabela no Compara	ição chiti e as oraciis	do ci i o de dici etiz	ação obtidas a	priori e a posiciiorii

Variável	Ordens a priori	Ordens a posteriori
$T_{ m C}$	2,4,6,	2,4,6,
T_2	2,4,6,	2,4,6,
T_{m}	2,4,6,	2,4,6,
L	2,4,6,	2,3,4,
Q _e	2,3,4,	2,4,6,
On	2,3,4,	2,3,4,

A utilização de MER com ordens verdadeiras erradas promove um efeito negativo nas soluções numéricas. A Fig. 4.9 mostra este efeito. Portanto, não há uma efetiva redução do erro de discretização se as ordens verdadeiras aplicadas na Eq. (3.1) não forem as ordens corretas.

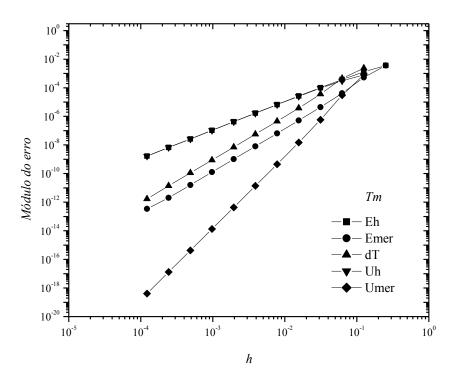


Figura 4.9 - Efeito da utilização da ordem errada de p_m.

4.5. CONCLUSÃO

MER, para a equação de Laplace bidimensional mostrou ser uma ferramenta extremamente eficiente na redução do erro de discretização. Foi possível obter resultados

muito precisos chegando ao erro de arredondamento de máquina. O objetivo de testar MER, em maior número de malhas e observar e apreender o seu comportamento, foi atendido, sendo conclusivo que MER é mais efetivo em precisão quádrupla.

O estimador de erro de Richardson funciona para resultados numéricos obtidos com MER.

Os procedimentos propostos nos capítulos anteriores desta tese funcionam bem. MER é sensível a qualquer tipo de erro inclusive o de arredondamento.

.

5 EQUAÇÃO DE POISSON UNIDIMENSIONAL

No Cap. 4 foi desenvolvido, aperfeiçoado e generalizado o uso de MER utilizando a equação de Laplace 2D; as variáveis de interesse foram a temperatura no centro do domínio, x=y=½, a temperatura em x=½ e y=¾, a média do campo de temperaturas, as taxas de transferência de calor nos contornos de Lx=1 e Ly=1, concluindo-se que MER é eficiente para a redução do erro de discretização. As malhas utilizadas, para o problema de Laplace 2D, foram as malhas 5x5, 9x9, 17x17, 33x33,..., 8193x8193. No entanto, todas essas malhas numéricas possuíam algo em comum: havia soluções numéricas no ponto de interesse das variáveis estudadas. Por exemplo, para o caso da temperatura em x=½ e y=¾, as Figs. 5.1 e 5.2 mostram duas malhas distintas, a 5x5 e a 9x9, sendo o ponto branco as coordenadas x=½ e y=¾, o qual é o ponto de interesse. Observa-se que, com o refino da malha, 5x5 para 9x9, o ponto branco sempre será um nó existente da malha e sempre haverá a solução numérica para esse ponto. Esse raciocínio é válido para outras malhas do capítulo 4, isto é, para as malhas 17x17, 33x33 até 8193x8193.

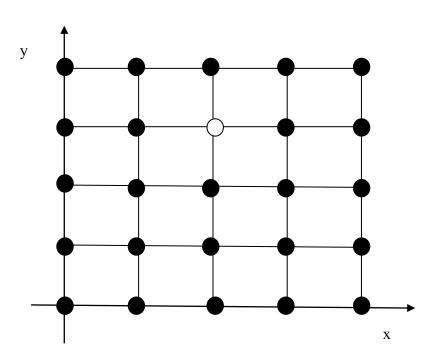


Figura 5.1 - Malha 5x5 - Laplace 2D

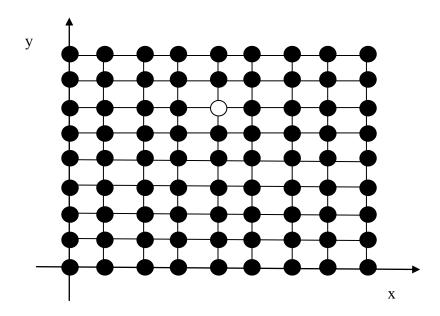


Figura 5.2 - Malha 9x9 - Laplace 2D

Marchi *et al.* (2009) aborda o escoamento dentro de uma cavidade quadrada cuja tampa tem velocidade constante, modelado pela equação de Navier-Stokes. Os resultados obtidos com 10 malhas, utilizando MER para velocidade máxima para *u* e *v* entre outras variáveis, mostram serem elas ineficientes na redução do erro de discretização.

O capítulo 5 procura investigar e tornar MER eficiente para problemas de máximo e mínimo em que estes pontos são indefinidos no domínio estudado.

Portanto, a diferença entre a metodologia aplicada para as equações de Laplace e de Poisson nesta tese consiste em haver nas malhas em que é aplicada a equação de Laplace um nó sempre no ponto de máximo devido este ser conhecido. Para as malhas em que é aplicada a equação de Poisson, o ponto de máximo é desconhecido e não haverá um nó na malha que coincidirá com este ponto.

Observa-se que no estudo de Marchi *et al.* (2009) as variáveis da velocidade eram incógnitas do problema e, por isso, as coordenadas destes pontos de interesse não eram conhecidas *a priori*, como no caso de Laplace 2D.

As hipóteses consideradas no estudo do problema de Poisson 1D são:

Hipótese 1: as malhas com diferentes *h* utilizadas para formar o banco de dados de entrada para MER foram obtidas aleatoriamente, respeitando uma razão de refino definida pelo analista.

Hipótese 2: a frequência com que são obtidos os dados não coincide com o ponto de máximo e, portanto, pode haver um deslocamento entre o nó e o real ponto de máximo como já tratado no item 3.5 desta tese.

No capítulo anterior foi constatado, demonstrado e analisado que MER é sensível a qualquer outro tipo de erro, seja de arredondamento, de programação ou de usuário, inibindo sua eficiência. MER é exclusivamente eficiente para erros de discretização.

Neste capítulo, os dados base são aquelas informações que estão nos nós de interesse obtidos por meio da solução numérica sem extrapolação. Para resolver o problema de Poisson 1D com alta acurácia é introduzido por esta tese o erro de posição da variável de interesse. Esse erro é maior que o erro de discretização, contaminando os dados base e tornando MER ineficiente para a redução do erro de discretização.

Justifica-se o estudo de MER associado a uma equação unidimensional em virtude desta equação ter ponto de máximo indefinido possibilitando <u>avaliar</u>, aperfeiçoar e generalizar o uso MER em variáveis de campo.

A equação utilizada é a de transferência de calor unidimensional por meio de uma parede plana com geração interna de calor. Essa equação é conhecida como equação de Poisson 1D. A geração interna de energia é introduzida na equação como um termo fonte, sendo esta uma função senoidal e assimétrica. Esta formulação com o termo fonte desloca o ponto de temperatura máxima do centro do domínio, isto é, T_{max} e x_{max} em $x \ne 1/2$ no domínio definido por [0,L]. Este ponto de maior temperatura é posicionado em um local onde nem sempre haverá um nó definido. A equação de Poisson é numericamente simples e possibilita grande número de testes para avaliar MER.

5.1. MODELO MATEMÁTICO

O modelo matemático utilizado é a equação de Poisson 1D definida no intervalo de [0,1] com condições de contorno de Dirichlet , desenvolvida a partir da equação da conservação da energia definida por

$$T(0) = 0 \text{ e } T(1) = \frac{\sqrt{2}}{2},$$
 (5.1)

$$\frac{d^2T}{dx^2} + S = 0; (5.2)$$

onde T é a temperatura em °C, x é a coordenada espacial para $0 \le x \le 1$ em metros e S é o termo fonte definido por

$$S = \left(\frac{3\pi}{4}\right)^2 sen\left(\left(\frac{3\pi}{4}\right)x\right); \tag{5.3}$$

portanto, resolvendo a equação diferencial, a solução analítica se resume a

$$T(x) = sen\left(\left(\frac{3\pi}{4}\right)x\right);$$
 (5.4)

Seja T(x) uma função contínua limitada e derivável no intervalo de interesse e possuindo um único ponto de máximo neste intervalo, definido pela equação

$$\frac{d(T)}{dx} = 0; (5.5)$$

o valor da coordenada de máximo definido x máx, no intervalo de [0,1] é

$$x = \frac{2}{3} \cong 6,6666666666 - 01. \tag{5.6}$$

substituindo x na Eq. (5.4) resulta em

$$T_{m\acute{a}x}=1. ag{5.7}$$

O campo de temperatura resultante da Eq. (5.4) com alguns pontos destacados é mostrado na Fig. 5.3. Observa-se que o ponto de máximo está deslocado do centro do domínio.

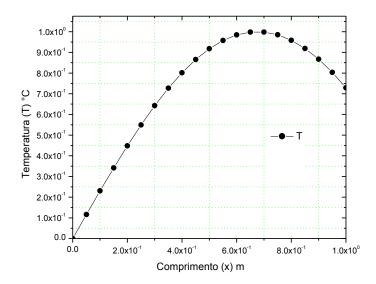


Figura 5.3 - Temperatura T em função do x.

5.2. MODELO NUMÉRICO

A Eq. (5.2) é aproximada pelo método de diferenças finitas utilizando esquema CDS (*Central Difference Scheme*), malha uniforme, a partir da série de Taylor.

Expandindo a série de Taylor para os nós p-1 e p+1 da Fig. 6.7, sendo p o nó e h o tamanho da malha, obtém-se

$$T_{p-1} = T_p - \left(\frac{dT}{dx}\right)_p h + \left(\frac{d^2T}{dx^2}\right)_p \frac{h^2}{2} - \left(\frac{d^3T}{dx^3}\right)_p \frac{h^3}{6} + \left(\frac{d^4T}{dx^4}\right)_p \frac{h^4}{24} - \left(\frac{d^5T}{dx^5}\right)_p \frac{h^5}{120} + \dots$$
 (5.8)

$$T_{p+1} = T_p + \left(\frac{dT}{dx}\right)_p h + \left(\frac{d^2T}{dx^2}\right)_p \frac{h^2}{2} + \left(\frac{d^3T}{dx^3}\right)_p \frac{h^3}{6} + \left(\frac{d^4T}{dx^4}\right)_p \frac{h^4}{24} + \left(\frac{d^5T}{dx^5}\right)_p \frac{h^5}{120} + \dots;$$
 (5.9)

somando as Eqs. (5.8) e (5.9), resulta

$$T_{p-1} + T_{p+1} = 2T_p + \left(\frac{d^2T}{dx^2}\right)_p h^2 + \left(\frac{d^4T}{dx^4}\right)_p \frac{h^4}{12} + \left(\frac{d^6T}{dx^6}\right)_p \frac{h^6}{360} + \left(\frac{d^8T}{dx^8}\right)_p \frac{h^8}{20160} + \cdots; \quad \textbf{(5.10)}$$

isolando o termo $\left(\frac{d^2T}{dx^2}\right)_n$,

$$\left(\frac{d^2T}{dx^2}\right)_p = \frac{T_{p-1} - 2T_p + T_{p+1}}{h^2} - \left(\frac{d^4T}{dx^4}\right)_p \frac{h^2}{12} - \left(\frac{d^6T}{dx^6}\right)_p \frac{h^4}{360} - \left(\frac{d^8T}{dx^8}\right)_p \frac{h^6}{20160} - \dots,$$
(5.11)

a aproximação da Eq. (5.11) pode ser reescrita na forma de aproximação numérica e erro de truncamento, representada por

$$\left(\frac{d^2T}{dx^2}\right)_p = \lambda_{CDS}^{ii} + \varepsilon \left(\lambda_{CDS}^{ii}\right), \tag{5.12}$$

sendo λ_{CDS}^{ii} a aproximação numérica CDS e $\varepsilon(\lambda_{CDS}^{ii})$ o erro de truncamento.

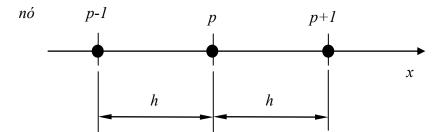


Figura 5.4 - Malha unidimensional uniforme.

Igualando-se as Eqs. (5.12) e (5.11), obtém-se a aproximação numérica para o esquema CDS

$$\lambda_{CDS}^{ii} = \frac{T_{p-1} - 2T_p + T_{p+1}}{h^2},$$
(5.13)

e o erro de truncamento

$$\varepsilon(\lambda_{CDS}^{ii}) = -\left(\frac{d^4T}{dx^4}\right)_p \frac{h^2}{12} - \left(\frac{d^6T}{dx^6}\right)_p \frac{h^4}{360} - \left(\frac{d^8T}{dx^8}\right)_p \frac{h^6}{20160} - \dots;$$
 (5.14)

substituindo-se a Eq. (5.13) na Eq. (5.2) resulta a equação discretizada para Poisson 1D

$$\frac{T_{p-1} - 2T_p + T_{p+1}}{h^2} + S = 0, (5.15)$$

sendo o termo fonte $S = (\pi)^2 sen((\pi)x)$, p representa cada nó da malha, h a distância entre dois nós.

5.3. METODOLOGIA

Para as soluções do modelamento matemático e numérico foi utilizado um microcomputador com um processador Intel® Xeon Quad Core X5355 com 2,66 GHz e com 16GB memória RAM e sistema operacional Windows® XP 64 bits. O *software* utilizado para a solução analítica foi o Maple® 7.00. Para os cálculos efetuados com esse *software* utilizaram-se precisões de 50 casas decimais; os valores analíticos de cada variável de interesse estão apresentados no apêndice A.

Para as soluções numéricas, em diferenças finitas, foram elaborados em Fortran® 95, versão 9.1 da Intel®, usando precisão quádrupla (Real*16).

As soluções numéricas sem extrapolação do sistema de equações resultante da Eq. (5.15) foram obtidas com TDMA (*Tridiagonal Matrix Algorithm*). O programa desenvolvido para as soluções numéricas sem extrapolação é denominado de Poisson_1DP1_64BITS. As soluções numéricas com extrapolação foram obtidas por meio do pós-processamento utilizando MER. O programa atualizado que calcula os resultados numéricos com MER denomina-se Richardson_3p0.

As interpolações polinomiais foram implementadas em Fortran® 95, versão 9.1 da Intel®, usando precisão quádrupla (Real*16) em diferenças divididas. O programa DDivididas.f90 está disponível no apêndice D.

5.4. RESULTADOS OBTIDOS

Para todos os resultados numéricos sem e com extrapolação foram utilizados precisão quádrupla nos cálculos. Para os resultados numéricos sem extrapolação da temperatura máxima e da sua coordenada x foram calculados para até 19 malhas com h variando de $2,500 \times 10^{-1}$ até $1,907 \times 10^{-6}$, ou seja:

$$h = \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \dots, \frac{1}{524288}$$

com razão de refino 2 do domínio de [0,1]. O padrão de refino escolhido foi aleatório, devido desconhecer os pontos de interesse que são a máxima temperatura e a sua coordenada espacial.

Para os resultados numéricos com extrapolação a partir da Eq. (3.1) utilizou-se G=19 e M=18 e $p_m=2,4,6,...$

MER utiliza solução numérica em diferentes malhas para o cálculo das extrapolações. Estas soluções numéricas que MER utiliza são obtidas da malha mais grossa até a malha mais fina para satisfazer a Eq. (3.1).

Os nós que carregam as informações da temperatura máxima e a sua coordenada são desconhecidos e as temperaturas máximas numéricas M^D obtidas em diversas malhas são calculadas por meio da Eq.(3.28). Portanto, existe uma oscilação entre o ponto de máximo da função contínua⁸ M^C definida pela Eq.(3.26) e o máximo discreto M^D mostrado na Fig. 5.5. Nesta figura, mostram-se os nós em três malhas com h=0,5; h=0,25 e 0,125 definidos por círculos cheios. O ponto de máximo contínuo, obtido pela derivada da função continua da equação de Poisson, está mostrado como triângulos.

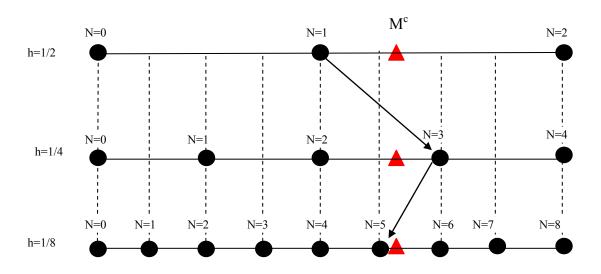


Figura 5.5 – Oscilação da localização do ponto de máximo discreto M^D em diversas malhas em função do ponto de máximo contínuo M^C.

As soluções numéricas para a temperatura máxima em diversas malhas necessárias para a Eq. (3.1) são obtidas para a malha h=0,5 no nó N=1, para h=0,25 no nó N=3 e para h=0,125 no

_

⁸ A solução analítica para Poisson 1D é conhecida, porém nesta tese trata-se como se não fosse. As informações obtidas da Eq. (5.4) são basicamente utilizadas na Eq. (1.1) e para fazer análises no experimento numérico.

nó N=5. Há uma oscilação na posição, como já comentado, entre os nós que carregam a informação de temperatura máxima numérica e o ponto de máximo M^c . A coordenada do ponto de máximo é consequência do resultado da temperatura máxima. O valor de x_{max} é obtido pela medição da distância entre os nós N=1, N=2, N=3,... até a origem do sistema coordenado

A Figs. 5.6 e 5.7 mostram para as variáveis $T_{m\acute{a}x}$ e $x_{m\acute{a}x}$ o resultado de MER ao utilizar as informações em diversas malhas com o procedimento adotado e explicitado na Fig. 5.5. Nas Figs. 5.6 e 5.7 mostram a variável $T_{m\acute{a}x}$ e $x_{m\acute{a}x}$ em função do tamanho da malha (h) o módulo do erro. Neste gráfico, Eh é o erro da solução numérica de ϕ (sem qualquer extrapolação), definido pela Eq. (1.1). d ϕ é definido pela Eq. (4.17); Uh é a estimativa de Eh, definido pela Eq. (3.19). Emer é o erro da solução numérica de ϕ com extrapolação, definido pelas Eqs. (1.1) e (3.1). E, Umer é a estimativa de Emer. Emer decai assintoticamente quando $h\rightarrow 0$. As ordens do erro de discretização *a priori* calculada pela Eq. (5.15) são 2,4,6,....

Para a Fig. 5.6, fixando um tamanho de malha de aproximadamente 10⁻⁴, verifica-se que o ganho na redução do erro de discretização é da ordem de 10⁻⁹ para 10⁻¹². Comparado estes resultados os obtidos no Cap. 4 para precisão quádrupla verifica-se que MER aplicado nestas condições resulta em reduções do erro de discretização muito abaixo do seu potencial. A estimativa do erro (Umer) não funciona para Emer. A estimativa apresentada é sem confiança e inacurada.

Para T_{max} e x_{max} comportamento de U_h coincide com o comportamento de E_h , mostrando que o estimador de erro de Richardson funciona bem quando associado com o erro de discretização da solução numérica sem extrapolação mesmo com erro de posição definido na Eq. (3.36) e Fig. 5.5.

Foram utilizadas as ordens verdadeiras do erro de discretização para a Eq.(3.1), com base no estudo *a priori*, como sendo a ordem assintótica p_L =2 e a sua variação dp_L = 2, isto é p_m = 2,4,6,... tanto para T_{max} e x_{max} .

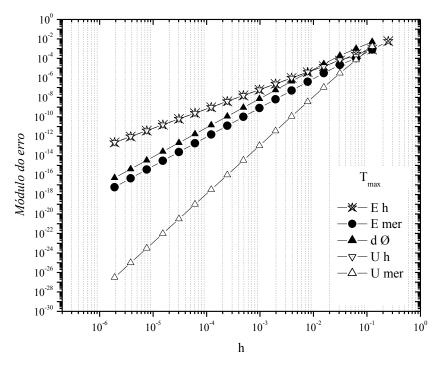


Figura 5.6 - Módulo do erro versus h para a variável T_{max} sem interpolação $p_L=2$ e $dp_L=2$.

Na Fig. 5.7, fixando um tamanho de malha de aproximadamente 10⁻⁴, verifica-se que não há o ganho algum na redução do erro de discretização. Além disso, não há parâmetro para afirmar que Umer funcione para as condições aplicadas.

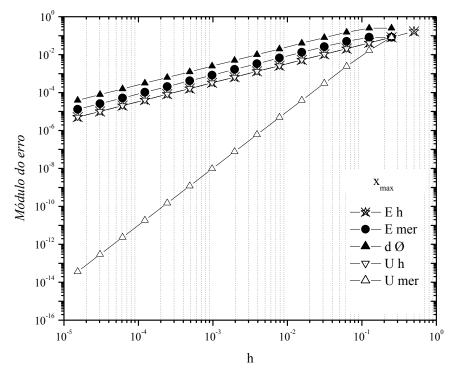


Figura 5.7 - Módulo do erro versus h para a variável x_{max} sem interpolação $p_L=2$ e $dp_L=2$.

As tendências das ordens efetivas para a variável T_{max} são mostradas na Tab. 5.1. Observa-se que para m=0 a ordem efetiva tende claramente a 2, comprovando *a posteriori* a ordem assintótica deduzida *a priori*. Para m=1 a tendência da ordem efetiva é 3, mostrando ser diferente da ordem obtida *a priori*. Esta ordem deveria ser 4.

Após a primeira extrapolação em m=1, não se pode afirmar mais sobre as tendências das ordens efetivas apresentadas na Tab. 5.1. Portanto, a partir de m>1 os resultados são incoerentes e inconsistentes com base na Eq. (3.2).

Para as tendências das ordens efetivas para a variável x_{max} são mostradas na Tab. 5.2. Observa-se que para m=0 a ordem efetiva tende claramente a 1. A respeito das outras ordens efetivas m>0 não se pode afirmar nada a respeito de suas tendências.

Os resultados obtidos na Tab. 5.1 sugerem que as ordens efetivas para a variável T_{max} são para $m=0 \rightarrow P_E=2$ e para $m=1 \rightarrow P_E=3$. Poderia se pensar que as ordens verdadeiras fossem $p_L=2$ com $dp_L=1$. Para confirmar esta nova situação, foi obtido para a variável T_{max} novo resultado utilizando as ordens do erro com sendo 2,3,4... Entretanto, observa-se que o novo resultado apresentado para Emer e mostrado na Fig. 5.8 é pior que o mostrado na Fig. 5.6. As ordens usadas com $dp_L=1$ apresentam ser incoerentes nesta simulação numérica.

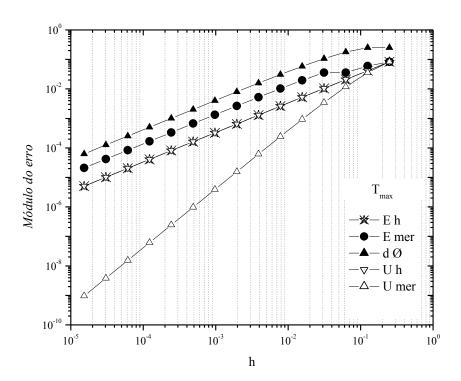


Figura 5.8 - Módulo do erro *versus h* para a variável T_{max} sem interpolação $p_L=2$ e $dp_L=1$.

 $Tabela~5.1-Tendência~da~Ordem~efetiva~(P_E~)~para~a~variável~T_{max}~de~Eh~e~Emer~em~função~de~h.$

h	m=0	m=1	m=2	m=3	m=4
5,0000000e-01					
2,5000000e-01	1,6560263e-01				
1,2500000e-01	2,8998917e+00	2,7437688e+00			
6,2500000e-02	1,5272758e+00	3,1592008e+00	3,0237008e+00		
3,1250000e-02	2,2288458e+00	2,9296199e+00	3,0077427e+00	3,0095331e+00	
1,5625000e-02	1,8839621e+00	3,0373066e+00	3,0016829e+00	3,0023582e+00	3,0025765e+00
7,8125000e-03	2,0575850e+00	2,9818986e+00	3,0004502e+00	3,0005873e+00	3,0006410e+00
3,9062500e-03	1,9711029e+00	3,0091858e+00	3,0001088e+00	3,0001467e+00	3,0001601e+00
1,9531250e-03	2,0144219e+00	2,9954412e+00	3,0000277e+00	3,0000367e+00	3,0000400e+00
9,7656250e-04	1,9927825e+00	3,0022879e+00	3,0000069e+00	3,0000092e+00	3,0000100e+00
4,8828125e-04	2,0036071e+00	2,9988582e+00	3,0000017e+00	3,0000023e+00	3,0000025e+00
2,4414063e-04	1,9981960e+00	3,0005714e+00	3,0000004e+00	3,0000006e+00	3,0000006e+00
1,2207031e-04	2,0009019e+00	2,9997144e+00	3,0000001e+00	3,0000001e+00	3,0000002e+00
	Tendência de Eh	Tendência de Emer	Tendência de Emer	Tendência de Emer	Tendência de Emer
	$m_0 \rightarrow 2$	$m_1 \rightarrow 3$	m ₂ →3	$m_3 \rightarrow 3$	m ₄ →3
			incoerente	incoerente	incoerente

Tabela 5.2 – Tendência da Ordem efetiva (P_{E}) para a variável x_{max} de Eh e Emer em função de h.

h	m=0	m=1	m=2	m=3	m=4
5,0000000e-01					
2,5000000e-01	1,000000000000e+00				
1,2500000e-01	1,000000000000e+00	1,000000000000e+00			
6,2500000e-02	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00		
3,1250000e-02	9,99999999999e-01	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00	
1,5625000e-02	1,000000000000e+00	9,99999999999e-01	1,000000000000e+00	1,000000000000e+00	9,99999999999e-01
7,8125000e-03	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00
3,9062500e-03	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00
1,9531250e-03	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00
9,7656250e-04	1,000000000000e+00	1,0000000000000e+00	1,000000000000e+00	1,000000000000e+00	9,99999999999e-01
4,8828125e-04	9,99999999999e-01	1,0000000000000e+00	1,000000000000e+00	1,000000000000e+00	1,0000000000000e+00
2,4414063e-04	1,000000000000e+00	9,999999999999e-01	1,000000000000e+00	1,000000000000e+00	1,0000000000000e+00
1,2207031e-04	1,000000000000e+00	1,000000000000e+00	1,000000000000e+00	9,99999999999e-01	9,99999999999e-01
	Tendência de Eh	Tendência de Emer	Tendência de Emer	Tendência de Emer	Tendência de Emer
	$m_0 \rightarrow 1$	$m_1 \rightarrow 1$	$m_2 \rightarrow 1$	$m_3 \rightarrow 1$	$m_4 \rightarrow 1$
		incoerente	incoerente	incoerente	incoerente

O problema da ineficiência de MER foi verificado nos ensaios numéricos desenvolvidos e mostrado nas Figs. 5.6, 5.7 e 5.8.

Para confirmar a teoria do erro de posição definido pela Eq.(3.34) é desenvolvido um experimento que fixa em h=2/3 um nó. É planejado um refino de malha que possibilita sempre haver um nó neste ponto quando h→0. As malhas utilizadas nesta análise são 3, 6, 12, 24, 48, 96, 192, 384, 768, 1536, 3072 e 6144. A razão de refino é 2. O ponto h=2/3 coincide com o ponto de temperatura máxima M^C definida pela Eq.(3.26), portanto a coordenada de M^C é igual à coordenada de M^D.

Na Fig. 5.9, o ponto de máximo contínuo é obtido pela derivada da função continua da equação de Poisson e a sua localização está mostrada por triângulos. A localização dos pontos de máximo numérico em diversas malhas é representada por nós negros nas posições a N=2 para a malha h=1/3, N=4 para a malha h=1/6 e N=8 para a malha h=1/12. O ponto de máximo, nas três malhas, foi representado com círculo menor para diferenciar do restante dos nós. Esta diferenciação é para enfatizar que agora M^C=M^D nas três malhas esquematizadas na Fig. 5.9 e diferenciando da Fig. 5.5.

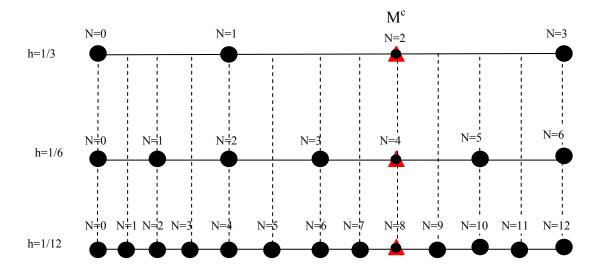


Figura 5.9 – Planejamento para coincidir o ponto de máximo M^C com um nó M^D em diversas malhas.

Na Fig. 5.9 observa-se que não há oscilação na posição como havia na Fig. 5.5.

Na Fig. 5.10 é apresentado o resultado para o experimento conforme descrito na Fig. 5.9. Observa-se que a eficiência de MER está relacionada a haver somente erro de discretização nas soluções numéricas sem extrapolação e a ausência do erro de posição. Emer cai até atingir o erro de arredondamento. Umer se comporta muito bem. A solução obtida para T_{max} a partir das malhas 3, 6, 12, 24, 48, 96, 192, 384, 768, 1536, 3072 e 6144 será definida como *solução ideal e impossível de se obter*.

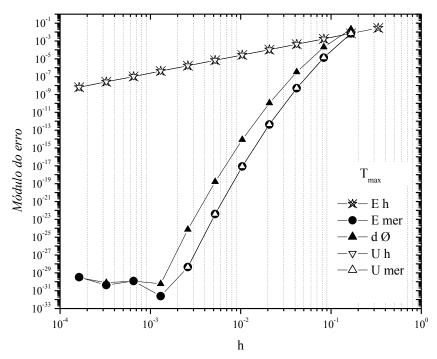


Figura 5.10 – Módulo do erro *versus h* para a variável T_{max} sem interpolação p_L =2 e dp_L =2 sem erro de posição.

As ordens efetivas p_E para a variável T_{max} sem erro de posição são apresentadas na Fig. 5.11 e comprovam *a posterior* as ordens do erro calculadas *a priori*. Os valores obtidos para a solução da Fig. 5.10 são claramente identificados nesta figura como sendo:

p0=2; p1=4; p2=6; p3=8; p4=10 e p5=12.

As demais ordens não são identificadas devido à interferência do erro de arredondamento.

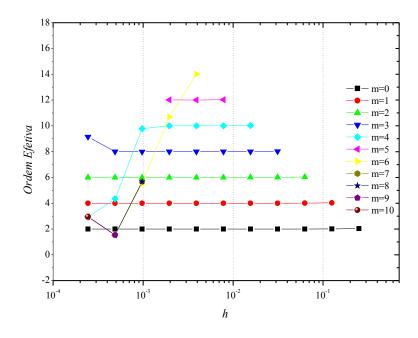


Figura 5.11 – Ordem efetiva (p_E) para T_{max} versus o tamanho da Malha (h) e número de extrapolações (m).

5.4.1 Interpolação polinomial em Poisson 1D com MER

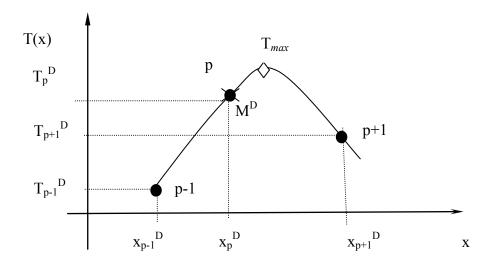
A partir das soluções numéricas obtidas pelo programa Poisson_1Dp_3p1_64BITS associadas à MER conforme Fig. 5.5 foram investigadas três situações para aumentar a eficiência de MER para as variáveis $T_{m\acute{a}x}$ e $x_{m\acute{a}x}$:

- (i) Aplicação de MER a partir do tratamento da base de dados por meio de interpolação de segunda ordem dos nós da malha obtida pelo programa Poisson_1Dp_3p1_64BITS.
- (ii) Aplicação de MER a partir do tratamento da base de dados por meio de interpolação de quarta ordem dos nós da malha obtida pelo programa Poisson_1Dp_3p1_64BITS
- (iii) Aplicação de MER a partir do tratamento da base de dados por meio de interpolação de décima ordem dos nós da malha obtida pelo programa Poisson 1Dp 3p1 64BITS.

É conclusivo que MER não é eficiente devido ao erro de posição contido nas soluções numéricas e comprovado nos resultados mostrados nas Figs. 5.6 e 5.7. O resultado observado na Fig. 5.10 deixa evidente que MER é extremamente eficiente quando a fonte principal de

erro da solução numérica é o erro de discretização. Para minimizar o erro de posição será interpolado nos nós da malha um polinômio.

Na Fig. 5.12 é mostrado o esquema de interpolação para quais os tópicos (i), (ii) e (iii) estão definidos. O ponto de $T_{m\acute{a}x}$ é indefinido como já explicado neste capitulo. Os círculos negros na Fig. 5.12 são as soluções numéricas obtidas pelo programa Poisson_1Dp_3p1_64BITS. O losango representa o ponto de temperatura máxima da função w(T) interpolada a partir dos nós p-1, p e p+1.



Legenda

nós da malha gerados por meio de Poisson_1Dp_3p1_64BITS.

função interpolada w(T)

máximo da função w(T)

Figura 5.12 - Esquema de interpolação dos dados obtidos por Poisson_1Dp_3p1_64BITS.

Para determinar $T_{m\dot{a}x}$ é executada a interpolação polinomial a partir dos pontos da malha. Neste caso, para a interpolação de segunda ordem utiliza-se 3 pontos da malha. Para interpolação de quarta e décima ordens utilizam-se 5 e 11 nós da malha. Os pontos a serem interpolados são escolhidos a partir do ponto de máximo discreto definido por M^D . O nó central é o ponto de máximo discreto definido por p. Como a interpolação nestes casos utiliza números ímpares de nós, o nó central é o ponto de máximo discreto.

A geração da função contínua w(T) é obtida por meio da interpolação de segunda ordem do ponto principal $Tp^D = M^D$ e seus pontos secundários definidos nesta ordem por

 $T_{p-1}{}^D$ e $T_{p+1}{}^D$; para interpolação de quarta ordem foram utilizados o ponto principal $T_p{}^D = M^D$ e seus pontos secundários nesta ordem por $T_{p-2}{}^D$, $T_{p-1}{}^D$, $T_{p+1}{}^D$ e $T_{p+2}{}^D$; para interpolação de décima ordem foram utilizados o ponto principal $T_p{}^D = M^D$ e seus pontos secundários $T_{p-5}{}^D$, $T_{p-4}{}^D$, $T_{p-3}{}^D$, $T_{p-2}{}^D$, $T_{p-1}{}^D$, $T_{p+1}{}^D$, $T_{p+2}{}^D$, $T_{p+4}{}^D$, $T_{p+5}{}^D$.

Foram desenvolvidos três programas computacionais que fazem interpolação de segunda, quarta e décima ordens nos pontos discretos da solução numérica.

Para interpolação de segunda ordem foi utilizada interpolação polinomial quadrática, definida pelas Eqs. (3.39), (3.40), (3.41) e (3.42).

Para interpolação de quarta e décima ordens foi utilizada interpolação polinomial de Newton, definida pelas Eqs. (3.43), (3.44), (3.45), (3.46) e (3.47).

Para a interpolação de quarta ordem são necessários 5 pontos; portanto, h inicia-se $h = \frac{1}{8}$ sendo as malhas anteriores a h=0,125 descartadas. Para a interpolação de décima ordem são necessários 11 pontos; sendo descartadas as malhas anteriores a h=0,0626.

Para localizar o ponto de temperatura máxima T_{max} e x_{max} da função interpolada w(T) foi seguido o seguinte procedimento:

- 1- Obter os coeficientes dos polinômios:
 - (a) Polinômio de segundo graus \rightarrow Eqs. (3.39), (3.40), (3.41) e (3.42).
 - (b) Polinômio de quarto e décimo graus → Eqs. (3.43), (3.44), (3.45), (3.46) e (3.47) programa numérico disponível no apêndice D.
- 2- Derivar a função interpolada w(T).
- 3- Utilizar o método de Newton Raphson para obter as raízes do polinômio derivado função wⁱ(T). O critério de parada utilizado foi diferença da solução atual e anterior menor que 1.0x10⁻¹⁶.
- 4- Verificar se o zero da função wⁱ(T), que corresponde a T_{max}, está no intervalo desejado, sendo:
 - (a) [p-1,p+1] → interpolação de segundo grau.
 - (b) [p-2,p+2] → interpolação de quarto grau.
 - (c) [p-5,p+5] → interpolação de décimo grau.
- 5- Utilizar T_{max} na função w(T) e calcular x_{max} .

5.4.2 Aplicação de MER por meio de interpolação de segunda ordem

A solução numérica foi interpolada com um polinômio de grau 2. Foi utilizado o procedimento descrito em 5.4.1 para obter T_{max}. Os resultados obtidos com esta interpolação são melhor que os obtidos sem qualquer interpolação. O polinômio interpolado de grau 2 limita a redução do erro numérico devido à ordem do polinômio interpolador. Pode-se observar que nos três primeiros pontos do gráfico Emer coincide com Umer divergindo logo após. Foram utilizados nos cálculos, Eq.(3.1), as ordens verdadeira com sendo 2,4,6...

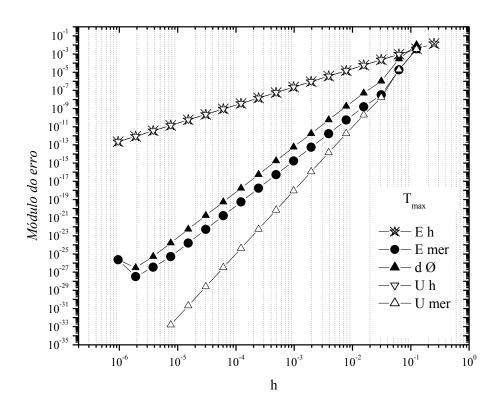


Figura 5.13 - Módulo do erro versus h para a variável $T_{m\acute{a}x}$ para interpolação de ordem 2 com p_L =2 e dp_L =2.

Como os valores de x_{max} são obtidos diretamente a partir da equação interpolada das soluções numéricas em diversas malhas, é previsível que os valores da variável de interesse x_{max} associados com MER também apresentem melhora na redução do erro numérico. É verificado este fato na Fig. 5.14 onde é mostrado que houve realmente a redução do erro numérico com h \rightarrow 0 como esperado.

Foram utilizados nos cálculos, Eq.(3.1), as ordens verdadeira com sendo 2,4,6...

As ordens efetivas da variável T_{max} são mostradas na Fig. 5.15 (a) onde fica claro que as ordens efetivas são 2 e 4 para solução numérica com a extrapolação apresentada. A partir da extrapolação m>1 não se pode afirmar a ordem efetiva da solução.

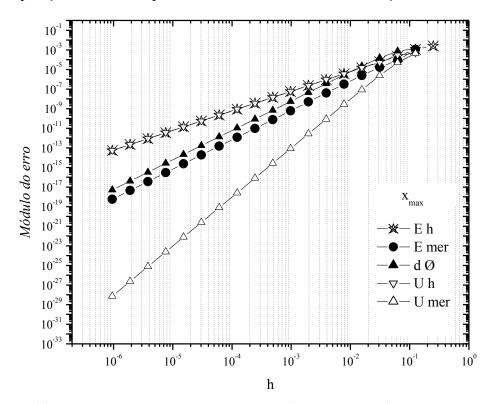


Figura 5.14 - Módulo do erro versus h para a variável $x_{m\acute{a}x}$ para interpolação de ordem 2 com p_L =2 e dp_L =2.

As ordens efetivas da variável x_{max} são mostradas na Fig. 5.15 (b) onde a única ordem claramente definida é p_E =2 para m=0. A partir da extrapolação m> 0 não se pode afirmar a ordem efetiva da solução.

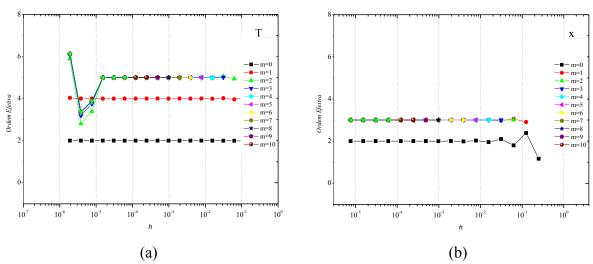


Figura 5.15 - Ordem efetiva (p_E) para T_{max} (a) e x_{max} (b) *versus* o tamanho da Malha (h) e número de extrapolações (m) para interpolação de ordem 2.

5.4.3 Aplicação de MER por meio de interpolação de quarta ordem

Para este item, a solução numérica foi interpolada com um polinômio de quarta ordem. Foi utilizado o procedimento descrito em 5.4.1 para obter T_{max}. Os resultados obtidos para MER associado a esta interpolação são bons. O erro numérico decai significativamente para Emer. O polinômio interpolado de grau 4 ainda limita a redução do erro numérico devido à ordem do polinômio interpolador. Entretanto, mostra-se claramente que a interpolação de ordem 4 na solução numérica sem extrapolação torna MER mais efetivo do que o apresentado no item 5.4.2.

Na Fig. 5.16, observa-se que a estimativa de MER (Umer) não é efetiva a partir do quarto nó. Isto acontece quando sua estimativa (Umer) do erro de discretização é inferior ao erro de discretização alcançado por Emer. Foram utilizados nos cálculos, Eq.(3.1), as ordens verdadeira com sendo 2,4,6...

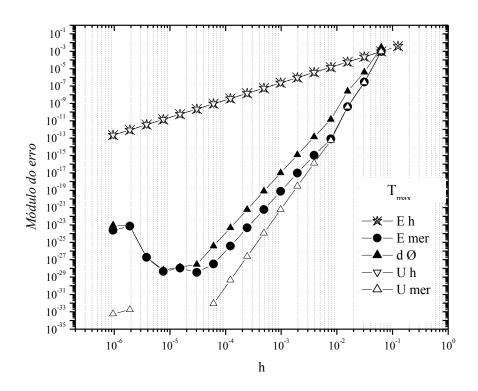


Figura 5.16 - Módulo do erro versus h para a variável $T_{m\acute{a}}x$ para interpolação de ordem 4 com $p_L=2$ e $dp_L=2$.

Como no item 5.4.2, os valores de x_{max} são obtidos diretamente a partir da equação interpolada das soluções numéricas em diversas malhas, é previsível que os valores da

variável de interesse x_{max} associados com MER também apresentem melhora na redução do erro numérico. É verificado este fato na Fig. 5.17 onde é mostrado que houve realmente a redução do erro numérico com h \rightarrow 0 como esperado.

As ordens efetivas da variável T_{max} são mostradas na Fig. 5.18 (a) onde fica claro que as ordens efetivas são 2, 4 e 6 para solução numérica com a extrapolação apresentada. A partir da extrapolação m>2 não se pode afirmar a ordem efetiva da solução.

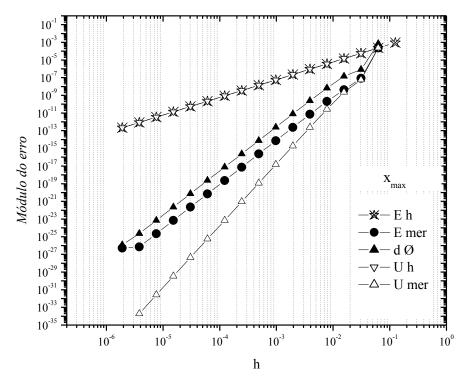


Figura 5.17 - Módulo do erro versus h para a variável $x_{m\acute{a}x}$ para interpolação de ordem 4 com p_L =2 e dp_L =2.

As ordens efetivas da variável x_{max} são mostradas na Fig. 5.18 (b) onde as ordens definidas são $p_E=2$ para m=0 e $p_E=4$ para m=1. A partir da extrapolação m>1 não se pode afirmar a ordem efetiva da solução.

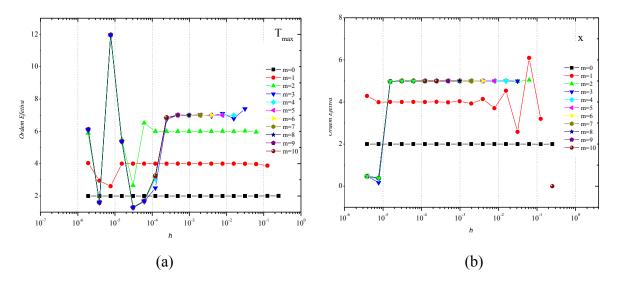


Figura 5.18 - Ordem efetiva (p_E) para T_{max} (a) e x_{max} (b) *versus* o tamanho da Malha (h) e número de extrapolações (m) para interpolação de ordem 4.

5.4.4 Aplicação de MER por meio de interpolação de décima ordem

A Fig. 5.19 mostra o modulo do erro da solução numérica interpolada com um polinômio de décima ordem. Foi utilizado o procedimento descrito em 5.4.1 para obter T_{max}. Os resultados obtidos para MER associado a esta interpolação são ótimos. Emer decai significativamente quando h→0. O polinômio interpolado de grau 10 não limita a redução do erro, mostrando-se que Emer atinge o erro de máquina. Neste ponto, o erro de arredondamento torna-se a principal fonte de erro. Verifica-se que ao atingir o erro de máquina, o erro numérico aumenta com a redução do tamanho da malha. Entretanto, mostra-se claramente que MER associado com a interpolação de ordem 10 é muito mais efetivo.

A estimativa de MER (Umer) é efetiva.

Foram utilizados nos cálculos, Eq.(3.1), as ordens verdadeira com sendo 2,4,6

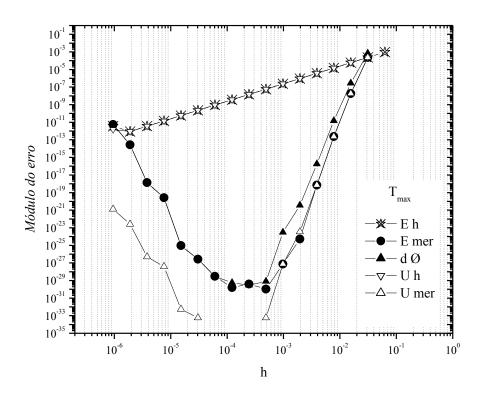


Figura 5.19 - Módulo do erro versus h para a variável $T_{m\acute{a}}x$ para interpolação de ordem 10 com p_L =2 e dp_L =2.

Como no item 5.4.2, os valores de x_{max} são obtidos diretamente a partir da equação interpolada das soluções numéricas em diversas malhas. A redução do erro de discretização da variável de interesse T_{max} associados com MER apresentou a melhor na redução do erro numérico dos experimentos efetuados. É verificado este fato na Fig. 5.20 onde é mostrado que houve realmente a redução do erro numérico com h \rightarrow 0 como esperado.

As ordens efetivas da variável T_{max} são mostradas na Fig. 5.18 (a) onde fica claro que as ordens efetivas são 2, 4, 6 e 8 para solução numérica com a extrapolação apresentada.

Nas Figs. 5.19 e 5.20 observa-se que existem descontinuidades nas curvas discretas de Umer devido ao zeramento dos valores promovidos pelo *software* de MER. O *software* identifica que estes valores são pequenos fazendo-os assumir como zero.

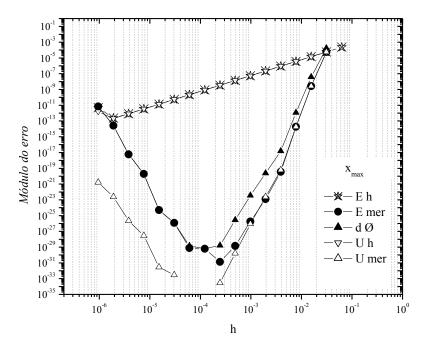


Figura 5.20 - Módulo do erro versus h para a variável $x_{m\acute{a}x}$ para interpolação de ordem 10 com p_L =2 e dp_L =2.

Para a variável $T_{m\acute{a}x}$ e $x_{m\acute{a}x}$, Fig. 5.21, são apresentadas, em função do tamanho da malha (h), as ordens efetivas (p_E) para cada número de extrapolação. As ordens efetivas são respectivamente 2, 4, 6 e 8, tanto para x quanto para T, isto é:

p0=2;

p1=4;

*p*2=6 e

p3=8.

São confirmadas a posteriori as ordens calculadas a priori.

As inclinações das curvas de Emer e Umer aproximam-se uma da outra à medida que o grau do polinômio aumenta.

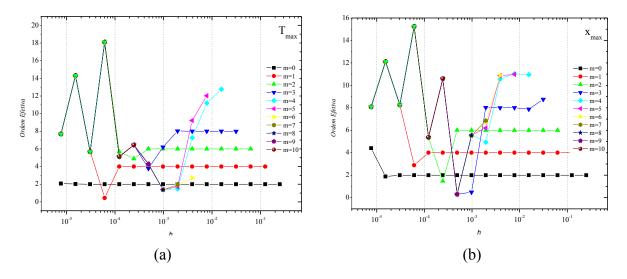


Figura 5.21 - Ordem efetiva (p_E) para T_{max} (a) e x_{max} (b) *versus* o tamanho da Malha (h) e número de extrapolações (m) para interpolação de ordem 10.

5.4.5 Comparação dos resultados

Nas Figs. 5.22 e 5.23 mostram-se a eficiência de MER para as variáveis de interesse T_{max} e x_{max} à medida que a ordem do polinômio interpolador da solução numérica sem extrapolação é aumentada. A interpolação de segundo ordem associada à MER é a mais simples e a que menos é eficiente. Não apresenta ganhos significativos na redução do erro numérico. MER se torna eficiente utilizando interpolação de 10^a ordem.

Portanto, a partir das soluções numéricas obtidas pelo programa Poisson_1Dp_3p1_64BITS associadas à MER com e sem interpolação conforme Figs. 5.22 e 5.23 foram investigadas quatro situações para aumentar a eficiência de MER para as variáveis $T_{m\acute{a}x}$ e $x_{m\acute{a}x}$:

- (0) Aplicação de MER a partir do tratamento da base de dados sem qualquer interpolação obtida pelo programa Poisson 1Dp 3p1 64BITS.
- (i) Aplicação de MER a partir do tratamento da base de dados por meio de interpolação de segunda ordem dos nós da malha obtida pelo programa Poisson 1Dp 3p1 64BITS.
- (ii) Aplicação de MER a partir do tratamento da base de dados por meio de interpolação de quarta ordem dos nós da malha obtida pelo programa Poisson_1Dp_3p1_64BITS

(iii) Aplicação de MER a partir do tratamento da base de dados por meio de interpolação de décima ordem dos nós da malha obtida pelo programa Poisson_1Dp_3p1_64BITS.

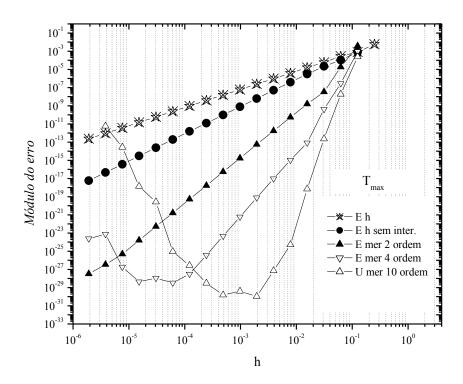


Figura 5.22 - Módulo do erro versus h para a variável $T_{m\acute{a}x}$ para diversas interpolações com p_L =2 e dp_L =2.

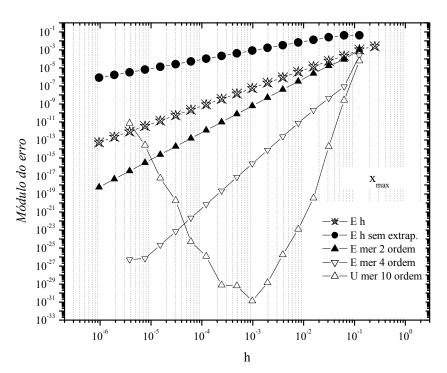


Figura 5.23 - Módulo do erro versus h para a variável $x_{m\acute{a}x}$ para diversas interpolações com p_L =2 e dp_L =2.

Conclui-se que quanto maior a ordem do polinômio interpolado maior é a estabilidade das soluções obtidas com MER e maior à eficiência na redução do erro de discretização.

5.4.6 Redução e estimativa do erro

Com base nos dados da Fig. 5.19, as Tabs. 5.4 e 5.5 foram desenvolvidas para mostrar o ganho de eficiência na utilização de MER. Na Tab. 5.4 foram fixadas três malhas com 64, 256 e 1024 nós. A razão de Eh e Emer mostra a eficiência de MER em relação à solução numérica sem extrapolação. Para uma malha de 256 nós, a eficiência de MER é 1.6248x10¹³ vezes maior que a solução numérica sem extrapolação. À medida que a malha é refinada, a eficiência é cada vez maior. Esse comportamento é demonstrado para as duas variáveis de interesse estudadas neste capítulo.

Tabela 5.3 – Redução do erro para malhas fixas para a variável x_{max} com interpolação de 10 $^{\rm a}$ ordem.

Malha	64	256	1024
h	1.5625e-2	3.9062e-3	9.7656e-4
Eh	1.4385e-5	8.9913e-7	5.6196e-8
Emer	2.5994e-9	3.5338e-20	1.7976e-26
m para <i>Eme</i> r	2	4	6
Eh / Emer	5533.96	1.6248e13	3.1261e18

A Tab.5.5 mostra para um nível fixo de erro para a variável $x_{máx}$ com interpolação de 10^a ordem. Observa-se que o erro numérico na ordem de $-8x10^{-11}$ precisa ser calculado com uma malha de 15 nós. Para MER são necessários somente 3 nós.

Tabela 5.4 – Redução de nós de malha para erros fixos para a variável x_{max} com interpolação de 10^{a} ordem.

Nível do erro	-2e-5	-3e-9	-8e-11
Eh	1.4385e-5	3.5122e-9	8.574e-13
Emer	5.7534e-5	2.59942-9	1.848e-14
m para <i>Emer</i>	2	2	3
Número de nós para <i>Eh</i>	3	9	15
Número de nós para Emer	2	2	4

Para diminuir o efeito do erro de programação sobre os resultados é relevante executar testes iniciais de sua eficiência. Estes testes iniciais com respostas de polinômios conhecidos e com malhas h=1, $\frac{1}{100}$ e $\frac{1}{100000}$ são necessários para analisar a efetividade do polinômio gerado com interpolação. Verifica-se a interceptação dos pontos dos polinômios teste. Devem-se fazer testes na programação executada com um polinômio conhecido para aferir o resultado antes da homologação do programa computacional. Os resultados dos testes de polinômio e os respectivos polinômios de teste estão descritos no apêndice C.

5.5. CONCLUSÃO

Para a equação de Poisson, MER funciona muito bem associado com interpolação de $10^{\rm a}$ ordem e melhora significativamente os resultados.

A ordem do polinômio interpolador afeta os resultados de MER. Polinômio interpolado de grau 10 produz resultados melhores do que os de graus 2 e 4.

A técnica de múltiplas extrapolações de Richardson (MER) tornou-se eficiente para os pontos de máximo, reduzindo o erro de discretização.

Umer mostra-se adequado na estimativa do erro de discretização com MER.

Mostra-se que o aumento de eficiência de MER está relacionado com o aumento da ordem do polinômio interpolado, podendo se concluir:

6 CONCLUSÕES E RECOMENDAÇÕES

Neste capítulo apresentam-se as conclusões referentes ao desenvolvimento da tese a respeito de MÚLTIPLAS EXTRAPOLAÇÕES DE RICHARDSON PARA REDUZIR E ESTIMAR O ERRO DE DISCRETIZAÇÃO EM CONDUÇÃO DE CALOR, tendo em vista os objetivos descritos no capítulo 1 e as recomendações para futuros trabalhos relacionados ao tema tratado.

6.1. METODOLOGIA DA PESQUISA

A metodologia apresentada neste trabalho engloba as etapas de revisão bibliográfica, avaliação, aperfeiçoamento e generalização das múltiplas extrapolações de Richardson aplicada à redução do erro de discretização, estudo de dois problemas práticos sendo eles com a equação de Laplace bidimensional e Poisson unidimensional e apresentação de conclusões e recomendações.

O capítulo 2 apresenta uma revisão bibliográfica sobre a base teórica das soluções de problema em engenharia e os fundamentos desta tese.

O capítulo 3 concentra-se basicamente no desenvolvimento da metodologia para aplicação do MER e o desenvolvimento do erro de posição.

Nos capítulos 4 e 5 foram desenvolvidos os estudos de caso para testar a metodologia desenvolvida. Para isso foi preciso avaliar, aperfeiçoar e generalizar o uso de MER.

6.2. CONCLUSÃO GERAL

Comprovou-se que o uso de múltiplas extrapolações de Richardson para estimar e reduzir o erro de discretização em transferência de calor computacional durante uma aplicação numérica é eficiente. Foi mostrado nesta tese que este erro numérico é reduzido e a acurácia dos cálculos é aumentada quando o parâmetro de malha *h* tende a um valor pequeno com a aplicação de MER.

Portanto, MER é uma ferramenta capaz de melhorar a acurácia das soluções numéricas diminuindo o seu custo computacional. Esta ferramenta foi empregada eficientemente na redução do erro de discretização nas equações de Laplace bidimensional e Poisson unidimensional. O método foi avaliado, aperfeiçoado e generalizado para o uso em problemas de condução de calor. Foram resolvidos, por meio do entendimento do erro de posição, os problemas de convergência de MER em situações onde as soluções apresentam extremos locais e/ou globais. Os resultados obtidos para variáveis principais e secundárias como a

temperatura no centro do domínio, média do campo de temperaturas, taxa de transferência de calor em dois contornos e norma do erro de discretização foram todos satisfatórios e que aprovam o método.

Foi definido nesta tese o erro de posição que associado à interpolação e extrapolação de Richardson resulta em resultados numéricos extremamente acurados para problemas de máximos e mínimos. Mostra-se, portanto, que MER reduz significativamente o erro de discretização, o estimador de erro de Richardson funciona para resultados numéricos obtidos com MER. As respostas apresentaram serem mais efetivas quando os cálculos utilizaram precisão quádrupla.

Em relação aos objetivos firmados no capítulo 1 conclui-se que:

O objetivo geral:

O objetivo geral foi cumprido, pois MER foi avaliado, aperfeiçoado e generalizado. Verificou-se que MER é capaz de reduzir e estimar o erro de discretização em condução de calor e que MER é capaz de diminuir o consumo da memória computacional e o tempo de CPU necessários para resolver problemas de *CHT*.

Os objetivos específicos:

MER foi desenvolvido melhorando o entendimento do método numérico para as variáveis de campo que possuem máximos e mínimos e suas coordenadas.

Foi desenvolvido um estimador de erro para soluções obtidas com MER. Todas as respostas de problemas apresentados nesta tese estão associadas à estimativa de seu erro.

A ferramenta MER está disponível nos *softwares* utilizados para o desenvolvimento desta tese bem como sua metodologia detalhada neste trabalho.

Portanto, confirma-se a justificativa deste trabalho, demonstrado pelos resultados desta tese, na qual o tema tem relevância, tendo-se em vista que este material poderá trazer beneficios à sociedade, à indústria e à ciência.

6.3. RECOMENDAÇÕES PARA TRABALHOS FUTUROS

As recomendações para os trabalhos futuros baseiam-se em:

- Aplicar a mesma metodologia para situações de máximo e mínimo bi e tridimensionais.
- Implementar um estimador de erro associado a sistemas 2D e 3D.

- Analisar a relação entre custo *vesus* tempo de processamento para sistemas 2D e
 3D.
- Desenvolver a mesma teoria para outras equações utilizadas na engenharia mecânica.

REFERÊNCIAS

ALLAIRE, G.. Numerical Analysis and Optimization: An introduction to Mathematical Modeling and Numerical Simulation. 1a ed. New York: Oxford University Press, 2007.

ANDERSON, J.D.. **Computational fluid dynamics**: the basics with applications. New York: McGraw Hill, 1995.

ANTON, H; RORRES, C.. Álgebra linear com aplicações. 8. ed. Porto Alegre: Bookman, 2001.

BERTELLE, R.; RUSSO, M. R.; VENTURIN, M.. On the application of the minimum polynomial extrapolation method to incompressible flows with heat transfer. **Calcolo**, v. 48, p. 33-45, 2011.

BASSANEZI, R. C.. Ensino-aprendizagem com modelagem matemática: uma nova estratégia. São Paulo: Contexto, 2002.

BENJAMIN, A. S.; DENNY, V. E.. On the convergence of numerical solutions for 2-D flows in a cavity at large Re. Journal of Computational Physics, v. 33, p. 340-358, 1979.

BOYCE, W. E.. **Equações diferenciais elementares e problemas de valores de contorno**. 6. ed. São Paulo: McGraw Hill do Brasil, 1995.

BREZINSKI, C.; ZAGLIA, M. R.. Extrapolation methods: theory and practice. Netherlands: Elsevier, 1991.

BROADIE, M.;DETEMPLE, J.. Recent Advances in Numerical Methods for Pricing Derivative Securities. Scientific Series. Montreal: Cirano, 1996.

BURDEN, R. L.; FAIRES, J. D.. Análise numérica. São Paulo: Pioneira Thompson, 2003.

BURG, C.; ERWIN, T.. Application of Richardson extrapolation to the numerical solution of partial differential equations. **Wiley InterSciencesics.** Published online 24 July, 2008. Disponível em: www http://onlinelibrary.wiley.com. Acesso em: 23/10/2010.

CAMARGO, D. Q.. **O** método das probabilidades de colisão tridimensional: criticalidade e fluxo neutrônico em um arranjo hexaédrico. 2007, 74 f. Dissertação (Mestrado em Engenharia Mecânica) - Universidade Federal do Rio Grande do Sul, Porto Alegre, RS.

CHAPRA, S. C.; CANALE, R.. Numerical methods for engineers. 3. ed. Singapore: McGraw-Hill, 1998.

COLLINS, G. W.. Fundamental Numerical Methods and Data Analysis. Internet ed. 1998. Disponível em: http://ads.harvard.edu/books, acesso em: 07/12/2011.

DEITEL, H.M.; DEITEL, P. J.; NIETO, T. R.; LIN, T.M.; SADHO, P.. **How to program**. New Jersey. Prentice Hall, 2001.

DICKSTEIN, F.. Esquemas finitas para malhas não-uniformes numéricos em diferenças. In: CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL. **Minicurso.** Universidade Federal do Paraná. Curitiba, 1995.

DRIKAKIS, D.; ILIEV, O. P.; VENTURIN, M.. On the application of the minimum polynomial extrapolation method to incompressible flows with heat transfer. **Journal of Computational Physics**, v. 146, n. 1, p. 301-321, 1998.

ERTURK, E.; CORKE, T. C.; GÖKÇÖL, C.. Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers. International Journal for Numerical Methods in Fluids, v. 48, p. 747-774, 2005.

FORTUNA, A. O.. **Técnicas computacionais para dinâmica dos fluidos**. São Paulo: Editora da Universidade de São Paulo, 2000.

FLEMMING, D.M.; GONÇALVES, M. B.. Cálculo A: função, limite, derivação e integração. 6. ed. São Paulo: Makron Books, 2006.

FERZIGER, J. H.; PERIC, M.. Computational methods for fluid dynamics. 2. ed. Berlin: Springer, 1999.

FRANKE, J.; HELLSTEN, A.; SCHLÜNZEN, H.; CARISSIMO, B.. Best Practice Guideline for the CFD Simulation of Flows in the Urban Environment. COST Office, 2007.

GONÇALVES, M. B.; FLEMMING, D.M.. Cálculo B: função de várias variáveis, integrais múltiplas, integrais curvilíneas e de superfície. 2ª. ed. São Paulo: Makron Books, 2007.

GRASSELLI, M.; PELINOVSKY, D.. **Numerical Mathematics**. USA: Jones end Bartlett, 2008.

HOLMAN, J. P., Transferencia de calor. 8. ed. Espanha: McGraw-Hill, 1998.

HOLMAN, J.P.. *Experimental methods for engineers.* 7. ed. Boston: McGraw-Hill, 2001.

INCROPERA, F. P.; DeWITT, D. P.. **Fundamentos de transferência de calor**. México: Prentice Hall Hispanoamericana, 1999.

ISAACSON, E.; KELLER, H. B.. **Analysis of Numerical Methods.** Dover ed. New York: Dover Publications, 1994.

JACQUES, I., Mathematics for Economics and Business. 5^a ed. Spain: Prentice Hall, 2006.

KLINE, M.. A mathematical thought from ancient to modern times. v. 1. New York: Oxford University Press, 1972.

KREYSZIG, E..Advanced Engineering Mathematics. 8a ed. New York: Wiley, 1999.

LOMAX, H.; PULLIAM, T. H.; ZINGG, D.W.. Fundamentals of computational fluid dynamics. Germany: Springer, 2003.

MAOR, E., E: historia de um numero. Mexico: D. R. Libraria, 2006.

MARCHI, C. H.. Verificação de soluções numéricas unidimensionais em dinâmica dos fluidos. 2001. Tese (Doutorado em Engenharia Mecânica). Universidade Federal de Santa Catarina. Florianópolis, SC.

MARCHI, C. H.; NOVAK, L. A; SANTIAGO, C.D.. Múltiplas extrapolações de Richardson para reduzir e estimar o erro da equação de Laplace 2D. In: 29° CONGRESSO IBERO LATINO-AMERICANO DE MÉTODOS COMPUTACIONAIS EM ENGENHARIA. **Anais...** XXIX CILAMCE. Maceió, 2008.

MARCHI, C. H.; SILVA A. F. C.. Multi-dimensional discretization error estimation for convergent apparent order. **Journal of the Brazilian Society of Mechanical Sciences and Engineering.**, v. 27, n. 4. Rio de Janeiro, 2005.

MARCHI, C. H., SUERO, R., ARAKI, L. K.. The lid-driven square cavity flow: numerical solution with a 1024x1024 grid. **ABCM**, July-September, 2009.

MELO Jr., P.A.. Fronteiras da engenharia química 1. 1ª ed. Rio de Janeiro: E-papers, 2005.

MONTGOMERY, D. C., RUNGER, G. C., *Applied Statistics and Probability for Engineers*. 3^a ed. USA: John Wiley & Sons, 2003.

MOIN, P.. Fundamentals of Engineering Numerical Analysis. New York: Cambridge, 2010.

MUNEM, M. A.; FOULIS, D. J.. Cálculo. 1ª ed. v. 1. Rio de Janeiro: LTC, 1982.

NOVOTNY, A.A; FANCELLO, E.A.. **Um refinamento** *h*, *p* E *hp* adaptativo na análise de **flexão de placas semi-espeças.** Revista Internacional de Métodos Numéricos para Cálculo y Diseño em Ingeniería, v. 14, p. 25-48, 1998.

OBERKAMPF, W. L.; BLOTTNER, F. G.. Issues in computational fluid dynamics code verification and validation. New Mexico: Sandia National Laboratories, 1997.

OZISIK, M. N.: Finite difference methods in heat transfer. Boca Raton: CRC Press, 1994.

PATANKAR, S. V.. Numerical heat transfer and fluid flow. New York: McGraw-Hill, 1980.

PINTO, J; SOARES, D.. Introdução aos métodos discretos de análise numérica de EDO e EDP. São Carlos, SP: SBMAC, 2006.

REDDY, J. N.. An introduction to the finite element method. 3. Ed. Texas: McGraw-Hill, 2006.

RICHARDSON, L. F; GAUNT, J. A.. The deferred approach to the limit. **Philosophical Proceedings of the Royal Society of London,** Series A, v. 226, p. 299-361, 1927.

ROACHE, P. J.. Perspective: a method for uniform reporting grid refinement studies. **ASME Journal of Fluids Engineering**. v. 116, p. 405-413, 1994.

ROY, C. J.. Review of code and solution verification procedures for computational simulation. Journal of Computational Physics, Vol. 205, pp.131-156, 2005.

RUDIN, W.. Principios de análisis matemático. 1200 -1800. 3. ed. Mexico: McGraw-Hill, 1980.

SCHNEIDER, F. A.; MARCHI, C. H.. Sobre a definição da razão de refino de malhas unidimensionais não-uniformes. In: Congresso Ibero Latino-Americano de Métodos Computacionais em Engenharia, 2004, Recife. **XXV CILAMCE**, 2004. p. 1-13

SCHREIBER, R.; KELLER, H. B.. **Driven cavity flows by efficient numerical techniques.** Journal of Computational Physics, v. 49, p. 310-333, 1983.

SIDI, A.. **Pratical extrapolation methods**: theory and applications. New York: Cambridge University Press, 2003.

SHIMANO, K.; YONEZU, S.; ENOMOTO, Y.. Acceleration of unsteady incompressible flow calculation using extrapolation methods. Computational Fluid Dynamics 2006: FOURTH INTERNATIONAL CONFERENCE ON COMPUTATIONAL FLUID DYNAMICS, ICCFD4, **Proceedings...**Ghent, Belgium, Part 3, p. 303-308, Springer, 2006

STERN, F.; COLEMAN, H. W.; PATERSON, E. G.. Comprehensive approach to verification and validation of CFD simulations - Part 1: methodology and procedures. **Journal of Fluids Engineering**, vol. 123, p. 793-802, 2001.

STRUIK, D. J.. A source book in mathematics, 1200 -1800. USA: MIT, 1969.

SUERO, R.. Verificação de soluções numéricas de escoamentos bidimensionais laminares em malhas uniformes. 2006. 188 f. Dissertação (Mestrado em Ciências) - Universidade Federal do Paraná, Curitiba, Paraná.

TANNEHILL, J. C.; ANDERSON, D. A.; PLETCHER, R. H.. *Computational fluid mechanics and heat transfer.* 2. ed. USA Washington: Taylor & Francis, 1997.

TÁBOAS, P. Z.. Cálculo de uma variável real. São Paulo: Editora da Universidade de São Paulo, 2008.

TANTON, J., Encyclopedia of mathematics. USA: Library of Congress, 2005.

VERSTEEG, H; MALALASEKRA, W.. An Introduction to Computational Fluid Dynamics: The Finite Volume Method. England: Longman Cientific & Technical, 2007.

WHITE, F.. Mecânica dos fluidos. 4. ed. Rio de Janeiro: McGraw-Hill, 2002.

APÊNDICE A

VALORES DOS CÁLCULOS ANALÍTICOS DAS VARIÁVEIS DE INTERESSE

VALORES DOS CÁLCULOS ANALÍTICOS DAS VARIÁVEIS DE INTERESSE DA EQUAÇÃO DE LAPLACE 2D

As variáveis de interesse mostrado neste apêndice foram calculadas por meio do Maple com 50 casas decimais. Estes valores são à base da Eq. (1.1).

As variáveis de interesse para este problema são:

(a) Variável primária local, temperatura TC definida no ponto $x = y = \frac{1}{2}$ é

 $T_C = 0,19926840766919334021681127064217158485299165190291$

(b) Variável primária local, a temperatura T_2 definida no ponto

$$x = \frac{1}{2} \quad e \quad y = \frac{3}{4} \cdot e$$

 $T_2 = 0,45268767117792042438520668109180785113614015381703$

(c) Variável secundária global, a temperatura média Tm o valor calculado é

 $T_m = 0.18585392046028583705427884380360863978099867236075$

(d) Variável secundária local, taxa de transferência de calor Q_e o valor calculado é

 $Q_e = 0.91715233566727434637309292144261877536792714860104$

(e) Variável secundária local, taxa de transferência de calor Q_n o valor calculado é

 $Q_n = -2,0074837463946425764031053823896000349249048459918$

VALORES DOS CÁLCULOS ANALÍTICOS DAS VARIÁVEIS DE INTERESSE DA EQUAÇÃO DE POISSON 1D

As variáveis de interesse para este problema são:

(a) Variável primária local, temperatura T_{max}

(b) Variável primária x max é

APÊNDICE B

TABELAS DOS GRÁFICOS DOS CAPITULOS 4 E 5

As tabelas do apêndice B são apresentadas de forma que as figuras aparecem no texto. Estas tabelas são necessárias para o leitor que irá verificar mais profundamente a informações apresentadas nesta tese possa estabelecer uma análise quantitativa. As tabelas que mostram extrapolações de Richardson são identificadas com a existência de m=1,2,3,... Estes valores são representados com um número razoável de algarismos significativos e, por isso, a tabela é extensa e é representada em duas partes. Para o método de Richardson, como explicado no Cap. 2, é observado que há uma redução do número de soluções numéricas à medida que o número de extrapolação aumenta.

Os valores numéricos apresentados neste apêndice são obtidos por meio das equações dos Caps. 2 e 3.

TABELAS DOS GRÁFICOS DOS CAPITULOS 4

Tabela B.1 – Ordem aparente (p_U) de T_C versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (a)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
5,000000000e-1	NA						
2,500000000e-1	NA						
1,250000000e-1	1,80561102509819529	NA	NA	NA	NA	NA	NA
6,250000000e-2	1,94119915674493553	3,59943404963908330	NA	NA	NA	NA	NA
3,125000000e-2	1,98441284068900465	3,87837765537820915	4,70035301118763619	NA	NA	NA	NA
1,562500000e-2	1,99604201066971232	3,96775238917588046	5,64374060757641911	7,39328119155035486	NA	NA	NA
7,812500000e-3	1,99900657632733074	3,99181140987329753	5,46274124630425524	7,81745346687807868	9,34778455378728184	NA	NA
3,906250000e-3	1,99975139705138308	3,99794473118451795	5,83860629739788420	7,95169191296055861	9,80339521617964666	11,3167886285995717	NA
1,953125000e-3	1,99993783379770510	3,99948567193799928	5,95731529215489611	7,98773872792142152	9,94793443822885925	11,7935998075531982	13,2943076988417746
9,765625000e-4	1,99998445748245203	3,99987138600424107	5,98916786067718433	7,99692286178093616	9,98678237313531238	11,9452978932438871	13,7863686348583289
4,882812500e-4	1,99999611431017081	3,99996784450148583	5,99728164195051716	7,99922997166057191	9,99668255745740542	11,9860688020593255	13,9421016414089404
2,441406250e-4	1,99999902857376496	3,99999196100038542	5,99931976104422229	7,99980745486923079	9,99918910785141157	12,0544686561954786	-

Tabela B.1 - Ordem aparente (p_U) de T_C versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (a) (continuação)

Malha h	m=7	m=8	m=9
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	NA	NA	NA
6,250000000e-2	NA	NA	NA
3,125000000e-2	NA	NA	NA
1,562500000e-2	NA	NA	NA
7,812500000e-3	NA	NA	NA
3,906250000e-3	NA	NA	NA
1,953125000e-3	NA	NA	NA
9,765625000e-4	15,2772901253925105	NA	NA
4,882812500e-4	15,7535464845409921	16,4099875156848093	NA
2,441406250e-4	-	-	-

Tabela B.2 – Ordem aparente (p_U) de T₂ versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (b)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
5,000000000e-1	NA						
2,500000000e-1	NA						
1,250000000e-1	NA						
6,250000000e-2	NA						
3,125000000e-2	NA						
1,562500000e-2	1,99477765368005858	NA	NA	NA	NA	NA	NA
7,812500000e-3	1,99868910162793487	3,99087640448343902	NA	NA	NA	NA	NA
3,906250000e-3	1,99967194122287440	3,99771018665772813	5,98866803658852027	NA	NA	NA	NA
1,953125000e-3	1,99991796438460896	3,99942698589984900	5,99715618288063703	7,98741283757060541	NA	NA	NA
9,765625000e-4	1,99997948978804114	3,99985671137030896	5,99928836461925017	7,99684103884794069	9,98656221301316748	NA	NA
4,882812500e-4	1,99999487236524487	3,99996417564764609	5,99982204851576145	7,99920949392466868	9,99662730449553752	11.9859328325610892	NA
2,441406250e-4	1,99999871808620074	3,99999104377471440	5,99995550946287525	7,99980232621991352	9,99907729393392831	11.8502578836183042	8,62570884306446527

 $Tabela\ B.3-\ Ordem\ aparente\ (p_U)\ de\ T_m\ versus\ tamanho\ da\ malha\ (h)\ e\ n\'umero\ de\ extrapolações\ (m)\ Fig.\ 4.3\ (c)$

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
5,000000000e-1	NA						
2,500000000e-1	NA						
1,250000000e-1	1,89718660458196738	NA	NA	NA	NA	NA	NA
6,250000000e-2	1,97497958361860972	3.70281934049543840	NA	NA	NA	NA	NA
3,125000000e-2	1.99378507243910636	3.90724755262891979	5.48721773906463455	NA	NA	NA	NA
1,562500000e-2	1.99844874199503859	3.97522083165470754	5.84572249691572335	7.40618760966479879	NA	NA	NA
7,812500000e-3	1.99961233954353815	3,99369561027411173	5,95917394642418269	7,82105311368216512	9,35334093282260024	NA	NA
3,906250000e-3	1,99990309450475769	3,99841687394932795	5,98963791053715755	7,95262613782933056	9,80506776463263356	11,3201966894186844	NA
1,953125000e-3	1,99997577422722964	3,99960377631598851	5,99739949832366679	7,98797468163698139	9,94837725408486834	11,7946367158002076	13,2964602901440395
9,765625000e-4	1,99999394359437024	3,99990091639788521	5,99934924674902635	7,99698200449671363	9,98689478403964141	11,9455737790542658	13,7870419947722353
4,882812500e-4	1,53193902259079235	3,99997522736869096	5,99983727238279439	7,99924476717895996	9,99671086477551413	11,9861771893403433	13,9437813819773206
2,441406250e-4	1,89718660458196738	3,99999380673398785	5,99995931563803177	7,99981113811392698	9,99913470073485638	11,9913607932572459	-

Tabela B.3 - Ordem aparente (p_U) de T_m versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (c) (continuação)

Malha h	m=7	m=8	m=9
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	NA	NA	NA
6,250000000e-2	NA	NA	NA
3,125000000e-2	NA	NA	NA
1,562500000e-2	NA	NA	NA
7,812500000e-3	NA	NA	NA
3,906250000e-3	NA	NA	NA
1,953125000e-3	NA	NA	NA
9,765625000e-4	15,2787717322315360	NA	NA
4,882812500e-4	15,8166665118868904	17,4708296375147235	NA
2,441406250e-4	-	-	-

Tabela B.4 – Ordem aparente (p_U) de L versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (d)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
5,000000000e-1	NA	NA	NA	NA	NA	NA	NA
2,500000000e-1	NA	NA	NA	NA	NA	NA	NA
1,250000000e-1	-0,26201722736106442	NA	NA	NA	NA	NA	NA
6,250000000e-2	1,34613321610851498	2,46513473155389564	NA	NA	NA	NA	NA
3,125000000e-2	1,73684860574927229	2,86492200613489183	4,65813720324935319	NA	NA	NA	NA
1,562500000e-2	1,88217064697232876	2,97030215357312865	5,10347597330922587	4,11270223293087249	NA	NA	NA
7,812500000e-3	1,94430986188316479	2,99475684492061254	5,48440537050370254	4,83802204320740599	6,95089210371530771	NA	NA
3,906250000e-3	1,97293608871563912	2,99974290937883580	7,34733700121628510	4,99561496969982861	10,1246424933118188	5,98522416235947212	NA
1,953125000e-3	1,98666025948278462	3,00045393821916785	-	5,01826611608083771	-	6,82360799668083348	8,95744084724740057
9,765625000e-4	1,99337780106496168	3,00037125081311311	3,29051437136921339	5,01441250464725486	5,35819178103935133	7,00005514700958329	-
4,882812500e-4	1,99670077063944007	3,00022151214583956	3,74530476369050295	5,00854673355442476	5,76535952970852965	7,02375341217325492	-1,73874841563154146
2,441406250e-4	1,99835334688526929	3,00011970377936474	3,88808771232761091	5,00461108875999491	5,89684611680995710	7,01803898965806612	7,41791933221595280

Tabela B.4 - Ordem aparente (p_U) de L versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (d) (continuação)

Malha h	m=7	m=8	m=9
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	NA	NA	NA
6,250000000e-2	NA	NA	NA
3,125000000e-2	NA	NA	NA
1,562500000e-2	NA	NA	NA
7,812500000e-3	NA	NA	NA
3,906250000e-3	NA	NA	NA
1,953125000e-3	NA	NA	NA
9,765625000e-4	7,91262992863556994	NA	NA
4,882812500e-4	8,81984411614499878	10,9928131999290692	NA
2,441406250e-4	9,00697326695633804	-	9,87169115757828350

Tabela B.5 - Ordem aparente (p_U) de Q_e versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (e)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
5,000000000e-1	NA						
2,500000000e-1	NA						
1,250000000e-1	2,95261531082530456	NA	NA	NA	NA	NA	NA
6,250000000e-2	2,62765453323765662	3,40681247792551076	NA	NA	NA	NA	NA
3,125000000e-2	2,23808663966074430	3,84107769947165045	5,53353056646082774	NA	NA	NA	NA
1,562500000e-2	2,06815774161549553	3,95916201639587959	5,86113909230704342	7,45119890507570316	NA	NA	NA
7,812500000e-3	2,01768113338114186	3,98971134979052985	5,96333970162549682	7,83410750881734010	9,37658128014050189	NA	NA
3,906250000e-3	2,00446232724834622	3,99742271644886833	5,99070044740432121	7,95604170558301418	9,81185480108127664	11,3327580979535543	NA
1,953125000e-3	2,00111824149505157	3,99935535625051523	5,99766647954683975	7,98883900079801676	9,95016178181981751	11,7984808432721415	13,3045750775442952
9,765625000e-4	2,00027972710986578	3,99983881883969852	5,99941607669956177	7,99719875241708441	9,98734703561025903	11,9465985846063437	13,7895971472446287
4,882812500e-4	2,00006994220640507	3,99995970344530151	5,99985398516791520	7,99929899615720969	9,99682433372642787	11,9864360774816202	13,9435447755026101
2,441406250e-4	2,00001748620353397	3,99998992578227564	5,99996349416569788	7,99982470822608401	9,99922510020271659	12,0220533452237294	-

Tabela B.5 - Ordem aparente (p_U) de Q_e versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (e) (continuação)

Malha h	m=7	m=8	m=9
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	NA	NA	NA
6,250000000e-2	NA	NA	NA
3,125000000e-2	NA	NA	NA
1,562500000e-2	NA	NA	NA
7,812500000e-3	NA	NA	NA
3,906250000e-3	NA	NA	NA
1,953125000e-3	NA	NA	NA
9,765625000e-4	6.35297168713470537	NA	NA
4,882812500e-4	7.63810986748570092	9.25551421510294219	NA
2,441406250e-4	7.91014820552893718	9.78760884663601624	7.88149937564022150

Tabela B.6 - Ordem aparente (p_U) de Q_n versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (f)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
5,000000000e-1	NA	NA	NA	NA	NA	NA	NA
2,500000000e-1	NA	NA	NA	NA	NA	NA	NA
1,250000000e-1	0,63298503093937520	NA	NA	NA	NA	NA	NA
6,250000000e-2	0,86215800425382869	2,78745988504271333	NA	NA	NA	NA	NA
3,125000000e-2	0,95774536037119594	2,16373440917325126	0 ,46759552182687098	NA	NA	NA	NA
1,562500000e-2	0,98804910896495673	2,61628265792463117	2,33111031246413692	3,48780283864183679	NA	NA	NA
7,812500000e-3	0,99667521445932584	2,79493624898078345	2,73814522124560222	3,89854809758785479	6,03635046547854550	NA	NA
3,906250000e-3	0,99904796343986348	2,83815096382694645	2,88534054738303607	4,00475462300345901	NR	4,93836750697826414	NA
1,953125000e-3	0,99970775474619516	2,80203428506231059	2,94659884436717282	4,01867943978194164	2,03765927814991888	5,73647451080901873	7,37809114765131633
9,765625000e-4	0,99990060872410028	2,70321526489182087	2,97426694603638028	4,01374440808588146	4,45882295551894049	5,93745057546841568	7,91372865722776165
4,882812500e-4	0,99996208669863983	2,55579264939367051	2,98737362591247564	4,00801709039555379	4,78857847985050805	5,98683699672094559	8,21075969418190321
2,441406250e-4	0,99998400143397848	2,39034648538105714	2,99374663966726963	4,00430045193967112	4,90475055222772403	5,99803283085138593	8,73473557958095861

Tabela B.6 - Ordem aparente (p_U) de Q_n versus tamanho da malha (h) e número de extrapolações (m) Fig. 4.3 (f) (continuação)

Malha h	m=7	m=8	m=9
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	NA	NA	NA
6,250000000e-2	NA	NA	NA
3,125000000e-2	NA	NA	NA
1,562500000e-2	NA	NA	NA
7,812500000e-3	NA	NA	NA
3,906250000e-3	NA	NA	NA
1,953125000e-3	NA	NA	NA
9,765625000e-4	15,2844887142148941	NA	NA
4,882812500e-4	15,7665672670181327	17,2192246618942117	NA
2,441406250e-4	-	-	-

Tabela B.7 – Módulo do erro da solução numérica da Variável Tc versus tamanho da malha h Fig. 4.4 (a)

Malha h	Eh	Emer	dφ	Uh	Umer
5,000000000e-1	5,0731592330806700e-2	NA	NA	NA	NA
2,500000000e-1	1,4119939979125100e-2	1,4119939979125100e-2	3,6611652351681602e-2	1,2203884117227200E-2	1,2203884117227200E-2
1,250000000e-1	3,6468158526342404e-3	1,5577447713729901e-4	1,7602813847605598e-3	3,4910413754969398E-3	1,1735209231736299E-4
6,250000000e-2	9,1961529485814998e-4	8,66706608038114830e-7	3,7555678211898001e-5	9,0906685259203911e-4	5,96121876389466010e-7
3,125000000e-2	2,3040891621176103e-4	1,6071097785896900e-9	2,6897762187005902e-7	2,2973545954879700e-4	1,0548142082011900e-9
1,562500000e-2	5,7633968425341304e-5	8,4973694747247899e-13	5,5144583344102195e-10	5,7591649262139811e-5	5,3904103403112903e-13
7,812500000e-3	1,4410478510140702e-5	1,3877787807814501e-16	3,1083469131942800e-13	1,4407829971724300e-5	8,3266726846886716e-17
3,906250000e-3	3,6027438113650501e-6	1,2434497875801801e-14	1,2212453270876700e-14	3,6025782329251995e-6	NA
1,953125000e-3	9,0069372754575394e-7	1,8457457784393202e-14	3,089195566019500e-14	9,0068336128235092e-7	NA
9,765625000e-4	2,2517389805520897e-7	2,9087843245179101e-14	4,7545301029572291e-14	2,2517327649684800e-7	NA
4,882812500e-4	5,6293531558449001e-8	4,1133763062361999e-14	7,0221606307541202e-14	5,6293455508171801e-8	NA
2,441406250e-4	1,407347044457550e-8	1,2115308756222001e-13	8,0019324499858196e-14	1,4073353704624500e-8	NA
1,220703125e-4	3,5185156455064301e-9	2,0453083671156903e-13	8,3377749149349294e-14	3,5183182756082203e-9	NA

Tabela B.8 – Módulo do erro da solução numérica da Variável T2 versus tamanho da malha h Fig. 4.4 (b)

Malha h	Eh	Emer	$d\phi$	Uh	Umer
5,000000000e-1	NA	NA	NA	NA	NA
2,500000000e-1	NA	NA	NA	NA	NA
1,250000000e-1	NA	NA	NA	NA	NA
6,250000000e-2	1,0930041348349900e-3	1,09300413483499000e-3	NA	NA	NA
3,125000000e-2	2,7404332371727702e-4	1,05638667802221000e-6	8,1896081111770803e-4	2,7298693703925502e-4	2,7298693703925502e-4
1,562500000e-2	6,8560648158910706e-5	4,2539244349271102e-10	9,8996370523352105e-7	6,8494225186121999e-5	6,5997580345200606e-8
7,812500000e-3	1,7143280324094502e-5	4,9182879990894397e-14	4,1869729905386102e-10	1,7139122611586899e-5	6,6459615588598800e-12
3,906250000e-3	4,2860150371581999e-6	1,5043521983670896e-14	6,3948846218409004e-14	4,2857550956454399e-6	2,7755575615628899e-16
1,953125000e-3	1,0715159670104899e-6	3,2029934260435804e-14	4,7017945092875398e-14	1,0714996900307301e-6	5,5511151231257802e-17
9,765625000e-4	2,6787973700370697e-7	2,6478819137310002e-14	5,8508753397745699e-14	2,6787874335409999e-7	NA
4,882812500e-4	6,6969995160537405e-8	2,1316282072803002e-14	4,7795101210112995e-14	6,6969913947723101e-8	NA
2,441406250e-4	1,6742602859665099e-8	1,4521717162097002e-13	1,2390088954816702e-13	1,6742464081786998e-8	NA
1,220703125e-4	4,1860384047964805e-9	5,5055959791161493e-13	4,0534242629064501e-13	4,18552148495621e-9	NA

Tabela B.9 – Módulo do erro da solução numérica da Variável Tm versus tamanho da malha h Fig. 4.4 (c)

Malha h	Eh	Emer	$d\phi$	Uh	Umer
5,000000000e-1	1,6460795397141598e-3	1,646079539714159800e-3	NA	NA	NA
2,500000000e-1	5,0447423720923505e-3	6,17762998288506980e-3	3,3986628323781905e-3	1,1328876107927200e-3	1,1328876107927200e-3
1,250000000e-1	1,6133538071340702e-3	8,90194612099549070e-5	5,7080723640704194e-3	1,1437961883194200e-3	3,8053815760469300e-4
6,250000000e-2	4,2673752740515802e-4	5,93183484776104960e-7	8,7044617135712015e-5	3,9553875990963605e-4	1,3816605894667600e-6
3,125000000e-2	1,0817093725576401e-4	1,17940446298093010e-9	5,89691564384375970e-7	1,0618886338312199e-4	2,3125159287484803e-9
1,562500000e-2	2,7136031478858202e-5	6,5139560412319497e-13	1,1776019326337200e-9	2,7011635258977799e-5	1,1511347430825899e-12
7,812500000e-3	6,7898450396564101e-6	1,9428902930940202e-16	6,5103478164019200e-13	6,7820621464098601e-6	1,6653345369377299e-16
3,906250000e-3	1,6978261743461099e-6	5,9952043329758501e-15	6,1894933622852509e-15	1,6973396217700998e-6	NA
1,953125000e-3	4,2447936124401205e-7	1,3239409568655001e-14	1,9234613901630802e-14	4,2444893769144599e-7	NA
9,765625000e-4	1,0612126705167300e-7	6,10622663543836e00-16	1,262878690511120e-14	1,0611936474003201e-7	NA
4,882812500e-4	2,6530416669112701e-8	1,5570877920367802e-14	1,496025525682400e-14	2,6530283470105302e-8	NA
2,441406250e-4	6,6326461545251902e-9	5,1569859493838502e-14	3,5998981573470701e-14	6,6325901715291702e-9	NA
1,220703125e-4	1,6582539508203098e-9	1,2931322679321503e-13	7,7743367299376599e-14	1,6581307438201500e-9	NA

Tabela B.10 – Módulo do erro da solução numérica da Variável L versus tamanho da malha h Fig. 4.4 (d)

Malha h	Eh	Emer	$d\phi$	Uh	Umer
5,000000000e-1	5,6368435923118600e-3	5,6368435923118600e-3	NA	NA	NA
2,500000000e-1	3,6687497202486201e-3	3,0127184295608701e-3	1,96809387206325040e-3	6,5603129068774997e-4	6,5603129068774997e-4
1,250000000e-1	1,3087016227754399e-3	1,6620470848298599e-4	2,49069950594314970e-3	7,8668269915772404e-4	3,5581421513473498e-4
6,250000000e-2	3,8038873947167503e-4	4,1332708493729100e-6	1,59691855624087020e-4	3,0943762776792301e-4	1,06461237082724000e-5
3,125000000e-2	1,0187229044399500e-4	1,07755721675000000e-7	3,89971777995734920e-6	9,2838816342560114e-5	1,2579734774056002e-7
1,562500000e-2	2,6317670076544804e-5	8,3201219225104903e-10	1,06891050525574990e-7	2,5184873455816804e-5	1,6966833416757999e-9
7,812500000e-3	6,6856297044832198e-6	6,5941324598268201e-12	8,1896948119910301e-10	6,5440134573538696e-6	6,4485785921189200e-12
3,906250000e-3	1,6846798872597801e-6	1,9123303155827099e-14	6,5874227326584907e-12	1,6669832724078099e-6	2,5833030324150998e-14
1,953125000e-3	4,2282863230292100e-7	1,15612526930503990e-14	3,0624625075735096e-14	4,2061708498562096e-7	5,9930773142339104e-17
9,765625000e-4	1,0591445554503700e-7	1,04129158047988980e-14	2,195270934892560e-14	1,0563805891929500e-7	2,1459148923681602e-17
4,882812500e-4	2,6504537072946503e-8	2,3704403849651200e-14	3,4100660806962602e-14	2,6469972824030102e-8	1,6658847487525299e-17
2,441406250e-4	6,6294209144157202e-9	7,7419895533896592e-14	5,370237755053340e-14	6,6250387195102704e-9	1,3114133711973102e-17
1,220703125e-4	1,6578275612652101e-9	9,6575716701715900e-14	1,9153482810743200e-14	1,6571977843835000e-9	2,3383570761551500e-18

Tabela B.11 – Módulo do erro da solução numérica da Variável Qe versus tamanho da malha h Fig. 4.4 (e)

Malha h	Eh	Emer	$d\phi$	Uh	Umer
5,000000000e-1	1,6715233566727400e-1	1,6715233566727400e-1	NA	NA	NA
2,500000000e-1	2,2545554480726700e-2	2,5656705914789201e-2	1,4460678118654799e-1	4,8202260395515902e-2	4,8202260395515902e-2
1,250000000e-1	3,8661560787414100e-3	8,0721699795138812e-4	2,3296395859535499e-2	6,2264661339951300e-3	1,5530930573023302e-3
6,250000000e-2	8,4369962626029693e-4	4,8133544806594503e-6	7,8986608654152402e-4	1,0074854841604099e-3	1,2537556929204099e-5
3,125000000e-2	2,0303780967079202e-4	8,7003024784593208e-9	4,7858859978378004e-6	2,1355393886313099e-4	1,8768180343187202e-8
1,562500000e-2	5,0263150029206103e-5	4,5959902550407597e-12	8,6872146143335305e-9	5,0924886547232303e-5	8,4918738707529006e-12
7,812500000e-3	1,2534715789991300e-5	1,8873791418627701e-15	4,5929926528742702e-12	1,2576144746367903e-5	1,1102230246251601e-15
3,906250000e-3	3,1317361600669003e-6	3,1086244689504402e-14	3,2973623831367099e-14	3,134326543308140e-6	NA
1,953125000e-3	7,8281257953616301e-7	3,4638958368304903e-14	6,5725203057809305e-14	7,8297452688058698e-7	NA
9,765625000e-4	1,9569557263565700e-7	2,8532731732866498e-14	6,3171690101171395e-14	1,9570566900384300e-7	NA
4,882812500e-4	4,8923308626491700e-8	1,6209256159527298e-13	1,9062529332813900e-13	4,892408800305500e-8	NA
2,441406250e-4	1,2230370827204200e-8	6,0629279374779799e-13	4,4420023215252508e-13	1,2230979229421700e-8	NA
1,220703125e-4	3,0573592546545602e-9	2,8499425042127799e-13	3,2129854332652000e-13	3,0576705611906602e-9	NA

Tabela B.12 – Módulo do erro da solução numérica da Variável Qn versus tamanho da malha h Fig. 4.4 (f)

Malha h	Eh	Emer	dφ	Uh	Umer
5,000000000e-1	1,6460795397141598e-3	1,6460795397141598e-3	NA	NA	NA
2,500000000e-1	5,0447423720923505e-3	6,1776299828850698e-3	3,3986628323781905e-3	1,1328876107927200e-3	1,1328876107927200e-3
1,250000000e-1	1,6133538071340704e-3	8,9019461209954907e-5	5,7080723640704194e-3	1,1437961883194200e-3	3,8053815760469300e-4
6,250000000e-2	4,2673752740515802e-4	5,9318348477610496e-7	8,7044617135712015e-5	3,9553875990963605e-4	1,3816605894667600e-6
3,125000000e-2	1,0817093725576401e-4	1,1794044629809301e-9	5,8969156438437597e-7	1,0618886338312199e-4	2,3125159287484803e-9
1,562500000e-2	2,7136031478858202e-5	6,5139560412319497e-13	1,1776019326337200e-9	2,7011635258977799e-5	1,1511347430825899e-12
7,812500000e-3	6,7898450396564101e-6	1,9428902930940202e-16	6,510347816401920e-13	6,7820621464098601e-6	1,6653345369377299e-16
3,906250000e-3	1,6978261743461097e-6	5,9952043329758501e-15	6,1894933622852509e-15	1,6973396217700998e-6	NA
1,953125000e-3	4,2447936124401205e-7	1,3239409568655001e-14	1,9234613901630802e-14	4,2444893769144599e-7	NA
9,765625000e-4	1,0612126705167300e-7	6,1062266354383600e-16	1,262878690511120e-14	1,0611936474003201e-7	NA
4,882812500e-4	2,6530416669112701e-8	1,5570877920367802e-14	1,496025525682400e-14	2,6530283470105302e-8	NA
2,441406250e-4	6,6326461545251902e-9	5,1569859493838502e-14	3,5998981573470701e-14	6,6325901715291702e-9	NA
1,220703125e-4	1,6582539508203098e-9	1,2931322679321503e-13	7,7743367299376599e-14	1,658130743820150e-9	NA

Tabela B.13 – Módulo do erro da solução numérica da Variável Tc versus tamanho da malha h Fig. 4.5 (a)

Malha h	Eh	Emer	$d\phi$	Uh	Umer
5,000000000e-1	5,0731592330806700e-2	NA	NA	NA	NA
2,500000000e-1	1,4119939979125100e-2	1,41199399791251000e-2	3,6611652351681602e-2	1,2203884117227200e-2	1,2203884117227200e-2
1,250000000e-1	3,6468158526343007e-3	1,55774477137371050e-4	1,7602813847605399e-3	3,4910413754969303e-3	1,1735209231737001e-4
6,250000000e-2	9,1961529485812016e-4	8,66706607971024000e-7	3,7555678212030402e-5	9,0906685259206101e-4	5,9612187638143593e-7
3,125000000e-2	2,3040891621178706e-4	1,60710981373757010e-9	2,6897762177585101e-7	2,2973545954877805e-4	1,0548142030425501e-9
1,562500000e-2	5,7633968423666605e-5	8,4728113713245913e-13	5,5144832955788209e-10	5,7591649262706706e-5	5,3905017552090206e-13
7,812500000e-3	1,4410478509770804e-5	1,2081907701471400e-16	3,0811014253454298e-13	1,4407829971298602e-5	7,5240572047507313e-17
3,906250000e-3	3,602743819843010e-6	4,5369311968924087e-21	4,5573968036010303e-17	3,6025782299759197e-6	2,7817840466343301e-21
1,953125000e-3	9,0069371764482001e-7	4,4196445379040903e-26	1,7551029538126996e-21	9,0068336739939597e-7	2,6781154403070703e-26
9,765625000e-4	2,2517391458925799e-7	9,8342797785273198e-32	1,7415192633172400e-26	2,2517326768518702e-7	6,6444583081359604e-32
4,882812500e-4	5,6293508971103802e-8	1,0180947168517001e-31	1,3370768638908402e-31	5,6293468539384801e-8	NA
2,441406250e-4	1,4073379138015303e-8	7,048903596457281e-32	3,1320435720597401e-32	1,4073376611029503e-8	NA

Tabela B.14 – Módulo do erro da solução numérica da Variável T2 versus tamanho da malha h Fig. 4.4 (b)

Malha h	Eh	Emer	$d\phi$	Uh	Umer
5,000000000e-1	NA	NA	NA	NA	NA
2,500000000e-1	NA	NA	NA	NA	NA
1,250000000e-1	NA	NA	NA	NA	NA
6,250000000e-2	1,09300413483482e-3	NA	NA	NA	NA
3,125000000e-2	2,7404332371765405e-4	2,7404332371765405e-4	8,1896081111716798e-4	2,7298693703905601e-4	2,7298693703905601e-4
1,562500000e-2	6,8560648156853907e-5	6,6422969920421608e-8	9,8996370867803792e-7	6,8494225186933511e-5	6,5997580578535896e-8
7,812500000e-3	1,7143280325811905e-5	6,6985008058302402e-12	4,1869084107984603e-10	1,71391226103473000e-5	6,6458863663467598e-12
3,906250000e-3	4,2860150481007898e-6	2,0732082966933397e-16	5,2407118653809404e-14	4,2857550925703596e-6	2,0551811236787998e-16
1,953125000e-3	1,07151594859336000e-6	1,7769281613953198e-21	1,8009403732924600e-18	1,0714996998358099e-6	1,7604500227688000e-21
9,765625000e-4	2,67879748826864010e-7	4,0623345168196003e-27	1,6474076292006799e-23	2,6787873325550002e-7	4,0229734680259305e-27
4,882812500e-4	6,6969984811905199e-8	1,0650392592461401e-31	3,9467552719592901e-29	6,6969921338319704e-8	2,4074124304840404e-33
2,441406250e-4	1,6742499178305000e-8	1,6500404798537602e-31	5,60927096302782e-32	1,6742495211200099e-8	NA

Tabela B.15 – Módulo do erro da solução numérica da Variável Tm versus tamanho da malha h Fig. 4.5 (c)

Malha h	Eh	Emer	$d\phi$	Uh	Umer
5,000000000e-1	1,6460795397141598e-3	NA	NA	NA	NA
2,500000000e-1	5,0447423720923106e-3	5,0447423720923106e-3	3,3986628323781501e-3	1,13288761079272000e-3	1,132887610792720e-3
1,250000000e-1	1,6133538071340498e-3	4,6955761881463300e-4	5,7080723640703900e-3	1,14379618831942000e-3	3,8053815760469300e-4
6,250000000e-2	4,2673752740500704e-4	1,97484407403750970e-6	8,7044617135902902e-5	3,9553875990968202e-4	1,3816605894587801e-6
3,125000000e-2	1,0817093725553500e-4	3,49192010946491000e-9	5,8969156446927398e-7	1,0618886338315701e-4	2,3125159390951900e-9
1,562500000e-2	2,7136031478673301e-5	1,8023866717473997e-12	1,1776017836979701e-9	2,7011635258953899e-5	1,1511258882678098e-12
7,812500000e-3	6,7898450395273096e-6	2,5467631957742299e-16	6,5100610716001605e-13	6,7820621463819995e-6	1,5897585034432601e-16
3,906250000e-3	1,6978261784344199e-6	9,5119211233394796e-21	9,5690957311973298e-17	1,6973396203642997e-6	5,8408690296022288e-21
1,953125000e-3	4,2447935351000101e-7	9,2346123606519495e-26	3,6709597476136398e-21	4,2444894164147297e-7	5,6015255179780303e-26
9,765625000e-4	1,0612126395715000e-7	2,1435600281029904e-31	3,6330654070736398e-26	1,0611936318428298e-7	1,38594733622966e-31
4,882812500e-4	2,6530405088380000e-8	2,5350052892997003e-32	1,0111132208033001e-31	2,6530286289590101e-8	NA
2,441406250e-4	6,6326068407939605e-9	5,3564926578270008e-32	2,821487368527300e-32	6,6325994158620099e-9	NA

Tabela B.16 – Módulo do erro da solução numérica da Variável L versus tamanho da malha h Fig. 4.5 (d)

Malha h	Eh	Emer	dφ	Uh	Umer
5,000000000e-1	5,6368435923118505e-3	NA	NA	NA	NA
2,500000000e-1	3,6687497202486101e-3	3,6687497202486101e-3	1,9680938720632504e-3	6,5603129068774900e-4	6,5603129068774900e-4
1,250000000e-1	1,3087016227754601e-3	5,2201892361774301e-4	2,4906995059431098e-3	7,8668269915771504e-4	3,5581421513473005e-4
6,250000000e-2	3,8038873947166500e-4	6,5128528588759496e-6	1,59691855624137e-4	3,0943762776793103e-4	1,0646123708275802e-5
3,125000000e-2	1,0187229044406401e-4	2,3355306929482598e-7	3,8997177801050099e-6	9,2838816342533714e-5	1,2579734774532299e-7
1,562500000e-2	2,6317670075884003e-5	8,6467231379854306e-10	1,06891049235704e-7	2,5184873456060102e-5	6,9668332120164995e-10
7,812500000e-3	6,6856297044092899e-6	1,3043139611380200e-11	8,18967867791729e-10	6,5440134571582397e-6	6,4485658881238500e-12
3,906250000e-3	1,6846798910769700e-6	1,3238863334115500e-14	6,5813348599222291e-12	1,6669832711107700e-6	2,5809156313420499e-14
1,953125000e-3	4,2282862749978801e-7	5,2067374423959499e-17	1,25182256048811e-14	4,2061708785905895e-7	2,4497506076088202e-17
9,765625000e-4	1,0591445885175399e-7	1,35213484299488000e-20	2,75563469994413e-17	1,0563805621601098e-7	2,6936800585964100e-20
4,882812500e-4	2,6504525448572098e-8	1,43122265592252000e-23	1,3401139929456099e-20	2,6469977801060801e-8	6,5467219977801991e-24
2,441406250e-4	6,6293700738586399e-9	9,3909673869884200e-28	7,7645654647062581e-24	6,6250517915711410e-9	1,8961087825900502e-27

Tabela B.17 – Módulo do erro da solução numérica da Variável Qe versus tamanho da malha h Fig. 4.5 (e)

Malha h	Eh	Emer	dφ	Uh	Umer
1,250000000e-1	3,8661560787415701e-3	2,3603100552535101e-3	2,3296395859535499e-2	6,2264661339950806e-3	1,5530930573023701e-3
6,250000000e-2	8,4369962626068204e-4	1,73509114093542020e-5	7,8986608654179203e-4	1,0074854841603000e-3	1,2537556929234799e-5
3,125000000e-2	2,0303780967093101e-4	2,74684827723718990e-8	4,7858859973469999e-6	2,1355393886325001e-4	1,8768180381752903e-8
1,562500000e-2	5,0263150032000505e-5	1,30837411775430990e-11	8,6872186494413815e-9	5,0924886546310006e-5	8,4919048381636195e-12
7,812500000e-3	1,2534715791804601e-5	1,7800812462235600e-15	4,5900562581332394e-12	1,2576144746732002e-5	1,1208928591290000e-15
3,906250000e-3	3,1317361391017502e-6	6,5151004224594802e-20	6,5912323609033895e-16	3,1343265509009400e-6	4,0232145278052798e-20
1,953125000e-3	7,8281259630175803e-7	6,2436048165682004e-25	2,4918234586060296e-20	7,8297451426666306e-7	3,8022788715456804e-25
9,765625000e-4	1,9569555896893001e-7	1,5297661548267797e-30	2,4413106473609704e-25	1,9570567911094299e-7	9,3128342460844807e-31
4,882812500e-4	4,8923415357420803e-8	1,6745960866447003e-31	7,6594233888280414e-31	4,8924047870503106e-8	NA
2,441406250e-4	1,2230824190255201e-8	2,7444501707518100e-32	1,9490411037198803e-31	1,2230863722388601e-8	NA

Tabela B.18 – Módulo do erro da solução numérica da Variável Qn versus tamanho da malha h Fig. 4.5 (f)

Malha h	Eh	Emer	$d\phi$	Uh	Umer
5,000000000e-1	1,6715233566727400e-1	NA	NA	NA	
2,500000000e-1	2,2545554480726801e-2	2,2545554480726801e-2	1,4460678118654799e-1	4,8202260395515797e-2	2,0438279433268003e-1
1,250000000e-1	3,8661560787415701e-3	2,3603100552535101e-3	2,3296395859535499e-2	6,2264661339950806e-3	2,2208523502623503e-2
6,250000000e-2	8,4369962626068204e-4	1,7350911409354202e-5	7,8986608654179203e-4	1,0074854841603000e-3	7,7637498522558301e-4
3,125000000e-2	2,0303780967093101e-4	2,7468482772371899e-8	4,7858859973469999e-6	2,1355393886325001e-4	3,8263381698501297e-6
1,562500000e-2	5,0263150032000505e-5	1,3083741177543099e-11	8,6872186494413815e-9	5,0924886546310006e-5	2,3574345927097901e-7
7,812500000e-3	1,2534715791804601e-5	1,7800812462235600e-15	4,5900562581332394e-12	1,2576144746732002e-5	6,9468842863799195e-10
3,906250000e-3	3,1317361391017502e-6	6,5151004224594802e-20	6,5912323609033895e-16	3,1343265509009400e-6	2,3446846766806900e-12
1,953125000e-3	7,8281259630175803e-7	6,2436048165682004e-25	2,4918234586060296e-20	7,8297451426666306e-7	3,9778648357454399e-15
9,765625000e-4	1,9569555896893001e-7	1,5297661548267797e-30	2,4413106473609704e-25	1,9570567911094299e-7	1,2210603550913901e-18
4,882812500e-4	4,8923415357420803e-8	1,6745960866447003e-31	7,6594233888280414e-31	4,8924047870503106e-8	1,0226497798287198e-21
2,441406250e-4	1,2230824190255201e-8	2,7444501707518100e-32	1,9490411037198803e-31	1,2230863722388601e-8	4,5854269601061394e-26

Tabela B.19 – Erro (E) das variáveis T_C versus tamanho da malha (h) e número de extrapolações (m) em Real*16 Fig. 4.6 (a)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6	<i>m</i> =7	m=8	m=9	m=10	m=11
5,000000000e-1	5,073e-02	NA	NA	NA	NA	NA						
2,500000000e-1	1,412e-02	1,916e-03	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
1,250000000e-1	3,647e-03	1,558e-04	8,422e-06	NA	NA	NA	NA	NA	NA	NA	NA	NA
6,250000000e-2	9,196e-04	1,055e-05	8,667e-07	2,706e-07	NA	NA	NA	NA	NA	NA	NA	NA
3,125000000e-2	2,304e-04	6,735e-07	1,512e-08	1,607e-09	5,523e-10	NA	NA	NA	NA	NA	NA	NA
1,562500000e-2	5,763e-05	4,232e-08	2,433e-10	7,122e-12	4,728e-14	3,082e-13	NA	NA	NA	NA	NA	NA
7,812500000e-3	1,441e-05	2,649e-09	3,830e-12	2,876e-14	9,481e-16	1,208e-16	4,558e-17	NA	NA	NA	NA	NA
3,906250000e-3	3,603e-06	1,656e-10	5,996e-14	1,133e-16	5,990e-20	3,403e-20	4,537e-21	1,755e-21	NA	NA	NA	NA
1,953125000e-3	9,007e-07	1,035e-11	9,373e-16	4,436e-19	9,460e-22	8,630e-24	3,211e-25	4,420e-26	1,742e-26	NA	NA	NA
9,765625000e-4	2,252e-07	6,469e-13	1,465e-17	1,734e-21	9,260e-25	2,127e-27	2,037e-29	7,727e-31	9,834e-32	3,190e-32	NA	NA
4,882812500e-4	5,629e-08	4,043e-14	2,289e-19	6,773e-24	9,047e-28	4,188e-31	1,005e-31	1,018e-31	1,018e-31	1,018e-31	1,018e-31	NA
2,441406250e-4	1,407e-08	2,527e-15	3,576e-21	2,646e-26	8,132e-31	7,037e-32	7,049e-32	7,049e-32	7,049e-32	7,049e-32	7,049e-32	7,049e-32

Tabela B.20 – erro (e) das variáveis T₂ versus tamanho da malha (h) e número de extrapolações (m) em Real*16 Fig. 4.6 (b)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11	m=12
5,000000000e-1	NA	NA	NA	NA	NA								
2,500000000e-1	NA	NA	NA	NA	NA								
1,250000000e-1	NA	NA	NA	NA	NA								
6,250000000e-2	1,093e-03	NA	NA	NA	NA	NA							
3,125000000e-2	2,740e-04	1,056e-06	NA	NA	NA	NA	NA						
1,562500000e-2	6,856e-05	6,642e-08	4,254e-10	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
7,812500000e-3	1,714e-05	4,158e-09	6,699e-12	5,261e-14	NA	NA	NA	NA	NA	NA	NA	NA	NA
3,906250000e-3	4,286e-06	2,600e-10	1,049e-13	2,073e-16	1,803e-18	NA	NA	NA	NA	NA	NA	NA	NA
1,953125000e-3	1,072e-06	1,625e-11	1,639e-15	8,116e-19	1,777e-21	1,648e-23	NA	NA	NA	NA	NA	NA	NA
9,765625000e-4	2,679e-07	1,016e-12	2,562e-17	3,172e-21	1,739e-24	4,062e-27	3,936e-29	NA	NA	NA	NA	NA	NA
4,882812500e-4	6,697e-08	6,347e-14	4,003e-19	1,239e-23	1,699e-27	8,853e-31	1,065e-31	1,089e-31	NA	NA	NA	NA	NA
2,441406250e-4	1,674e-08	3,967e-15	6,255e-21	4,841e-26	1,495e-30	1,648e-31	1,650e-31	1,650e-31	1,650e-31	NA	NA	NA	NA

Tabela B.21 – erro (e) das variáveis T_m versus tamanho da malha (h) e número de extrapolações (m) em Real*16 Fig. 4.6 (c)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11	m=12
5,000000000e-1	1,646e-03	NA	NA										
2,500000000e-1	5,045e-03	6,178e-03	NA	NA									
1,250000000e-1	1,613e-03	4,696e-04	8,902e-05	NA	NA								
6,250000000e-2	4,267e-04	3,120e-05	1,975e-06	5,932e-07	NA	NA							
3,125000000e-2	1,082e-04	1,982e-06	3,429e-08	3,492e-09	1,179e-09	NA	NA						
1,562500000e-2	2,714e-05	1,244e-07	5,510e-10	1,544e-11	1,802e-12	6,513e-13	NA	NA	NA	NA	NA	NA	NA
7,812500000e-3	6,790e-06	7,783e-09	8,671e-12	6,230e-14	2,015e-15	2,547e-16	9,570e-17	NA	NA	NA	NA	NA	NA
3,906250000e-3	1,698e-06	4,866e-10	1,357e-13	2,454e-16	2,039e-18	7,169e-20	9,512e-21	3,671e-21	NA	NA	NA	NA	NA
1,953125000e-3	4,245e-07	3,041e-11	2,122e-15	9,606e-19	2,009e-21	1,817e-23	6,729e-25	9,235e-26	3,633e-26	NA	NA	NA	NA
9,765625000e-4	1,061e-07	1,901e-12	3,316e-17	3,754e-21	1,967e-24	4,480e-27	4,269e-29	1,623e-30	2,144e-31	7,576e-32	NA	NA	NA
4,882812500e-4	2,653e-08	1,188e-13	5,181e-19	1,467e-23	1,922e-27	1,071e-30	2,273e-32	2,533e-32	2,535e-32	2,535e-32	2,535e-32	NA	NA
2,441406250e-4	6,633e-09	7,425e-15	8,095e-21	5,729e-26	1,823e-30	5,330e-32	5,356e-32	5,356e-32	5,356e-32	5,356e-32	5,356e-32	5,356e-32	NA

Tabela B.22 – erro (e) das variáveis L versus tamanho da malha (h) e número de extrapolações (m) em Real*16 Fig. 4.6 (d)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11	m=12
5,000000000e-1	5,637e-03	NA	NA										
2,500000000e-1	3,669e-03	3,013e-03	NA	NA									
1,250000000e-1	1,309e-03	5,220e-04	1,662e-04	NA	NA								
6,250000000e-2	3,804e-04	7,095e-05	6,513e-06	4,133e-06	NA	NA							
3,125000000e-2	1,019e-04	9,033e-06	1,881e-07	2,336e-07	1,078e-07	NA	NA						
1,562500000e-2	2,632e-05	1,133e-06	4,128e-09	8,136e-09	8,647e-10	8,320e-10	NA	NA	NA	NA	NA	NA	NA
7,812500000e-3	6,686e-06	1,416e-07	1,905e-11	2,549e-10	6,712e-13	1,304e-11	6,595e-12	NA	NA	NA	NA	NA	NA
3,906250000e-3	1,685e-06	1,770e-08	6,184e-12	7,866e-12	1,028e-13	1,150e-13	1,324e-14	1,257e-14	NA	NA	NA	NA	NA
1,953125000e-3	4,228e-07	2,212e-09	6,147e-13	2,434e-13	2,490e-15	8,986e-16	1,497e-19	5,207e-17	2,757e-17	NA	NA	NA	NA
9,765625000e-4	1,059e-07	2,764e-10	4,551e-14	7,562e-15	4,570e-17	6,906e-18	1,153e-19	1,152e-19	1,352e-20	1,342e-20	NA	NA	NA
4,882812500e-4	2,650e-08	3,455e-11	3,065e-15	2,356e-16	7,666e-19	5,328e-20	6,735e-22	2,239e-22	1,094e-24	1,431e-23	7,766e-24	NA	NA
2,441406250e-4	6,629e-09	4,318e-12	1,985e-16	7,350e-18	1,238e-20	4,132e-22	3,057e-24	4,283e-25	8,987e-27	7,927e-27	9,391e-28	9,570e-28	NA

Tabela B.23 – erro (e) das variáveis Q_e versus tamanho da malha (h) e número de extrapolações (m) em Real*16 Fig. 4.6 (e)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11	m=12
5,000000000e-1	1,672e-01	NA	NA										
2,500000000e-1	2,255e-02	2,566e-02	NA	NA									
1,250000000e-1	3,866e-03	2,360e-03	8,072e-04	NA	NA								
6,250000000e-2	8,437e-04	1,638e-04	1,735e-05	4,813e-06	NA	NA							
3,125000000e-2	2,030e-04	1,052e-05	2,981e-07	2,747e-08	8,700e-09	NA	NA						
1,562500000e-2	5,026e-05	6,617e-07	4,777e-09	1,203e-10	1,308e-11	4,592e-12	NA	NA	NA	NA	NA	NA	NA
7,812500000e-3	1,253e-05	4,143e-08	7,512e-11	4,845e-13	1,456e-14	1,780e-15	6,592e-16	NA	NA	NA	NA	NA	NA
3,906250000e-3	3,132e-06	2,590e-09	1,176e-12	1,907e-15	1,471e-17	4,997e-19	6,515e-20	2,492e-20	NA	NA	NA	NA	NA
1,953125000e-3	7,828e-07	1,619e-10	1,838e-14	7,465e-18	1,450e-20	1,266e-22	4,601e-24	6,244e-25	2,441e-25	NA	NA	NA	NA
9,765625000e-4	1,957e-07	1,012e-11	2,872e-16	2,918e-20	1,419e-23	3,120e-26	2,919e-28	1,106e-29	1,530e-30	5,985e-31	NA	NA	NA
4,882812500e-4	4,892e-08	6,325e-13	4,487e-18	1,140e-22	1,386e-26	7,468e-30	1,495e-31	1,673e-31	1,675e-31	1,675e-31	1,675e-31	NA	NA
2,441406250e-4	1,223e-08	3,953e-14	7,011e-20	4,452e-25	1,357e-29	2,927e-32	2,744e-32	2,744e-32	2,744e-32	2,744e-32	2,744e-32	2,744e-32	NA

Tabela B.24 – erro (e) das variáveis Q_n versus tamanho da malha (h) e número de extrapolações (m) em Real*16 Fig. 4.6 (f)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11	m=12
5,000000000e-1	1,257e+00	NA	NA										
2,500000000e-1	7,261e-01	1,948e-01	NA	NA									
1,250000000e-1	3,835e-01	4,089e-02	1,041e-02	NA	NA								
6,250000000e-2	1,950e-01	6,542e-03	4,909e-03	4,122e-03	NA	NA							
3,125000000e-2	9,799e-02	9,393e-04	9,281e-04	3,594e-04	1,085e-04	NA	NA						
1,562500000e-2	4,906e-02	1,321e-04	1,370e-04	2,399e-05	1,623e-06	1,826e-06	NA	NA	NA	NA	NA	NA	NA
7,812500000e-3	2,454e-02	1,920e-05	1,843e-05	1,493e-06	6,192e-09	5,874e-08	3,069e-08	NA	NA	NA	NA	NA	NA
3,906250000e-3	1,227e-02	3,011e-06	2,385e-06	9,215e-08	1,258e-09	1,099e-09	1,841e-10	5,606e-11	NA	NA	NA	NA	NA
1,953125000e-3	6,136e-03	5,254e-07	3,031e-07	5,706e-09	5,669e-11	1,793e-11	7,630e-13	6,809e-13	4,637e-13	NA	NA	NA	NA
9,765625000e-4	3,068e-03	1,027e-07	3,819e-08	3,547e-10	2,045e-12	2,826e-13	2,573e-15	3,415e-15	7,583e-16	1,477e-16	NA	NA	NA
4,882812500e-4	1,534e-03	2,208e-08	4,794e-09	2,211e-11	6,820e-14	4,422e-15	6,016e-18	1,419e-17	8,579e-19	6,244e-19	4,807e-19	NA	NA
2,441406250e-4	7,670e-04	5,070e-09	6,004e-10	1,380e-12	2,198e-15	6,908e-17	8,865e-21	5,631e-20	8,635e-22	8,137e-22	2,041e-22	3,060e-23	NA

Tabela B.25 – Ordem efetiva (p_e) versus o tamanho da Malha (h) e número de extrapolações (m). Fig. 4.7

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9	m=10	m=11
5,000000000e-1	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
2,500000000e-1	1,845e0	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
1,250000000e-1	1,953e0	3,6206e0	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
6,250000000e-2	1,987e0	3,884e0	5,469e0	NA	NA	NA	NA	NA	NA	NA	NA	NA
3,125000000e-2	1,996e0	3,969e0	5,840e0	7,395e0	NA	NA	NA	NA	NA	NA	NA	NA
1,562500000e-2	1,999e0	3,992e0	5,957e0	7,818e0	9,349e0	NA	NA	NA	NA	NA	NA	NA
7,812500000e-3	1,999e0	3,998e0	5,989e0	7,951e0	9,800e0	1,131e1	NA	NA	NA	NA	NA	NA
3,906250000e-3	1,999e0	3,999e0	5,997e0	7,987e0	9,949e0	1,179e1	1,329e1	NA	NA	NA	NA	NA
1,953125000e-3	1,999e0	3,999e0	5,999e0	7,996e0	9,987e0	1,194e1	1,378e1	1,527e1	NA	NA	NA	NA
9,765625000e-4	1,999e0	3,999e0	5,999e0	7,999e0	9,996e0	1,198e1	1,394e1	1,580e1	1,743e1	NA	NA	NA
4,882812500e-4	1,999e0	3,999e0	5,999e0	7,999e0	9,999e0	1,231e1	7,662e0	2,924e0	-4,998e-2	-1,674e0	NA	NA
2,441406250e-4	1,999e0	3,999e0	5,999e0	7,999e0	1,0119e1	2,573e0	5,122e-1	5,3005e-1	5,304e-1	5,304e-1	5,304e-1	NA

Tabela B.26 – Ordem Aparente (pu) versus o tamanho da Malha (h) e número de extrapolações (m). Fig. 4.8 (a)

Malha h	m=0	m=1	m=2
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	1,80561102509820000e0	NA	NA
6,250000000e-2	1,94119915674494000e0	NA	NA
3,125000000e-2	1,9844128406890000e0	3,7466019264687103e0	NA
1,562500000e-2	1,9960420106697101e0	3,9244478348690199e0	NA
7,812500000e-3	1,9990065763273299e0	3,9801000821079198e0	5,7353871104177294e0
3,906250000e-3	1,9997513970513801e0	3,9949560053352497e0	5,9256823468083697e0
1,953125000e-3	1,9999378337977101e0	3,9987345863306198e0	5,9806531025192298e0
9,765625000e-4	1,9999844574824501e0	3,9996833690075997e0	5,9951094110423702e0

Tabela B.27 – Ordem Aparente (p_u) versus o tamanho da Malha (h) e número de extrapolações (m). Fig. 4.8 (b)

Malha h	m=0	m=1	m=2
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	NA	NA	NA
6,250000000e-2	NA	NA	NA
3,125000000e-2	NA	NA	NA
1,562500000e-2	1,9947776536800601e0	NA	NA
7,812500000e-3	1,9986891016279298e0	NA	NA
3,906250000e-3	1,9996719412228701e0	3,9954471628033401e0	NA
1,953125000e-3	1,9999179643846099e0	3,9988581193331099e0	NA
9,765625000e-4	1,9999794897880401e0	3,9997142989963801e0	5,995499999999998e0

Tabela B.28 – Ordem Aparente (p_u) versus o tamanho da Malha (h) e número de extrapolações (m). Fig. 4.8 (c)

Malha h	m=0	m=1	m=2
5,000000000e-1	1,5319390225907901e0	NA	NA
2,500000000e-1	1,8971866045819699e0	NA	NA
1,250000000e-1	1,9749795836186101e0	4,5422942221705203e0	NA
6,250000000e-2	1,9937850724391102e0	4,1058183698912503e0	NA
3,125000000e-2	1,9984487419950399e0	4,0250656172096297e0	6,2586630980441100e0
1,562500000e-2	1,9996123395435399e0	4,0061847356884002e0	6,0656374570076501e0
7,812500000e-3	1,99990309450476e0	4,0015411540510399e0	6,0164954240078696e0
3,906250000e-3	1,9999757742272299e0	4,0003849752871998e0	6,0041296645976301e0
1,953125000e-3	1,9999939435943701e0	4,0000962242629496e0	6,0010327860623702e0
9,765625000e-4	1,5319390225907901e0	NA	NA

Tabela B.29 - Ordem Aparente (pu) versus o tamanho da Malha (h) e número de extrapolações (m). Fig. 4.8 (d)

Malha h	m=0	m=1	m=2
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	9,8212328456596498e-1	NA	NA
6,250000000e-2	1,8300043439576799e0	NA	NA
3,125000000e-2	1,9618048819342198e0	5,6156561262768099e0	NA
1,562500000e-2	1,9907190838404398e0	4,2894670885647104e0	NA
7,812500000e-3	1,9976965432567499e0	4,0718833382809603e0	6,1155853392177297e0
3,906250000e-3	1,9994251847601898e0	4,0180293346855196e0	5,9357070956067801e0
1,953125000e-3	1,9998563617605298e0	4,0045125044678098e0	5,9743522306389796e0
9,765625000e-4	1,9999640945384702e0	4,0011284736720398e0	5,9928905054641204e0

Tabela B.30 – Ordem Aparente (p_u) versus o tamanho da Malha (h) e número de extrapolações (m). Fig. 4.8 (e)

Malha h	m=0	m=1	m=2
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	9,8212328456596498e-1	NA	NA
6,250000000e-2	1,8300043439576799e0	NA	NA
3,125000000e-2	1,9618048819342198e0	5,6156561262768099e0	NA
1,562500000e-2	1,9907190838404398e0	4,2894670885647104e0	NA
7,812500000e-3	1,9976965432567499e0	4,0718833382809603e0	6,1155853392177297e0
3,906250000e-3	1,9994251847601898e0	4,0180293346855196e0	5,9357070956067801e0
1,953125000e-3	1,9998563617605298e0	4,0045125044678098e0	5,9743522306389796e0
9,765625000e-4	1,9999640945384702e0	4,0011284736720398e0	5,9928905054641204e0

Tabela B.31 - Ordem Aparente (pu) versus o tamanho da Malha (h) e número de extrapolações (m). Fig. 4.8 (f)

Malha h	m=0	m=1	m=2
5,000000000e-1	NA	NA	NA
2,500000000e-1	NA	NA	NA
1,250000000e-1	1,1839083258226100e0	NA	NA
6,250000000e-2	1,5893531816288800e0	NA	NA
3,125000000e-2	1,8085569971861399e0	3,1442280949089101e0	NA
1,562500000e-2	1,9100307615332701e0	3,2176054016115998e0	NA
7,812500000e-3	1,9567501875619899e0	3,1585731795457002e0	NA
3,906250000e-3	1,9788457818291700e0	3,0956308424980601e0	2,99000000000000002e0
1,953125000e-3	1,9895451527424100e0	3,0526945732587198e0	3,5806649288768897e0
9,765625000e-4	1,99480371265394000e0	3,0276975700191602e0	3,7899601536872396e0

 $Tabela\ B.32-\ EEfeito\ da\ utilização\ da\ ordem\ errada\ de\ p_m.\ Fig.\ 4.9$

Malha h	Eh	Emer	dφ	Uh	Umer
2,5200e-1	3,6687497202486201e-3	3,6687497202486201e-3	NA	NA	NA
1,2500e-1	1,3087016227754399e-3	5,22018923617721e-4	2,3600480974731702e-3	7,8668269915772404e-4	7,8668269915772404e-4
6,2500e-2	3,8038873947167503e-4	4,0879924242821002e-5	4,5106781191396909e-4	3,0943762776792301e-4	3,0071187460931303e-5
3,1250e-2	1,0187229044399500e-4	4,3346110763884992e-6	3,5974292648207003e-5	9,2838816342560114e-5	5,7102051822550792e-7
1,5625e-2	2,6317670076544804e-5	5,2294870785648602e-7	3,7967730624049395e-6	2,5184873455816804e-5	1,4889306127078102e-8
7,8125e-3	6,6856297044832198e-6	6,48228143233466070e-8	4,5767850496523595e-7	6,5440134573538696e-6	4,4738856790345204e-10
3,9062e-3	1,6846798872597801e-6	8,08606773308826060e-9	5,6722894845485303e-8	1,6669832724078099e-6	1,38517447730118e-11
1,9531e-3	4,2282863230292100e-7	1,01024567199417000e-9	7,0753901871889906e-9	4,2061708498562096e-7	4,3187390509612201e-13
9,7656e-4	1,0591445554503700e-7	1,2625604801075602e-10	8,8397613537220897e-10	1,056380589192950e-7	1,3488611205802198e-14
4,8828e-4	2,6504537072946503e-8	1,5801043388597998e-11	1,1045458326975400e-10	2,6469972824030102e-8	4,2135240410602303e-16
2,4414e-4	6,6294209144157202e-9	2,0408861369776702e-12	1,376014412891180e-11	6,6250387195102704e-9	1,3122708560567901e-17
1,2207e-4	1,6578275612652101e-9	3,3885796862438999e-13	1,7020277625581398e-12	1,6571977843835e-9	4,0579513748298303e-19

TABELAS DOS GRÁFICOS DOS CAPITULOS 5

Tabela B.33 – Módulo do erro versus h para a variável T_{max} sem interpolação $p_L 2$ e $dp_L = 2$, Fig. 5.6

Malha h	Eh	Emer	$d\phi$	Uh	Umer
2,5000e-01	5,9608e-03	NA	NA	NA	NA
1,2500e-01	7,9863e-04	7,9863e-04	5,1621e-03	1,7207e-03	1,7207e-03
6,2500e-02	2,7707e-04	1,0322e-04	1,0253e-03	1,7385e-04	6,8353e-05
3,1250e-02	5,9107e-05	2,1331e-05	1,9290e-04	7,2655e-05	3,0619e-06
1,5625e-02	1,6014e-05	3,0442e-06	2,7438e-05	1,4364e-05	1,0760e-07
7,8125e-03	3,8470e-06	3,9380e-07	3,5456e-06	4,0558e-06	3,4659e-09
3,9062e-03	9,8120e-07	4,9652e-08	4,4692e-07	9,5525e-07	1,0914e-10
1,9531e-03	2,4286e-07	6,2200e-09	5,5982e-08	2,4611e-07	3,4171e-12
9,7656e-04	6,1019e-08	7,7792e-10	7,0014e-09	6,0613e-08	1,0683e-13
4,8828e-04	1,5217e-08	9,7254e-11	8,7529e-10	1,5268e-08	3,3390e-15
2,4414e-04	3,8089e-09	1,2157e-11	1,0941e-10	3,8026e-09	1,0435e-16
1,2207e-04	9,5164e-10	1,5197e-12	1,3677e-11	9,5244e-10	3,2608e-18
6,1035e-05	2,3798e-10	1,8996e-13	1,7096e-12	2,3789e-10	1,0190e-19
3,0518e-05	5,9487e-11	2,3745e-14	2,1370e-13	5,9499e-11	3,1844e-21
1,5259e-05	1,4873e-11	2,9681e-15	2,6713e-14	1,4871e-11	9,9513e-23

Tabela B.34 – Módulo do erro versus h para a variável x_{max} sem interpolação $p_L 2$ e $dp_L = 2$, Fig. 5.7

Malha h	Eh	Emer	$d\phi$	Uh	Umer
5,0000e-1	1,6667e-01	NA	NA	NA	NA
2,5000e-1	8,3333e-02	8,3333e-02	2,5000e-01	8,3333e-02	8,3333e-02
1,2500e-1	4,1667e-02	8,3333e-02	2,5000e-01	4,1667e-02	1,6667e-02
6,2500e-2	2,0833e-02	5,0000e-02	1,5000e-01	2,0833e-02	2,3810e-03
3,1250e-2	1,0417e-02	2,6190e-02	7,8571e-02	1,0417e-02	3,0812e-04
1,5625e-2	5,2083e-03	1,3249e-02	3,9748e-02	5,2083e-03	3,8854e-05
7,8125e-3	2,6042e-03	6,6441e-03	1,9932e-02	2,6042e-03	4,8675e-06
3,9062e-3	1,3021e-03	3,3245e-03	9,9734e-03	1,3021e-03	6,0877e-07
1,9531e-3	6,5104e-04	1,6625e-03	4,9876e-03	6,5104e-04	7,6106e-08
9,7656e-4	3,2552e-04	8,3131e-04	2,4939e-03	3,2552e-04	9,5136e-09
4,8828e-4	1,6276e-04	4,1566e-04	1,2470e-03	1,6276e-04	1,1892e-09
2,4414e-4	8,1380e-05	2,0783e-04	6,2349e-04	8,1380e-05	1,4865e-10
1,2207e-4	4,0690e-05	1,0392e-04	3,1175e-04	4,0690e-05	1,8581e-11
6,1035e-5	2,0345e-05	5,1958e-05	1,5587e-04	2,0345e-05	2,3227e-12
3,0518e-5	1,0173e-05	2,5979e-05	7,7936e-05	1,0173e-05	2,9034e-13
1,5259e-5	5,0863e-06	1,2989e-05	3,8968e-05	5,0863e-06	3,6292e-14

Tabela B.35 – Módulo do erro versus h para a variável T_{max} sem interpolação $p_L 2$ e $dp_L = 1$, Fig. 5.8

Malha h	Eh	Emer	$d\phi$	Uh	Umer
2,5000e-1	8,3333e-02	8,3333e-02	2,5000e-01	8,3333e-02	8,3333e-02
1,2500e-1	4,1667e-02	5,9524e-02	2,5000e-01	4,1667e-02	3,5714e-02
6,2500e-2	2,0833e-02	3,5714e-02	1,7857e-01	2,0833e-02	1,1905e-02
3,1250e-2	1,0417e-02	3,5714e-02	1,0714e-01	1,0417e-02	3,4562e-03
1,5625e-2	5,2083e-03	1,9585e-02	5,8756e-02	5,2083e-03	9,3263e-04
7,8125e-3	2,6042e-03	1,0259e-02	3,0777e-02	2,6042e-03	2,4234e-04
3,9062e-3	1,3021e-03	5,2506e-03	1,5752e-02	1,3021e-03	6,1772e-05
1,9531e-3	6,5104e-04	2,6562e-03	7,9686e-03	6,5104e-04	1,5594e-05
9,7656e-4	3,2552e-04	1,3359e-03	4,0077e-03	3,2552e-04	3,9176e-06
4,8828e-4	1,6276e-04	6,6991e-04	2,0097e-03	1,6276e-04	9,8179e-07
2,4414e-4	8,1380e-05	3,3545e-04	1,0063e-03	8,1380e-05	2,4575e-07
1,2207e-4	4,0690e-05	1,6785e-04	5,0354e-04	4,0690e-05	6,1474e-08
6,1035e-5	2,0345e-05	8,3954e-05	2,5186e-04	2,0345e-05	1,5373e-08
3,0518e-5	1,0173e-05	4,1984e-05	1,2595e-04	1,0173e-05	3,8439e-09
1,5259e-5	5,0863e-06	2,0994e-05	6,2982e-05	5,0863e-06	9,6105e-10

Tabela B.36 – Módulo do erro versus h para a variável T_{max} sem interpolação $p_L 2$ e $dp_L = 2$, Fig. 5.10

Malha h	Eh	Emer	$d\phi$	Uh	Umer	
3,3333e-01	2,8031e-02	NA	NA	NA	NA	
1,6667e-01	6,8457e-03	6,8457e-03	2,1185e-02	7,0618e-03	7,0618e-03	
8,3333e-02	1,7015e-03	1,3195e-05	2,0288e-04	1,7147e-03	1,3525e-05	
4,1667e-02	4,2477e-04	5,0440e-09	3,2477e-07	4,2559e-04	5,1550e-09	
2,0833e-02	1,0615e-04	4,2520e-13	1,1064e-10	1,0620e-04	4,3388e-13	
1,0417e-02	2,6536e-05	8,3137e-18	8,6702e-15	2,6539e-05	8,4753e-18	
5,2083e-03	6,6338e-06	3,8725e-23	1,6156e-19	6,6340e-06	3,9453e-23	
2,6042e-03	1,6585e-06	4,3867e-29	7,2793e-25	1,6585e-06	4,4432e-29	
1,3021e-03	4,1461e-07	2,3882e-32	5,8856e-31	4,1461e-07	0,0000e+00	
6,5104e-04	1,0365e-07	1,2145e-30	1,1906e-30	1,0365e-07	0,0000e+00	
3,2552e-04	2,5913e-08	4,1985e-31	7,9464e-31	2,5913e-08	0,0000e+00	
1,6276e-04	6,4783e-09	3,2385e-30	2,8186e-30	6,4783e-09	0,0000e+00	

Tabela B.37 – Ordem aparente (p_E) de T_{max} versus tamanho da malha (h) e número de extrapolações (m) Fig. 5.11

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
3,33330e-01	2,80310e-02	NA	NA	NA	NA	NA	NA
1,66670e-01	6,84571e-03	2,16072e-04	NA	NA	NA	NA	NA
8,33330e-02	1,70153e-03	1,31953e-05	3,29812e-07	NA	NA	NA	NA
4,16670e-02	4,24768e-04	8,19979e-07	5,04398e-09	1,11064e-10	NA	NA	NA
2,08330e-02	1,06154e-04	5,11752e-08	7,83936e-11	4,25199e-13	8,67851e-15	NA	NA
1,04170e-02	2,65360e-05	3,19730e-09	1,22327e-12	1,65265e-15	8,31367e-18	1,61598e-19	NA
5,20830e-03	6,63384e-06	1,99813e-10	1,91073e-14	6,44762e-18	8,08013e-21	3,87249e-23	7,27972e-25
2,60420e-03	1,65845e-06	1,24881e-11	2,98527e-16	2,51782e-20	7,88135e-24	9,41046e-27	4,38673e-29
1,30210e-03	4,14612e-07	7,80500e-13	4,66438e-18	9,83445e-23	7,69436e-27	2,27086e-30	2,65778e-32
6,51040e-04	1,03653e-07	4,87812e-14	7,28806e-20	3,84150e-25	8,72658e-30	1,21372e-30	1,21449e-30
3,25520e-04	2,59133e-08	3,04882e-15	1,13876e-21	1,50016e-27	4,28134e-31	4,20045e-31	4,19853e-31
1,62760e-04	6,47831e-09	1,90551e-16	1,77931e-23	2,63785e-30	3,23479e-30	3,23749e-30	3,23826e-30

 $Tabela \ B.38 - \ Ordem \ aparente \ (p_E) \ de \ T_{max} \ versus \ tamanho \ da \ malha \ (h) \ e \ número \ de \ extrapolações \ (m) \ Fig. \ 5.11 \ (continuação)$

Malha h	m=7	m=8	m=9	m=10	m=11
3,33330e-01	,33330e-01 NA		NA	NA	NA
1,66670e-01	1,66670e-01 NA		NA	NA	NA
8,33330e-02	NA	NA	NA	NA	NA
4,16670e-02	4,16670e-02 NA		NA	NA	NA
2,08330e-02	2,08330e-02 NA		NA	NA	NA
1,04170e-02	1,04170e-02 NA		NA	NA	NA
5,20830e-03	NA	NA	NA	NA	NA
2,60420e-03	5,64683e-31	NA	NA	NA	NA
1,30210e-03	2,38815e-32	2,38815e-32	NA	NA	NA
6,51040e-04	1,21449e-30	1,21449e-30	1,21449e-30	NA	NA
3,25520e-04	4,19853e-31	4,19853e-31	4,19853e-31	4,19853e-31	NA
1,62760e-04	3,23845e-30	3,23845e-30	3,23845e-30	3,23845e-30	3,23845e-30

 $Tabela \ B.39 - \ M\'odulo \ do \ erro \ versus \ h \ para \ a \ vari\'avel \ T_{max} \ com \ interpolação \ de \ segunda \ ordem \ com \ p_L 2 \ e \ dp_L = 2, \ Fig. \ 5.13$

Malha h	Eh	Emer	$d\phi$	Uh	Umer
2,5000e-01	1,4075e-02	NA	NA	NA	NA
1,2500e-01	3,7480e-03	3,7480e-03	1,0327e-02	3,4424e-03	3,4424e-03
6,2500e-02	9,5062e-04	1,8156e-05	2,8749e-04	9,3246e-04	1,9166e-05
3,1250e-02	2,3853e-04	3,2955e-08	1,0430e-06	2,3736e-04	1,6555e-08
1,5625e-02	5,9686e-05	1,5325e-09	5,1042e-08	5,9614e-05	2,0016e-10
7,8125e-03	1,4925e-05	5,3966e-11	1,7866e-09	1,4920e-05	1,7465e-12
3,9062e-03	3,7314e-06	1,7394e-12	5,7452e-11	3,7312e-06	1,4030e-14
1,9531e-03	9,3287e-07	5,4783e-14	1,8082e-12	9,3286e-07	1,1037e-16
9,7656e-04	2,3322e-07	1,7153e-15	5,6609e-14	2,3322e-07	8,6379e-19
4,8828e-04	5,8305e-08	5,3630e-17	1,7698e-15	5,8305e-08	6,7513e-21
2,4414e-04	1,4576e-08	1,6761e-18	5,5313e-17	1,4576e-08	5,2750e-23
1,2207e-04	3,6441e-09	5,2381e-20	1,7286e-18	3,6441e-09	4,1212e-25
6,1035e-05	9,1101e-10	1,6369e-21	5,4018e-20	9,1101e-10	3,2197e-27
3,0518e-05	2,2775e-10	5,1154e-23	1,6881e-21	2,2775e-10	2,5154e-29
1,5259e-05	5,6938e-11	1,5984e-24	5,2752e-23	5,6938e-11	1,9654e-31
7,6294e-06	1,4235e-11	4,9999e-26	1,6484e-24	1,4235e-11	1,5407e-33
3,8147e-06	3,5586e-12	3,3963e-27	5,3395e-26	3,5586e-12	NA
1,9073e-06	8,8966e-13	3,2849e-28	3,0678e-27	8,8966e-13	NA

Tabela B.40 – Módulo do erro versus h para a variável x_{max} com interpolação de segunda ordem com $p_L 2$ e $dp_L = 2$, Fig. 5.14

Malha h	Eh	Emer	$d\phi$	Uh	Umer
2,5000e-01	2,4535e-03	NA	NA	NA	NA
1,2500e-01	1,0974e-03	1,0974e-03	1,3561e-03	4,5203e-04	4,5203e-04
6,2500e-02	2,0976e-04	8,6123e-05	7,3151e-04	2,9588e-04	4,8768e-05
3,1250e-02	6,0193e-05	1,6767e-05	1,5166e-04	4,9856e-05	2,4073e-06
1,5625e-02	1,4061e-05	2,3920e-06	2,1567e-05	1,5377e-05	8,4575e-08
7,8125e-03	3,6375e-06	3,0941e-07	2,7860e-06	3,4746e-06	2,7234e-09
3,9062e-03 8,9403e-07		3,9011e-08	3,5114e-07	9,1449e-07	8,5749e-11
1,9531e-03	2,2542e-07	4,8869e-09	4,3984e-08	2,2287e-07	2,6847e-12
9,7656e-04	5,6116e-08	6,1120e-10	5,5008e-09	5,6435e-08	8,3937e-14
4,8828e-04	1,4059e-08	7,6410e-11	6,8769e-10	1,4019e-08	2,6233e-15
2,4414e-04	3,5110e-09	9,5515e-12	8,5964e-11	3,5160e-09	8,1982e-17
1,2207e-04	8,7822e-10	1,1940e-12	1,0746e-11	8,7759e-10	2,5619e-18
6,1035e-05	2,1950e-10	1,4924e-13	1,3432e-12	2,1957e-10	8,0061e-20
3,0518e-05	5,4881e-11	1,8656e-14	1,6790e-13	5,4871e-11	2,5019e-21
1,5259e-05	1,3719e-11	2,3319e-15	2,0988e-14	1,3721e-11	7,8185e-23
7,6294e-06	3,4300e-12	2,9149e-16	2,6234e-15	3,4298e-12	2,4433e-24
3,8147e-06	8,5748e-13	3,6437e-17	3,2793e-16	8,5749e-13	7,6352e-26
1,9073e-06	2,1437e-13	4,5546e-18	4,0991e-17	2,1437e-13	2,3860e-27

Tabela B.41 – Ordem aparente (p_E) de T_{max} versus tamanho da malha (h) e número de extrapolações (m) Fig. 5.15 (a)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
2,50000e-01	1,90895e+00	NA	NA	NA	NA	NA	NA
1,25000e-01	1,97919e+00	4,07334e+00	NA	NA	NA	NA	NA
6,25000e-02	1,99470e+00	3,96125e+00	4,93773e+00	NA	NA	NA	NA
3,12500e-02	1,99870e+00	4,01857e+00	5,05159e+00	5,01373e+00	NA	NA	NA
1,56250e-02	1,99967e+00	3,99055e+00	4,97949e+00	5,00370e+00	5,00482e+00	NA	NA
7,81250e-03	1,99988e+00	4,00460e+00	5,01146e+00	5,00079e+00	5,00111e+00	5,00121e+00	NA
3,90620e-03	1,99998e+00	3,99765e+00	4,99455e+00	5,00023e+00	5,00030e+00	5,00033e+00	5,00033e+00
1,95310e-03	2,00002e+00	4,00123e+00	5,00288e+00	5,00013e+00	5,00015e+00	5,00016e+00	5,00016e+00
9,76560e-04	2,00000e+00	3,99941e+00	4,99862e+00	5,00001e+00	5,00002e+00	5,00002e+00	5,00002e+00
4,88280e-04	2,00000e+00	4,00029e+00	5,00070e+00	5,00000e+00	5,00000e+00	5,00001e+00	5,00001e+00
2,44140e-04	2,00000e+00	3,99985e+00	4,99965e+00	5,00000e+00	5,00000e+00	5,00000e+00	5,00000e+00
1,22070e-04	2,00000e+00	4,00007e+00	5,00017e+00	5,00000e+00	5,00000e+00	5,00000e+00	5,00000e+00
6,10350e-05	2,00005e+00	4,00006e+00	5,00003e+00	5,00012e+00	5,00012e+00	5,00012e+00	5,00012e+00
3,05180e-05	2,00000e+00	4,00002e+00	5,00021e+00	5,00011e+00	5,00010e+00	5,00010e+00	5,00010e+00
1,52590e-05	1,99996e+00	3,99992e+00	4,99736e+00	4,99828e+00	4,99847e+00	4,99852e+00	4,99853e+00
7,62940e-06	2,00000e+00	4,00001e+00	3,38596e+00	3,75180e+00	3,84756e+00	3,87177e+00	3,87784e+00
3,81470e-06	1,99992e+00	3,99988e+00	2,81140e+00	3,19727e+00	3,32390e+00	3,35822e+00	3,36698e+00

Tabela B.42 – Ordem aparente (p_E) de x_{max} versus tamanho da malha (h) e número de extrapolações (m) Fig. 5.15 (b)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
2,500000e-01	1,160727e+00	NA	NA	NA	NA	NA	NA
1,250000e-01	2,387288e+00	2,905696e+00	NA	NA	NA	NA	NA
6,250000e-02	1,801086e+00	3,058635e+00	3,008069e+00	NA	NA	NA	NA
3,125000e-02	2,097859e+00	2,973714e+00	3,002234e+00	3,002884e+00	NA	NA	NA
1,562500e-02	1,950709e+00	3,013878e+00	3,000528e+00	3,000718e+00	3,000784e+00	NA	NA
7,812500e-03	2,024513e+00	2,993192e+00	3,000080e+00	3,000124e+00	3,000140e+00	3,000145e+00	NA
3,906200e-03	1,987702e+00	3,003423e+00	3,000033e+00	3,000045e+00	3,000049e+00	3,000050e+00	3,000050e+00
1,953100e-03	2,006173e+00	2,998345e+00	3,000053e+00	3,000056e+00	3,000057e+00	3,000057e+00	3,000057e+00
9,765600e-04	1,996927e+00	3,000853e+00	3,000002e+00	3,000003e+00	3,000003e+00	3,000003e+00	3,000003e+00
4,882800e-04	2,001536e+00	2,999574e+00	3,000001e+00	3,000001e+00	3,000001e+00	3,000001e+00	3,000001e+00
2,441400e-04	1,999232e+00	3,000213e+00	3,000000e+00	3,000000e+00	3,000000e+00	3,000000e+00	3,000000e+00
1,220700e-04	2,000384e+00	2,999894e+00	3,000000e+00	3,000000e+00	3,000000e+00	3,000000e+00	3,000000e+00
6,103500e-05	1,999855e+00	3,000124e+00	3,000071e+00	3,000071e+00	3,000071e+00	3,000071e+00	3,000071e+00
3,051800e-05	2,000096e+00	2,999973e+00	3,000000e+00	3,000000e+00	3,000000e+00	3,000000e+00	3,000000e+00
1,525900e-05	1,999914e+00	2,999957e+00	2,999943e+00	2,999943e+00	2,999943e+00	2,999943e+00	2,999943e+00
7,629400e-06	2,000024e+00	2,999993e+00	3,000000e+00	3,000000e+00	3,000000e+00	3,000000e+00	3,000000e+00
2,500000e-01	1,160727e+00						

Tabela B.43 – Módulo do erro versus h para a variável T_{max} com interpolação de quarta ordem com $p_L 2$ e $dp_L = 2$, Fig. 5.16

Malha h	Eh	Emer	$d\phi$	Uh	Umer
1,2500e-01	3,8383e-03	NA	NA	NA	NA
6,2500e-02	9,5642e-04	9,5642e-04	2,8819e-03	9,6064e-04	9,6064e-04
3,1250e-02	2,3889e-04	2,8747e-07	3,9283e-06	2,3918e-04	2,6189e-07
1,5625e-02	5,9709e-05	4,1337e-10	2,5173e-08	5,9727e-05	3,9958e-10
7,8125e-03	1,4926e-05	8,2140e-14	1,3875e-11	1,4927e-05	5,4412e-14
3,9062e-03	3,7315e-06	1,0690e-15	1,3762e-13	3,7316e-06	1,3453e-16
1,9531e-03	9,3288e-07	9,4090e-18	1,2129e-15	9,3288e-07	2,9619e-19
9,7656e-04	2,3322e-07	7,5836e-20	9,7810e-18	2,3322e-07	5,9702e-22
4,8828e-04	5,8305e-08	5,9716e-22	7,7030e-20	5,8305e-08	1,1754e-24
2,4414e-04	1,4576e-08	4,6746e-24	6,0301e-22	1,4576e-08	2,3003e-27
1,2207e-04	3,6441e-09	3,6537e-26	4,7134e-24	3,6441e-09	4,4951e-30
6,1035e-05	9,1101e-10	3,1784e-28	3,6859e-26	9,1101e-10	8,7630e-33
3,0518e-05	2,2775e-10	3,3457e-29	2,8439e-28	2,2775e-10	NA
1,5259e-05	5,6938e-11	1,0831e-28	1,4177e-28	5,6938e-11	NA
7,6294e-06	1,4235e-11	4,4034e-29	6,4275e-29	1,4235e-11	NA
3,8147e-06	3,5586e-12	1,8352e-27	1,8792e-27	3,5586e-12	NA
1,9073e-06	8,8966e-13	7,3143e-24	7,3125e-24	8,8966e-13	1,7333e-33
9,5367e-07	2,2242e-13	2,4151e-24	9,7294e-24	2,2242e-13	5,7778e-34

Tabela B.44 – Módulo do erro versus h para a variável x_{max} com interpolação de quarta ordem com $p_L 2$ e $dp_L = 2$, Fig. 5.17

Malha h	Eh	Emer	$d\phi$	Uh	Umer
1,2500e-01	9,2191e-04	NA	NA	NA	NA
6,2500e-02	2,2990e-04	2,2990e-04	6,9201e-04	2,3067e-04	2,3067e-04
3,1250e-02	5,7538e-05	8,3969e-08	8,5484e-07	5,7454e-05	5,6989e-08
1,5625e-02	1,4385e-05	4,2889e-09	1,4525e-07	1,4384e-05	2,3055e-09
7,8125e-03	3,5965e-06	2,0677e-10	6,8012e-09	3,5963e-06	2,6672e-11
3,9062e-03	8,9913e-07	7,3031e-12	2,4075e-10	8,9912e-07	2,3534e-13
1,9531e-03	2,2478e-07	2,3565e-13	7,7741e-12	2,2478e-07	1,8984e-15
9,7656e-04	5,6196e-08	7,4239e-15	2,4497e-13	5,6196e-08	1,4953e-17
4,8828e-04	1,4049e-08	2,3247e-16	7,6713e-15	1,4049e-08	1,1706e-19
2,4414e-04	3,5122e-09	7,2683e-18	2,3985e-16	3,5122e-09	9,1497e-22
1,2207e-04	8,7806e-10	2,2716e-19	7,4964e-18	8,7806e-10	7,1491e-24
6,1035e-05	2,1951e-10	7,0991e-21	2,3427e-19	2,1951e-10	5,5855e-26
3,0518e-05	5,4879e-11	2,2185e-22	7,3210e-21	5,4879e-11	4,3637e-28
1,5259e-05	1,3720e-11	6,9329e-24	2,2878e-22	1,3720e-11	3,4091e-30
7,6294e-06	3,4299e-12	2,1676e-25	7,1496e-24	3,4299e-12	2,6674e-32
3,8147e-06	8,5748e-13	6,8833e-27	2,2365e-25	8,5748e-13	1,9259e-34
1,9073e-06	2,1437e-13	5,2514e-27	1,2135e-26	2,1437e-13	

Tabela B.45 – Ordem aparente (p_E) de T_{max} versus tamanho da malha (h) e número de extrapolações (m) Fig. 5.18 (a)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
2,5000e-01	2,0048e+00	NA	NA	NA	NA	NA	NA
1,2500e-01	2,0013e+00	3,8743e+00	NA	NA	NA	NA	NA
6,2500e-02	2,0003e+00	3,9692e+00	5,9518e+00	NA	NA	NA	NA
3,1250e-02	2,0001e+00	3,9925e+00	6,0182e+00	7,3916e+00	NA	NA	NA
1,5625e-02	2,0000e+00	3,9980e+00	5,9893e+00	6,7866e+00	6,9969e+00	NA	NA
7,8125e-03	2,0000e+00	3,9995e+00	6,0049e+00	7,1015e+00	6,9992e+00	7,00e+00	NA
3,9062e-03	2,0000e+00	3,9999e+00	5,9975e+00	6,9482e+00	6,9999e+00	7,00e+00	7,00e+00
1,9531e-03	2,0000e+00	4,0000e+00	6,0013e+00	7,0257e+00	7,0000e+00	7,00e+00	7,00e+00
9,7656e-04	2,0000e+00	4,0000e+00	5,9994e+00	6,9871e+00	7,0000e+00	7,00e+00	7,00e+00
4,8828e-04	2,0000e+00	4,0000e+00	6,0003e+00	7,0065e+00	7,0001e+00	7,00e+00	7,00e+00
2,4414e-04	2,0000e+00	4,0000e+00	5,9998e+00	6,7347e+00	6,8200e+00	6,84e+00	6,84e+00
1,2207e-04	2,0000e+00	4,0001e+00	5,9980e+00	2,4941e+00	3,0280e+00	3,19e+00	3,23e+00
6,1035e-05	2,0000e+00	4,0000e+00	6,5086e+00	1,6444e+00	1,6783e+00	1,69e+00	1,69e+00
3,0518e-05	2,0000e+00	3,9999e+00	2,6556e+00	1,2777e+00	1,2933e+00	1,30e+00	1,30e+00
1,5259e-05	2,0000e+00	4,0000e+00	5,4977e+00	5,3627e+00	5,3765e+00	5,38e+00	5,38e+00
7,6294e-06	1,9999e+00	2,6014e+00	1,1961e+01	1,1960e+01	1,1960e+01	1,20e+01	1,20e+01
3,8147e-06	2,0001e+00	2,9490e+00	1,6927e+00	1,6216e+00	1,6044e+00	1,60013e+00	1,60e+00

Tabela B.46 – Ordem aparente (p_E) de x_{max} versus tamanho da malha (h) e número de extrapolações (m) Fig. 5.18 (b)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
2,5000e-01	2,0036e+00	NA	NA	NA	NA	NA	NA
1,2500e-01	1,9984e+00	3,1985e+00	NA	NA	NA	NA	NA
6,2500e-02	1,9999e+00	6,0964e+00	5,0385e+00	NA	NA	NA	NA
3,1250e-02	1,9999e+00	2,5838e+00	4,9734e+00	4,9951e+00	NA	NA	NA
1,5625e-02	2,0000e+00	4,5357e+00	5,0115e+00	4,9988e+00	4,9984e+00	NA	NA
7,8125e-03	2,0000e+00	3,7062e+00	4,9937e+00	4,9996e+00	4,9995e+00	4,9995e+00	NA
3,9062e-03	2,0000e+00	4,1391e+00	5,0030e+00	4,9999e+00	4,9999e+00	4,9999e+00	4,9999e+00
1,9531e-03	2,0000e+00	3,9288e+00	4,9985e+00	5,0001e+00	5,0000e+00	5,0000e+00	5,0000e+00
9,7656e-04	2,0000e+00	4,0352e+00	5,0008e+00	5,0000e+00	5,0000e+00	5,0000e+00	5,0000e+00
4,8828e-04	2,0000e+00	3,9823e+00	4,9996e+00	5,0000e+00	5,0000e+00	5,0000e+00	5,0000e+00
2,4414e-04	2,0000e+00	4,0088e+00	5,0002e+00	5,0000e+00	5,0000e+00	5,0000e+00	5,0000e+00
1,2207e-04	2,0000e+00	3,9956e+00	4,9999e+00	5,0000e+00	5,0000e+00	5,0000e+00	5,0000e+00
6,1035e-05	2,0000e+00	4,0023e+00	5,0001e+00	5,0001e+00	5,0001e+00	5,0001e+00	5,0001e+00
3,0518e-05	2,0000e+00	3,9989e+00	4,9987e+00	4,9991e+00	4,9992e+00	4,9993e+00	4,9993e+00
1,5259e-05	2,0000e+00	4,0005e+00	4,9604e+00	4,9730e+00	4,9759e+00	4,9766e+00	4,9767e+00
7,6294e-06	2,0000e+00	3,9882e+00	3,8094e-01	1,7375e-01	3,3437e-01	3,7626e-01	3,8685e-01
3,8147e-06	1,9999e+00	4,2877e+00	4,7421e-01	4,7314e-01	4,7085e-01	4,7009e-01	4,6989e-01

 $Tabela~B.47-~M\'odulo~do~erro~versus~h~para~a~vari\'avel~T_{max}~com~interpola\~c\~ao~de~d\'ecima~ordem~com~p_L 2~e~dp_L=2,~Fig.~5.19$

Malha h	Eh	Emer	$d\phi$	Uh	Umer
6,2500e-02	9,5645e-04	NA	NA	NA	NA
3,1250e-02	2,3889e-04	2,3889e-04	7,1756e-04	2,3919e-04	2,3919e-04
1,5625e-02	5,9709e-05	1,8487e-08	2,7753e-07	5,9727e-05	1,8502e-08
7,8125e-03	1,4926e-05	2,3066e-13	1,4543e-11	1,4927e-05	2,3084e-13
3,9062e-03	3,7315e-06	7,0875e-19	1,7981e-16	3,7316e-06	7,0514e-19
1,9531e-03	9,3288e-07	5,1795e-25	3,6080e-21	9,3288e-07	3,5269e-24
9,7656e-04	2,3322e-07	7,2764e-28	3,0082e-24	2,3322e-07	7,3460e-28
4,8828e-04	5,8305e-08	1,0491e-30	8,0070e-30	5,8305e-08	5,7778e-34
2,4414e-04	1,4576e-08	4,0185e-30	2,9688e-30	1,4576e-08	0,0000e+00
1,2207e-04	3,6441e-09	1,5470e-30	5,5655e-30	3,6441e-09	0,0000e+00
6,1035e-05	9,1101e-10	3,0163e-29	2,8616e-29	9,1101e-10	0,0000e+00
3,0518e-05	2,2775e-10	2,6426e-27	2,6124e-27	2,2775e-10	5,7778e-34
1,5259e-05	5,6938e-11	9,2537e-26	8,9894e-26	5,6938e-11	5,3926e-33
7,6294e-06	1,4235e-11	2,6049e-20	2,6049e-20	1,4235e-11	3,8816e-28
3,8147e-06	3,5586e-12	1,3417e-18	1,3157e-18	3,5586e-12	4,9013e-27
1,9073e-06	8,7094e-13	2,7187e-14	2,7188e-14	8,9590e-13	2,5321e-23
9,5367e-07	3,6664e-12	5,6389e-12	5,6117e-12	1,5124e-12	1,3066e-21

Tabela B.48 – Módulo do erro versus h para a variável x_{max} com interpolação de décima ordem com $p_L 2$ e $dp_L = 2$, Fig. 5.20

Malha h	Eh	Emer	$d\phi$	Uh	Umer
6,2500e-02	2,3001e-04	NA	NA	NA	NA
3,1250e-02	5,7534e-05	5,7534e-05	1,7248e-04	5,7493e-05	5,7493e-05
1,5625e-02	1,4385e-05	2,5995e-09	3,8974e-08	1,4383e-05	2,5983e-09
7,8125e-03	3,5965e-06	1,8483e-14	1,1635e-12	3,5963e-06	1,8468e-14
3,9062e-03	8,9913e-07	3,5340e-20	1,5230e-17	8,9912e-07	5,9726e-20
1,9531e-03	2,2478e-07	1,2337e-23	2,4398e-20	2,2478e-07	2,3850e-23
9,7656e-04	5,6196e-08	1,7977e-26	3,6205e-23	5,6196e-08	8,8413e-27
4,8828e-04	1,4049e-08	1,4094e-29	2,6832e-26	1,4049e-08	1,6378e-30
2,4414e-04	3,5122e-09	1,3559e-31	1,5867e-29	3,5122e-09	2,8889e-34
1,2207e-04	8,7806e-10	6,2558e-30	6,3917e-30	8,7806e-10	NA
6,1035e-05	2,1951e-10	7,7148e-30	1,3971e-29	2,1951e-10	NA
3,0518e-05	5,4879e-11	1,1942e-26	1,1934e-26	5,4879e-11	2,8889e-33
1,5259e-05	1,3720e-11	4,9351e-25	4,8157e-25	1,3720e-11	2,8696e-32
7,6294e-06	3,4299e-12	1,9186e-20	1,9186e-20	3,4299e-12	2,8589e-28
3,8147e-06	8,5748e-13	5,8695e-18	5,8503e-18	8,5748e-13	2,1794e-26
1,9073e-06	2,3225e-13	2,5975e-14	2,5980e-14	2,0841e-13	2,4196e-23
9,5367e-07	4,9282e-12	7,0710e-12	7,0450e-12	1,5653e-12	1,6403e-21

Tabela B.49 – Ordem aparente (p_E) de T_{max} versus tamanho da malha (h) e número de extrapolações (m) Fig. 5.21 (a)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
2,5000e-01	2,0013e+00	NA	NA	NA	NA	NA	NA
1,2500e-01	2,0003e+00	4,0011e+00	NA	NA	NA	NA	NA
6,2500e-02	2,0001e+00	4,0003e+00	6,0011e+00	NA	NA	NA	NA
3,1250e-02	2,0000e+00	4,0000e+00	6,0002e+00	7,9925e+00	NA	NA	NA
1,5625e-02	2,0000e+00	4,0000e+00	6,0001e+00	7,9997e+00	1,2766e+01	NA	NA
7,8125e-03	2,0000e+00	4,0001e+00	6,0001e+00	8,0001e+00	1,1194e+01	1,201390e+01	NA
3,9062e-03	2,0000e+00	4,0000e+00	6,0000e+00	8,0000e+00	7,2614e+00	9,212618e+00	2,7296e+00
1,9531e-03	2,0000e+00	4,0000e+00	6,0000e+00	8,0354e+00	1,4782e+00	1,711827e+00	1,9375e+00
9,7656e-04	2,0000e+00	4,0000e+00	6,0000e+00	6,2184e+00	1,3827e+00	1,378525e+00	1,3775e+00
4,8828e-04	2,0000e+00	4,0000e+00	6,0008e+00	3,7948e+00	4,2902e+00	4,286508e+00	4,2856e+00
2,4414e-04	2,0000e+00	4,0001e+00	4,8886e+00	6,4527e+00	6,4531e+00	6,453145e+00	6,4532e+00
1,2207e-04	2,0000e+00	4,0000e+00	5,6793e+00	5,1301e+00	5,1300e+00	5,130017e+00	5,1300e+00
6,1035e-05	2,0000e+00	4,4375e-01	1,8102e+01	1,8102e+01	1,8102e+01	1,810241e+01	1,8102e+01
3,0518e-05	2,0000e+00	5,7538e+00	5,6873e+00	5,6869e+00	5,6868e+00	5,686737e+00	5,6867e+00
1,5259e-05	2,0306e+00	1,4304e+01	1,4305e+01	1,4306e+01	1,4306e+01	1,430598e+01	1,4306e+01
7,6294e-06	2,0738e+00	7,6972e+00	7,6967e+00	7,6966e+00	7,6966e+00	7,696581e+00	7,6966e+00

Tabela B.50 – Ordem aparente (p_E) de x_{max} versus tamanho da malha (h) e número de extrapolações (m) Fig. 5.21 (b)

Malha h	m=0	m=1	m=2	m=3	m=4	m=5	m=6
2,500e-01	1,999e+00	NA	NA	NA	NA	NA	NA
1,250e-01	2,000e+00	3,999e+00	NA	NA	NA	NA	NA
6,250e-02	2,000e+00	4,000e+00	5,999e+00	NA	NA	NA	NA
3,125e-02	2,000e+00	4,000e+00	6,000e+00	8,755e+00	NA	NA	NA
1,563e-02	2,000e+00	4,000e+00	6,000e+00	7,877e+00	1,095e+01	NA	NA
7,813e-03	2,000e+00	4,000e+00	6,000e+00	8,014e+00	1,103e+01	1,097e+01	NA
3,906e-03	2,000e+00	4,000e+00	6,000e+00	7,998e+00	1,056e+01	1,086e+01	1,089e+01
1,953e-03	2,000e+00	4,000e+00	6,000e+00	8,021e+00	4,933e+00	6,196e+00	6,709e+00
9,766e-04	2,000e+00	4,000e+00	6,000e+00	4,851e-01	5,605e+00	5,563e+00	5,537e+00
4,883e-04	2,000e+00	4,000e+00	6,003e+00	2,884e-01	3,009e-01	3,021e-01	3,023e-01
2,441e-04	2,000e+00	4,000e+00	1,471e+00	1,060e+01	1,060e+01	1,060e+01	1,060e+01
1,221e-04	2,000e+00	4,000e+00	5,378e+00	5,369e+00	5,369e+00	5,369e+00	5,369e+00
6,104e-05	2,000e+00	2,885e+00	1,525e+01	1,525e+01	1,525e+01	1,525e+01	1,525e+01
3,052e-05	2,000e+00	8,270e+00	8,257e+00	8,257e+00	8,257e+00	8,257e+00	8,257e+00
1,526e-05	1,884e+00	1,211e+01	1,211e+01	1,211e+01	1,211e+01	1,211e+01	1,211e+01
7,629e-06	4,407e+00	8,089e+00	8,089e+00	8,089e+00	8,089e+00	8,089e+00	8,089e+00

APÊNDICE C

TESTE DE POLINOMIOS

Para verificar a validade dos métodos e programas de interpolação do Cap. 5, estes foram submetidos aos seguintes procedimentos:

Procedimento 1 - Obtenção dos pares ordenados (x,T) por meio por meio de um polinômio analítico teste.

Caso 1: polinômio teste de grau 2:

$$T(x) = \frac{1}{2} x^2 + 7 x - 3.$$
 (C1)

Caso 2: polinômio teste de grau 1:

$$T(x) = +7x - 3.$$
 (C2)

Caso 3: polinômio teste de grau 0:

$$T(x) = -3. (C3)$$

Para o caso 1, são escolhidos dentro de intervalo de interesse x_1 , x_2 e x_3 e calculados por meio da Eq. (C1) os valores de T_1 , T_2 e T_3 . São obtidos três pares ordenados (x_1 , T_1), (x_2 , T_2) e (x_3 , T_3) que são denominados os dados de entrada.

Para o caso 2, são escolhidos dentro de intervalo de interesse x_1 e x_2 e x_3 e calculados por meio da Eq. (C2) os valores de T_1 , T_2 e T_3 . São obtidos três pares ordenados (x_1 , T_1), (x_2 , T_2) e (x_3 , T_3). Observa-se que os três pares ordenados formam um reta.

Para o caso 3, são escolhidos dentro de intervalo de interesse x_1 e x_2 e x_3 e calculados por meio da Eq. (C3) os valores de T_1 , T_2 e T_3 . São obtidos três pares ordenados (x_1 , T_1), (x_2 , T_2) e (x_3 , T_3). Observa-se que os três pares ordenados resultam em T_1 = T_2 = T_3 .

Procedimento 2 - Obtenção dos coeficientes A, B, C, ... do polinômio interpolado. O programa numérico, por meio dos dados de entrada, calcula para os casos 1, 2 e 3 os valores dos coeficientes A, B e C da equação

$$Ax^2 + Bx + C. (C4)$$

Para os três casos, são observadas três situações distintas:

- a) O teste de interpolação obtido pelos pares ordenados (x,T) do caso 1 devem resultar em coeficientes A, B e C diferentes.
- b) O teste de interpolação obtido pelos pares ordenados (x,T) do caso 2 devem resultar em coeficientes A = 0, B e C diferentes de zero.
- c) O teste de interpolação obtido pelos pares ordenados (x,T) do caso 3 devem resultar em coeficientes A=0, B=0 e C diferente de zero.

Procedimento 3 - Obtenção de x_{max} e T_{max} por meio do polinômio interpolado.

Para todos os polinômios de graus maiores que 1, os resultados obtidos x_{max} e T_{max} por meio do polinômio interpolado devem convergir para o x_{max} e T_{max} obtido analiticamente.

Procedimento 4 - Para **os casos 1, 2 e 3,** foram executados em diferentes passos de x sendo eles h=1, 0,001 e 0,0000001.

Por exemplo, para os casos estudados foram obtidos dados de entrada para diferentes passos de x como exemplificado e esquematizado a seguir:

para h = 1, os pontos de x foram determinados da seguinte maneira:

 x_1 = inteiro qualquer no intervalo de interesse.

 x_2 = inteiro qualquer no intervalo de interesse + h.

 x_3 = inteiro qualquer no intervalo de interesse + 2h.

Os valores de T são obtidos diretamente em T(x)=f(x).

para h =0,001, os pontos de x foram determinados da seguinte maneira:

 x_1 = decimal qualquer no intervalo de interesse.

 x_2 = decimal qualquer no intervalo de interesse.+ h.

 x_3 = decimal qualquer no intervalo de interesse.+ 2h.

Os valores de T são obtidos diretamente em T(x) = f(x).

para h =0,00001, os pontos de x foram determinados da seguinte maneira:

 x_1 = decimal qualquer no intervalo de interesse.

 x_2 = decimal qualquer no intervalo de interesse.+ h.

 x_3 = decimal qualquer no intervalo de interesse.+ 2h.

Os valores de T são obtidos diretamente em T(x)=f(x).

Procedimento 5 - Testar o comportamento dos programas verificando os procedimentos de 1 a 4.

Os procedimentos descritos valem para testar a interpolação polinomial de graus 4 e 10.

Os procedimentos adotados para testes de polinômios são necessário devido os números serem na maioria representados com precisão quádrupla e usarem malhas com $h \rightarrow 0$.

Observou-se que as respostas obtidas para uma interpolação simples de segundo grau podem ser <u>errada</u>s devido a problemas no equacionamento matemático. Nesta tese, foram obtidas inicialmente respostas erradas até mesmo para interpolação de segunda ordem. Estas respostas promoveram custos desnecessários de tempo devido os retrabalhos de programação e entendimento deste tipo de erro.

Entretanto, aprendeu-se que existe um erro pouco considerado nas programações numéricas que é intrínseco do equacionamento matemático. Este erro é difícil de ser detectado devido um determinado programa funcionar bem para malhas grossas e erroneamente para malhas finas. À medida que a malha é refinada, os erros de arredondamento começam a aparecem e interferem substancialmente nos resultados.

Este erro definido como sendo o erro de equacionamento matemático é devido às rotinas prontas do FORTRAN como é o caso da utilização de rotina de potência. Esta rotina é uma composição de multiplicações como, por exemplo: $e^2 = e.e.$ Estes erros são internos do software de programação e muitas vezes não são considerados. Devem-se utilizar multiplicações em vez de potencias devido esta operação ser função primária. Portanto, a forma de minimizar este erro é utilizando de rotinas primárias.

O erro de arredondamento nas rotinas dos *softwares* de programação é inevitável. Este erro fica mais evidente quando utilizados em programas com grande número de cálculos como é o caso do apresentado no apêndice D.

Para as soluções analíticas dos polinômios e para a interpolação numérica foi utilizado um microcomputador com um processador Intel® Xeon Quad Core X5355 com 2,66 GHz e com 16GB memória RAM e sistema operacional Windows® XP 64 bits. O *software* utilizado para a solução analítica foi o Maple® 7.00. Para os cálculos efetuados com esse *software* utilizaram-se precisões de 40 casas decimais.

Para as soluções numéricas, os programas foram elaborados em Fortran® 95, versão 9.1 da Intel®, usando precisão quádrupla (Real*16).

C1. Polinômio numérico na forma: $Ax^2 + Bx + C$

$$T(x) = \frac{1}{2} x^2 + 7 x - 3$$

	xmax	Tmax
Resultado		
ANALITICO	7,000000000000000000000000000000000000	-2,75000000000000000000000000000000000000

Teste 1: os pontos (x,T) são baseados em um polinômio de ordem 2. Utiliza-se h=1.

Conclusão: O polinômio interpolado se comporta como esperado.

Teste 2: os pontos (x,T) são baseados em um polinômio de ordem 2. Utiliza-se h=0,001.

xmax

Гтах

Conclusão: O polinômio interpolado se comporta como esperado.

Teste 3: os pontos (x,T) são baseados em um polinômio de ordem 2. Utiliza-se h=0,000001.

TESTE DE POLINÔMIO h=0,000001

DADOS DE ENTRADA

x1= -7,00001250000000000000000000000000910000E+00

T1= -2,7499999999999218750000000000000001050000E+01

T3= -2,749999999999971874999999999999999270000E+01

DADOS DE SAÍDA

xmax

Tmax

Conclusão: O polinômio interpolado se comporta como esperado.

C2. Polinômio numérico de quarta ordem na forma: $Ax^4 + Bx^3 + Cx^2 + Dx + E$

O polinômio testado é de quarto grau. Os pares ordenados (x,T) são obtidos em polinômio de terceiro e segundo graus com precisão de:

- (a) 1;
- (b) 0,001;
- (c) 0,000001
- 1 Testes com pares ordenados obtidos de uma função do terceiro grau.

$$T(x) = -\frac{1}{3}x^3 + \frac{1}{2}x^2 + 42x$$

	xmax	Tmax	
Resultado			
ANALITICO	6,000000000000000000000000000000000000	1,62000000000000000000000000000000000000	

Teste 1: os pontos (x,T) são baseados em um polinômio de ordem 3. Utiliza-se h=1.

Função: POLINOMIO GRAU 4

DADOS DE ENTRADA:

- $x(2) = \quad 6,000000250000000000000000000000003260000E + 00$
- x(4)= 8,000000249999999999999999999995560000E+00

.....

- T(0)= 1,3866667216666638541666145833333333400000E+02
- T(1) = 1,5583333633333298958332812500000000290000E+02
- T(2)= 1,61999999999999959374999479166666667890000E+02
- T(3) = 1,5516666316666619791666145833333333720000E+02
- T(4)= 1,3333332583333280208332812500000000260000E+02

DADOS DE SAÍDA

Coeficientes A, B, C, D, E:

- E= 4,9303806576313237838233035330174139350000E-32

Ponto crítico e coordenada:

xmax

Tmax

Conclusão: O polinômio interpolado se comporta como esperado. O zeramento de A é confirmado. Os valores para T_{max} e x_{max} são acurados.

Teste 2: os pontos (x,T) são baseados em um polinômio de ordem 3. Utiliza-se h=0,001.

TESTE DE POLINÔMIO

Função: POLINOMIO GRAU 4

DADOS DE ENTRADA:

- x(0)= 5,9980002500000000000000000000000003630000E+00
- x(1) = 5,999000250000000000000000000000003450000E+00

- T(0) = 1,6199997400916526054166145833333333460000E+02
- T(1) = 1,6199999350358267714582812500000000210000E+02
- T(2)= 1,619999999999999999479166666667890000E+02
- T(3) = 1,6199999349641601035416145833333334050000E+02
- T(4) = 1,6199997399083192695832812500000001170000E+02

DADOS DE SAÍDA

coeficientes A, B, C, D, E:

E= -1,3298765982413991714401861771868305420000E-18

D= 4,2000000000000000000886805388921525330000E+01

C= -5,0000000000000000022175660151556360470000E-01

B= -3,33333333333333333333333868757085867474390000E-01

A= -1,0271626370065257882965215693786280120000E-21

Ponto crítico e coordenada:

xmax

Tmax

Conclusão: O polinômio interpolado se comporta como esperado. O zeramento de A é confirmado. Os valores para T_{max} e x_{max} são acurados.

Teste 3: os pontos (x,T) são baseados em um polinômio de ordem 3, Utiliza-se h=0.000001.

TESTE DE POLINÔMIO

Função: POLINOMIO GRAU 4

DADOS DE ENTRADA:

x(4)= 6,0000022499999999999999999999998560000E+00

T(1) = 1.6199999999999999934375014062500000000200000E+02T(2)= 1,6199999999999999999479166666667890000E+02

T(3) = 1,61999999999999989843749348958333333333850000E+02

T(4) = 1.6199999999999670937462031249999999760000E+02

DADOS DE SAÍDA

Coeficientes A, B, C, D, E:

E= 2,8248513449187014894528489535149077090000E-13

D= 4,199999999999858757460748747730468720000E+01

C= -4,999999999997645958145724060528509930000E-01

B= -3,3333333333333464113410427636454358830000E-01

Ponto crítico e coordenada:

xmax

6,00000000000000000000000000009087677630000E+00

O polinômio interpolado se comporta como esperado. O zeramento de A é Conclusão: confirmado. Os valores para T_{max} e x_{max} são acurados. Verificou-se que a precisão não afeta os cálculos de T_{max} e x_{max}.

2 - Testes com pares ordenados obtidos de uma função do segundo grau.

$$T(x) = \frac{1}{2} x^2 + 7 x - 3$$

	xmax	Tmax
Resultado ANALITICO	-7,000000000000000000000000000000000000	-2,75000000000000000000000000000000000000

Teste 1: os pontos (x,T) são baseados em um polinômio de ordem 2. Utiliza-se h=1.

TESTE DE POLINÔMIO

Função: POLINOMIO GRAU 4

DADOS DE ENTRADA:

.....

DADOS DE SAÍDA

coeficientes A, B, C, D, E:

Ponto crítico e coordenada:

xmax

Tmax

Conclusão: O polinômio interpolado se comporta como esperado. O zeramento de A e B são confirmados. Os valores para T_{max} e x_{max} são acurados.

Teste 2: os pontos (x,T) são baseados em um polinômio de ordem 2. Utiliza-se h=0,001.

TESTE DE POLINÔMIO

h= 9,99999999999999999999999999994280000E-04

Função: POLINOMIO GRAU 4

DADOS DE ENTRADA:

- x(3) = -6,9990002500000000000000000000000003450000E+00
- x(4) = -6.998000250000000000000000000000003630000E+00

- T(0) = -2.7499997999499968750000000000000000420000E+01
- T(1) = -2.7499999499749968750000000000000000820000E+01
- T(3) = -2.7499999500249968750000000000000000790000E+01
- T(4) = -2,749999800049996875000000000000000360000E+01

DADOS DE SAÍDA

coeficientes A, B, C, D, E:

- E= -2,999999999999999999754327595915160410000E+00
- D= 7,0000000000000000005283619277075674170000E+00
- 5,000000000000000011322850665222794670000E-01
- B= 1,0784437701776754760219424698494147530000E-20
- A= 3,8518598887744717061119558851698546370000E-22

Ponto crítico e coordenada:

xmax

-6,999999999999999999999999977019800000E+00

Tmax

-2.7499999999999999999999999999999840000E+01

O polinômio interpolado se comporta como esperado. Não há o zeramento Conclusão: completo de A e B devido aos erros de arredondamento. Os valores para T_{max} e x_{max} ainda são acurados.

Teste 3: os pontos (x,T) são baseados em um polinômio de ordem 2. Utiliza-se h=0,000001.

TESTE DE POLINÔMIO

Função: POLINOMIO GRAU 4

DADOS DE ENTRADA:

- x(0) = -7,000002249999999999999999999999998560000E+00

- T(1) = -2.7499999999999218750000000000000001050000E+01

DADOS DE SAÍDA

Coeficientes A, B, C, D, E:

E= -2,9999990751683966657597905499176894550000E+00

D= 7,0000005284751956139399705486319076830000E+00

C= 5,0000011324468342626917583586889879960000E-01

B= 1,0785207816963850402929200015540790110000E-08

A= 3,8518598887744717061119558851698546370000E-10

Ponto crítico e coordenada:

xmax

-6,999999999999999999999984993793280000E+00

Imax

Conclusão: O polinômio interpolado se comporta como esperado. Não há o zeramento completo de A e B devido aos erros de arredondamento. Estes erros crescem à medida que $h\rightarrow 0$ devido à quantidade de cálculos executados no método de diferenças divididas. O programa DDivididas,f90 que está disponível no apêndice D para consulta. Os valores para T_{max} e x_{max} ainda são acurados.

C3, Polinômio numérico de quarta ordem na forma:

$$Ax^{10} + Bx^9 + Cx^8 + Dx^7 + Ex^6 + Fx^5 + Gx^4 + Hx^3 + Ix^2 + Jx + K$$

$$T(x) = -x^{10} - 7x^9 + 477x^8 + 2373x^7 - 88641x^6 - 282933x^5 + 7893103x^4 + 14251447x^3 - 335485578x^2 - 257826240x + 5448643200$$

O polinômio testado é de décimo grau. Os pares ordenados (x,T) são obtidos em polinômio de décimo com precisão de:

- (a) 1;
- (b) 0,001;
- (c) 0,000001.

Os testes executados para os polinômios de terceiro e segundo graus valem aqui devido ser utilizado a mesma programação para obtenção do polinômio interpolado pelo método de diferenças divididas. Portanto, o teste somente foi executado para o polinômio de décimo grau.

	xmax	Tmax
Resultado		
ANALITICO	-3,7773922774304365840709668282908229946899e-01	5,4975593647818534632186337584272058794206e+09

Teste 1: os pontos (x,T) são baseados em um polinômio de ordem 10. Utiliza-se h=1.

```
TESTE DE POLINÔMIO
Função: POLINOMIO GRAU 10
DADOS DE ENTRADA:
x(2=) -3,3777000249999999999999999999999060000E+00
x(5) =
     6,222999749999999999999999999999999800000E-01
x(6) =
x(7=)
     1.6222999750000000000000000000000000940000E+00
x(8) =
    2,6222999750000000000000000000000000940000E+00
x(9) = 3,622299975000000000000000000000000940000E+00
T(0) = 6,7766958703331816038772968949507134400000E+08
T(1) =
     1,6750326982117945739622506582169842180000E+09
T(2)= 2,9593337188859776949980406165304590780000E+09
T(3) = 4,2306917509428313045593935138890955320000E+09
T(4) = 5,1590306389388761332021681565054626550000E+09
T(5) =
     5,4975593642520181420066634024431672820000E+09
T(6) = 5,1628654287700623513589543574246564710000E+09
T(7) = 4,2582404696228441870906013808047635000000E+09
T(8) = 3,0349361296884533684526437111599758060000E+09
T(9) = 1,8047718296782884172657895209434485550000E+09
T(10) = 8,3325574653066292245471420965574731680000E+08
DADOS DE SAÍDA
Coeficientes A, B, C, D, E:
K= 5,4486432000000000000000000000000009990000E+09
J= -2,578262399999999999999999999998650660000E+08
I= -3,354855779999999999999999999999307240000E+08
   1,425144700000000000000000000003478360000E+07
G= 7.89310300000000000000000000014483740000E+06
E= -8,86409999999999999999999999974529260000E+04
D= 2,37300000000000000000000000169194890000E+03
C= 4,770000000000000000000000000325626990000E+02
A= -1,00000000000000000000000000573291570000E+00
Ponto crítico e coordenada:
```

- -3,7773922774304365840709668282906855590000E-01
- -3,<u>777392277430436584070966828290</u>8229946899E-01 Resultado analítico

Tmax

- 5,4975593647818534632186337584272104960000E+09
- 5,4975593647818534632186337584272058794206E+09 Resultado analítico

Conclusão: O polinômio interpolado se comporta como esperado. Os valores para T_{max} e x_{max} ainda são acurados.

Teste 2: os pontos (x,T) são baseados em um polinômio de ordem 10. Utiliza-se h=0,001.

TESTE DE POLINÔMIO

Função: POLINOMIO GRAU 10

DADOS DE ENTRADA:

- x(0) = -3.827000250000000000000000000000000580000E-01

- T(0)= 5,4975508803374025746180588361957083250000E+09
- T(1)= 5,4975539562022130685316432278703212260000E+09
- T(2) = 5,4975563425145531962223959001993560450000E+09
- T(3) = 5,4975580392858508135602163778558403210000E+09
- T(4) = 5,4975590465277314184851527277427052660000E+09
- T(5) = 5,4975593642520181420066634024431672820000E+09
- T(6) = 5,4975589924707317391359228247228640120000E+09
- $\Gamma(7)$ = 5,4975579311960905797511752303350960040000E+09
- T(8) = 5,4975561804405106393961413063148887620000E+09
- T(9)= 5,4975537402166054900114821818808519110000E+09 T(10)= 5,4975506105371862905993253489958151270000E+09
- 1(10) 5,17755001055710027057752551077501512700002

DADOS DE SAÍDA

Coeficientes A, B, C, D, E:

- K= 5,4486431999961022480428449352193367290000E+09
- J = -2,5782624010320910967869407551721900980000E + 08
- I= -3,3548557922978719614971647756681153780000E+08
- H= 1,4251438316521735197190059667852757130000E+07 G= 7,8930627632953594013458445440285344960000E+06
- F= -2,8306084681424616806125265239942961100000E+05
- E= -8,8923091863771214254900846011584821990000E+04
- D= 1,9461944529317658041961084102750474420000E+03
- C= 5,3225787054776227746321376235014193610000E+01
- B = -2,5633938610093066257959297974493819050000E+02
- A= -6,7016947776286964399303944959372018300000E+01

Ponto crítico e coordenada:

Xmax

- -3.7773922774304365840709668281929874660000E-01
- -3,7773922774304365840709668282998229946899E-01 Resultado analítico

Tmax

- 5,4975593647818534632186337584272038780000E+09
- **5,49755936478185346321863375842720587**94206E+09 Resultado analítico

Conclusão: O polinômio interpolado se comporta como esperado. Os valores para T_{max} e x_{max} ainda são acurados.

Teste 3: os pontos (x,T) são baseados em um polinômio de ordem 10. Utiliza-se h=0.000001.

TESTE DE POLINÔMIO

Função: POLINOMIO GRAU 10

DADOS DE ENTRADA:

- T(0) = 5.4975593643785519207195205703330911370000E+09
- T(1) = 5,4975593643546241762288531132593583010000E+09
- T(2) = 5.4975593643300069261001146960414388270000E+09
- T(3) = 5,4975593643047001703454241917639420410000E+09
- T(4) = 5.4975593642787039089769006711263806200000E+09
- T(5) = 5,4975593642520181420066634024431672820000E+09
- T(6) =5,4975593642246428694468318516435982410000E+09
- T(7) = 5,4975593641965780913095256822718507270000E+09
- T(8)= 5,4975593641678238076068647554869705810000E+09
- T(9) = 5.4975593641383800183509691300628623220000E+09T(10) = 5,4975593641082467235539590623882825340000E+09

DADOS DE SAÍDA

Coeficientes A, B, C, D, E:

- K = -1.3467467817197845044718905641016714570000E+27
- J= -3,5656694762438132295752467862150295260000E+28
- I= -4.2482369688941394403073594030279113770000E+29
- H= -2,9993876783082159405708277012966708980000E+30
- G= -1,3897151388285250966488991200508705080000E+31
- F= -4,4153198660520847776912142055233820310000E+31
- E= -9,7417294047364091117219253225259062500000E+31
- D= -1,4738495394796788774546285285470301560000E+32 C= -1,4633208292415139550555152415609662500000E+32
- B= -8,6095841934812718381523072433252953120000E+31
- A= -2,2794880196015974285477532299106132810000E+31 ------

Ponto crítico e coordenada:

- -3,7773922773857144821522659569722438780000E-01
- -3,7773922774304365840709668282908229946899E-01 Resultado analítico

- 5,4975593647816121578216552734375000000000E+09
- **5,497559364781**8534632186337584272058794206E+09 Resultado analítico

Conclusão: O polinômio interpolado se comporta como esperado. Os erros de arredondamento crescem à medida que $h \rightarrow 0$ devido à quantidade de cálculos que o método de diferenças divididas executa. O programa DDivididas.f90 está disponível no apêndice D para consulta.

APÊNDICE D

PROGRAMA DDIVIDIDAS.F90

O programa DDvididas.f90 tem a finalidade de mostrar a extensão e a quantidade de cálculos envolvidos para fazer a interpolação de décimo grau por diferenças divididas. A programação é complexa e teve que ser planejada e testada várias vezes para evitar todos os erros de programação bem como minimizar os erros de arredondamento.

Para a interpolação numérica foi utilizado um microcomputador com processador Intel® Xeon Quad Core X5355 com 2,66 GHz e com 16GB memória RAM e sistema operacional Windows® XP 64 bits. Para a interpolação numérica, o programa foi elaborado em Fortran® 95, versão 9.1 da Intel®, usando precisão quádrupla (Real*16).

```
! DDivididas .f90
1.1
FUNCTIONS:
! DDivididas - Entry point of console application.
ULTIMA VERSÃO 01.06.10 -APROVADA E TESTADA
O programa interpola onze pontos pelo método de Diferenças Divididas.
! Para localizar os zeros das funções utiliza-se o mátodo da bisseção
! Objetivo: testar o polinomio de Diferenças Divididas para o décimo grau
******************
program DDivididas
USE PORTLIB
! variáveis auxiliares
real*16 :: E0, E1, E2, E3, E4, E5, E6, E7, E8, E9, E10
real*16 :: fx0, fx1, fx2, fx3, fx4, fx5, fx6, fx7, fx8, fx9, fx10
real*16 :: fxA0, fxA1, fxA2, fxA3, fxA4, fxA5, fxA6, fxA7, fxA8, fxA9, fxA10
real*16 :: fx0A, fx0B, fx0C, fx0D, fx0E, fx0F, fx0G, fx0H, fx0I, fx0J, fx0K
real*16 :: fx1B, fx1C, fx1D, fx1E, fx1F, fx1G, fx1H, fx1I, fx1J, fx1K
real*16 :: fx2C, fx2D, fx2E, fx2F, fx2G, fx2H, fx2I, fx2J, fx2K
real*16 :: fx3D, fx3E, fx3F, fx3G, fx3H, fx3I, fx3J, fx3K
real*16 :: fx4E, fx4F, fx4G, fx4H, fx4I, fx4J, fx4K
real*16 :: fx5F, fx5G, fx5H, fx5I, fx5J, fx5K
real*16 :: fx6G, fx6H, fx6I, fx6J, fx6K
real*16 :: fx7H, fx7I, fx7J, fx7K
real*16 :: fx8I, fx8J, fx8K
real*16 :: fx9J, fx9K
real*16 :: fx10K
real*16 :: h
real*16 :: R, fx
integer :: P
real*16, dimension(0:10) :: T ! incógnita
real*16, dimension(0:10) :: x ! incógnita
! variáveis da interpolação
real*16 :: wa, wb, wc, DELTAX, ERRO
real*16 :: Z
!Dados
P=5 ! Ponto central
! Dados de T
h= 9.53674316406250q-07
```

```
T(0) = 9.999999994524225503466650313981018130000q-01
T(1) = 9.9999999996628058188667969458105564690000q-01
T(2) = 9.9999999998226971022620989532794383840000q-01
T(3) = 9.9999999999320964005317637309601157780000q-01
T(4) = 9.99999999999910037136752389000641502450000q-01
T(5) = 9.99999999999994190416922270258593092770000q-01
T(6) = 9.9999999999573423845826856176695595220000q-01
T(7) = 9.9999999998647737423468271288750677450000q-01
T(8) = 9.9999999997217131149851189569122027540000q-01
T(9) = 9.9999999995281605024982834432735325150000q-01
T(10) = 9.9999999992841159048872978735078164430000q-01
!INTERPOLAÇÃO
!Método de Newton-Gregory
!Geração do polinômio de 10 ordem
!Definição:
!P10(x) = f(x) + (A(1))f(x)/h(x-x0) + (A(2)f(x)/2!h2)(x-x0)(x-x1)...
!h = h2-h1 (tamanho da malha)
!Escrevendo os termos A(b) *f(x)/b! * h**(b) onde b é a potência ou expoente : 1,2,3,...,10
fxA0 = (T(P-5))
fxA1 = ((-T(P-5)) + T(P-4)) / h
fxA2 = (T(P-5) - 2*T(P-4) + T(P-3)) / (2*(h**2))
fxA3 = ((-T(P-5)) + 3*T(P-4) - 3*T(P-3) + T(P-2)) / (6*(h**3))
fxA4 = (T(P-5) - 4*T(P-4) + 6*T(P-3) - 4*T(P-2) + T(P-1)) / (24*(h**4))
fxA5 = ((-T(P-5)) + 5*T(P-4) - 10*T(P-3) + 10*T(P-2) - 5*T(P-1) + 1*T(P)) / (120*(h**5))
fxA6 = (T(P-5) - 6*T(P-4) + 15*T(P-3) - 20*T(P-2) + 15*T(P-1) - 6*T(P) + 1*T(P+1)) / (720*(h**6))
fxA7 = ((-T(P-5)) + 7*T(P-4) - 21*T(P-3) + 35*T(P-2) - 35*T(P-1) + 21*T(P) - 7*T(P+1) + 1*T(P+2)) / (5040*(h**7))
fxA8 = (T(P-5) - 8*T(P-4) + 28*T(P-3) - 56*T(P-2) + 70*T(P-1) - 56*T(P) + 28*T(P+1) - 8*T(P+2) + 1*T(P+3))
/(40320*(h**8))
fxA9 = ((-T(P-5)) + 9*T(P-4) - 36*T(P-3) + 84*T(P-2) - 126*T(P-1) + 126*T(P) - 84*T(P+1) + 36*T(P+2) - 9*T(P+3) + 126*T(P+1) + 126*T(
1*T(P+4))/
(362880*(h**9))
```

```
fxA10 = (T(P-5) - 10*T(P-4) + 45*T(P-3) - 120*T(P-2) + 210*T(P-1) - 252*T(P) + 210*T(P+1) - 120*T(P+2) + 45*T(P+3) - 120*T(P+3) - 120*T(P+3) + 120
10*T(P+4) + 1*T
(P+5)) / (3628800*(h**10))
!0 - Escrevendo termo independente
fx0A=1
!1- Escrevendo (x-x0)
fx0B = -x(P-5)
fx1B=1
!2- Escrevendo (x-x0)(x-x1)
fx0C = +(x(P-5) * x(P-4))
fx1C = -(x(P-5) + x(P-4))
fx2C = + 1
!3- Escrevendo (x-x0)(x-x1)(x-x2)
fx0D = -(x(P-5) * x(P-4) * x(P-3))
fx1D = + ((x(P-5)*x(P-4)) + (x(P-5)*x(P-3)) &
+ (x(P-4)*x(P-3))
fx2D = -(x(P-5) + x(P-4) + x(P-3))
fx3D = + 1
!4- Escrevendo (x-x0)(x-x1)(x-x2)(x-x3)
fx0E = + (x(P-5) * x(P-4) * x(P-3) * x(P-2))
fx1E = -((x(P-5)*x(P-4)*x(P-3)) + (x(P-5)*x(P-4)*x(P-2)) &
+(x(P-5)*x(P-3)*x(P-2)) &
+(x(P-4)*x(P-3)*x(P-2))
fx2E = + ((x(P-5)*x(P-4)) + (x(P-5)*x(P-3)) + (x(P-5)*x(P-2)) &
+ (x(P-4)*x(P-3)) + (x(P-4)*x(P-2)) &
+ (x(P-3)*x(P-2))
fx3E = -(x(P-5) + x(P-4) + x(P-3) + x(P-2))
f \times 4E = + 1
!5- Escrevendo (x-x0)(x-x1)(x-x2)(x-x3)(x-x4)
fx0F = -(x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P-1))
fx1F = + ((x(P-5)*x(P-4)*x(P-3)*x(P-2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1))
fx2F = -((x(P-5)*x(P-4)*x(P-3)) + (x(P-5)*x(P-4)*x(P-2)) + (x(P-5)*x(P-4)*x(P-1)) &
+(x(P-5)*x(P-3)*x(P-2)) + (x(P-5)*x(P-3)*x(P-1)) &
+ (x(P-5)*x(P-2)*x(P-1)) &
+(x(P-4)*x(P-3)*x(P-2)) + (x(P-4)*x(P-3)*x(P-1)) &
+ (x(P-4)*x(P-2)*x(P-1)) &
+ (x(P-3)*x(P-2)*x(P-1))
fx3F = + ((x(P-5)*x(P-4)) + (x(P-5)*x(P-3)) + (x(P-5)*x(P-2)) + (x(P-5)*x(P-1)) &
+ (x(P-4)*x(P-3)) + (x(P-4)*x(P-2)) + (x(P-4)*x(P-1)) &
+ (x(P-3)*x(P-2)) + (x(P-3)*x(P-1)) &
```

```
+ (x(P-2)*x(P-1))
fx4F = -(x(P-5) + x(P-4) + x(P-3) + x(P-2) + x(P-1))
!6- Escrevendo (x-x0)(x-x1)(x-x2)(x-x3)(x-x4)(x-x5)
fx0G = + (x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P))
fx1G = -((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P))
fx2G = + ((x(P-5)*x(P-4)*x(P-3)*x(P-2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-2)*x(P)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-3)*x(P-2)*x(P)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-4)*x(P-3)*x(P-2)*x(P)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P))
fx3G = -((x(P-5)*x(P-4)*x(P-3)) + (x(P-5)*x(P-4)*x(P-2)) + (x(P-5)*x(P-4)*x(P-1)) + (x(P-5)*x(P-4)*x(P)) & (x(P-5)*x(P-4)*x(P-1)) + (x(P-5)*x(P-4)*x(P-1)) & (x(P-5)*x(P-1)) & (x(P-5
+(x(P-5)*x(P-3)*x(P-2)) + (x(P-5)*x(P-3)*x(P-1)) + (x(P-5)*x(P-3)*x(P)) &
+ (x(P-5)*x(P-2)*x(P-1)) + (x(P-5)*x(P-2)*x(P)) &
+ (x(P-5)*x(P-1)*x(P)) &
+(x(P-4)*x(P-3)*x(P-2)) + (x(P-4)*x(P-3)*x(P-1)) + (x(P-4)*x(P-3)*x(P)) &
+ (x(P-4)*x(P-2)*x(P-1)) + (x(P-4)*x(P-2)*x(P)) &
+ (x(P-4)*x(P-1)*x(P)) &
+ (x(P-3)*x(P-2)*x(P-1)) + (x(P-3)*x(P-2)*x(P)) &
+ (x(P-3)*x(P-1)*x(P)) &
+ (x(P-2)*x(P-1)*x(P))
fx4G = +((x(P-5)*x(P-4)) + (x(P-5)*x(P-3)) + (x(P-5)*x(P-2)) + (x(P-5)*x(P-1)) + (x(P-5)*x(P)) &
+ (x(P-4)*x(P-3)) + (x(P-4)*x(P-2)) + (x(P-4)*x(P-1)) + (x(P-4)*x(P)) &
+ (x(P-3)*x(P-2)) + (x(P-3)*x(P-1)) + (x(P-3)*x(P)) &
+ (x(P-2)*x(P-1)) + (x(P-2)*x(P)) &
+ (x(P-1)*x(P))
fx5G = -(x(P-5) + x(P-4) + x(P-3) + x(P-2) + x(P-1) + x(P))
fx6G=+1
!7- Escrevendo (x-x0)(x-x1)(x-x2)(x-x3)(x-x4)(x-x5)(x-x6)
fxOH = -(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)*x(P+1))
fx1H = + ((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)) &
```

```
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1))
fx2H = -((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3
2) * x (P+1)  &
 + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)) &
 + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)) & (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P-1)*x(P+1)) & (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
 + (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)) &
 + (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)) &
 + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) &
 + (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)) &
 + (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)) &
  + (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1))
fx3H = + ((x(P-5)*x(P-4)*x(P-3)*x(P-2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P)) + (x(P-5)*x(P-3)*x(P-1)) + (x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P
4) *x (P-3) *x (P+1)  &
 + (x(P-5)*x(P-4)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-2)*x(P)) + (x(P-5)*x(P-4)*x(P-2)*x(P+1)) &
 + (x(P-5)*x(P-4)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-1)*x(P+1)) &
 + (x(P-5)*x(P-4)*x(P) *x(P+1)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-3)*x(P-2)*x(P)) + (x(P-5)*x(P-3)*x(P-2)*x(P+1)) & (x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x
 + (x(P-5)*x(P-3)*x(P-1)*x(P)) + (x(P-5)*x(P-3)*x(P-1)*x(P+1)) &
 + (x(P-5)*x(P-3)*x(P) *x(P+1)) &
 + (x(P-5)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-2)*x(P-1)*x(P+1)) &
 + (x(P-5)*x(P-2)*x(P) *x(P+1)) &
+ (x(P-5)*x(P-1)*x(P) *x(P+1)) &
 + (x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-4)*x(P-3)*x(P-2)*x(P)) + (x(P-4)*x(P-3)*x(P-2)*x(P+1)) & (x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x
 + (x(P-4)*x(P-3)*x(P-1)*x(P)) + (x(P-4)*x(P-3)*x(P-1)*x(P+1)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+1)) &
 + (x(P-4)*x(P-2)*x(P-1)*x(P)) + (x(P-4)*x(P-2)*x(P-1)*x(P+1)) &
 + (x(P-4)*x(P-2)*x(P) *x(P+1)) &
 + (x(P-4)*x(P-1)*x(P) *x(P+1)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-3)*x(P-2)*x(P-1)*x(P+1)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+1)) &
 + (x(P-3)*x(P-1)*x(P) *x(P+1)) &
+ (x(P-2)*x(P-1)*x(P) *x(P+1))
fx4H = -((x(P-5)*x(P-4)*x(P-3)) + (x(P-5)*x(P-4)*x(P-2)) + (x(P-5)*x(P-4)*x(P-1)) + (x(P-5)*x(P-4)*x(P)) + (x(P-5)*x(P-4)*x(P)) + (x(P-5)*x(P-4)*x(P-1)) + (x(P-5)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4
 4) *x(P+1)) &
+(x(P-5)*x(P-3)*x(P-2)) + (x(P-5)*x(P-3)*x(P-1)) + (x(P-5)*x(P-3)*x(P)) + (x(P-5)*x(P-3)*x(P-1)) &
 + (x(P-5)*x(P-2)*x(P-1)) + (x(P-5)*x(P-2)*x(P)) + (x(P-5)*x(P-2)*x(P+1)) &
```

```
+ (x(P-5)*x(P-1)*x(P)) + (x(P-5)*x(P-1)*x(P+1)) &
+ (x(P-5)*x(P) *x(P+1)) &
+(x(P-4)*x(P-3)*x(P-2)) + (x(P-4)*x(P-3)*x(P-1)) + (x(P-4)*x(P-3)*x(P)) + (x(P-4)*x(P-3)*x(P-1)) &
+ (x(P-4)*x(P-2)*x(P-1)) + (x(P-4)*x(P-2)*x(P)) + (x(P-4)*x(P-2)*x(P+1)) &
+ (x(P-4)*x(P-1)*x(P)) + (x(P-4)*x(P-1)*x(P+1)) &
+ (x(P-4)*x(P) *x(P+1)) &
+ (x(P-3)*x(P-2)*x(P-1)) + (x(P-3)*x(P-2)*x(P)) + (x(P-3)*x(P-2)*x(P+1)) &
+ (x(P-3)*x(P-1)*x(P)) + (x(P-3)*x(P-1)*x(P+1)) &
+ (x(P-3)*x(P) *x(P+1)) &
+ (x(P-2)*x(P-1)*x(P)) + (x(P-2)*x(P-1)*x(P+1)) &
+ (x(P-2)*x(P) *x(P+1)) &
+ (x(P-1)*x(P) *x(P+1))
fx5H = + ((x(P-5)*x(P-4)) + (x(P-5)*x(P-3)) + (x(P-5)*x(P-2)) + (x(P-5)*x(P-1)) + (x(P-5)*x(P)) + (x(P-5)*x(
+(x(P-4)*x(P-3)) + (x(P-4)*x(P-2)) + (x(P-4)*x(P-1)) + (x(P-4)*x(P)) + (x(P-4)*x(P+1)) &
+ (x(P-3)*x(P-2)) + (x(P-3)*x(P-1)) + (x(P-3)*x(P)) + (x(P-3)*x(P+1)) &
+ (x(P-2)*x(P-1)) + (x(P-2)*x(P)) + (x(P-2)*x(P+1)) &
+ (x(P-1)*x(P)) + (x(P-1)*x(P+1)) &
+ (x(P) * x(P+1))
fx6H = -(x(P-5) + x(P-4) + x(P-3) + x(P-2) + x(P-1) + x(P) + x(P+1))
fx7H = + 1
!8- Escrevendo (x-x0)(x-x1)(x-x2)(x-x3)(x-x4)(x-x5)(x-x6)(x-x7)
fx0I = + (x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+2))
fx1I = -((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1))&
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2))&
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+1))&
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+2) *x(P) *x(P) *x(P+1)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P+2)*x(P-1)*x(P) *x(P+1))&
+ (x(P-5)*x(P-4)*x(P+2)*x(P-2)*x(P-1)*x(P) *x(P+1))&
+ (x(P-5)*x(P+2)*x(P-3)*x(P-2)*x(P-1) *x(P) *x(P+1))&
+ (x(P+2) * x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+1)))
fx2I = + ((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-
3) *x (P-2) *x (P-1) *x
 (P+2)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x
 (P+2)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P-1)*x(P-1)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P
 (P+2) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
 (P+2)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) &
```

```
+ (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P-1)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P
 (P+2)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
  (P+2)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2))
fx3I = -((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4
2) *x (P+1) + (x (P-5) *x (P-
 4) \times x (P-3) \times x (P-2) \times x (P+2) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
 4) \times \times (P-3) \times \times (P-1) \times \times (P+2) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)) + (x(P-5)*x(P-1))
 4) *x (P-3) *x (P) *x (P+2)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
 4) \times \times (P-2) \times \times (P-1) \times \times (P+2) &
 + (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-1))
 4)*x(P-2)*x(P) *x(P+2)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P
 4) *x (P-1) *x (P) *x (P+2)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+2)) &
 +(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)
3) \times x (P-2) \times x (P-1) \times x (P+2) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-1))
3) *x (P-2) *x (P) *x (P+2)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
3) \times x(P-1) \times x(P) \times x(P+2) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
2) *x (P-1) *x (P) *x (P+2)) &
```

```
+ (x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) + (x(P-4)*x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)
3) \times x (P-2) \times x (P-1) \times x (P+2) &
 + (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-4)*x(P-1))
3) \times \times (P-2) \times \times (P) \times \times (P+2) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
3) \times x(P-1) \times x(P) \times x(P+2) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
2) \times \times (P-1) \times \times (P) \times \times (P+2) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) &
+ (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
2) *x (P-1) *x (P) *x (P+2)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2))
fx4I = + ((x(P-5)*x(P-4)*x(P-3)*x(P-2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P)) + (x(P-5)*x(P-3)*x(P-1)) + (x(P-5)*x(P-1)) + (x(P-5)
 4) *x (P-3) *x (P+1) +
   (x(P-5)*x(P-4)*x(P-3)*x(P+2)) &
 +(x(P-5)*x(P-4)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-2)*x(P)) + (x(P-5)*x(P-4)*x(P-2)*x(P+1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)
  (x(P-5)*x(P-4)*x(P-2)*x(P+2)) &
 + (x(P-5)*x(P-4)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-1)*x(P+1)) +
  (x(P-5)*x(P-4)*x(P-1)*x(P+2)) &
 + (x(P-5)*x(P-4)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P) *x(P+2)) &
 + (x(P-5)*x(P-4)*x(P+1)*x(P+2)) &
 +(x(P-5)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-3)*x(P-2)*x(P)) + (x(P-5)*x(P-3)*x(P-2)*x(P+1)) + (x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(
  (x(P-5)*x(P-3)*x(P-2)*x(P+2)) &
 + (x(P-5)*x(P-3)*x(P-1)*x(P)) + (x(P-5)*x(P-3)*x(P-1)*x(P+1)) +
  (x(P-5)*x(P-3)*x(P-1)*x(P+2)) &
 + (x(P-5)*x(P-3)*x(P) *x(P+1)) + (x(P-5)*x(P-3)*x(P) *x(P+2)) &
 + (x(P-5)*x(P-3)*x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-2)*x(P-1)*x(P+1)) +
  (x(P-5)*x(P-2)*x(P-1)*x(P+2)) &
+ (x(P-5)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-2)*x(P) *x(P+2)) &
 + (x(P-5)*x(P-2)*x(P+1)*x(P+2)) &
 + (x(P-5)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P) *x(P+2)) &
```

```
+ (x(P-5)*x(P-1)*x(P+1)*x(P+2)) &
 + (x(P-5)*x(P) *x(P+1)*x(P+2)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-4)*x(P-3)*x(P-2)*x(P)) + (x(P-4)*x(P-3)*x(P-2)*x(P+1)) + (x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(
  (x(P-4)*x(P-3)*x(P-2)*x(P+2)) &
 + (x(P-4)*x(P-3)*x(P-1)*x(P)) + (x(P-4)*x(P-3)*x(P-1)*x(P+1)) +
  (x(P-4)*x(P-3)*x(P-1)*x(P+2)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+1)) + (x(P-4)*x(P-3)*x(P) *x(P+2)) &
 + (x(P-4)*x(P-3)*x(P+1)*x(P+2)) &
 + (x(P-4)*x(P-2)*x(P-1)*x(P)) + (x(P-4)*x(P-2)*x(P-1)*x(P+1)) +
  (x(P-4)*x(P-2)*x(P-1)*x(P+2)) &
 + (x(P-4)*x(P-2)*x(P) *x(P+1)) + (x(P-4)*x(P-2)*x(P) *x(P+2)) &
+ (x(P-4)*x(P-2)*x(P+1)*x(P+2)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P) *x(P+2)) &
 + (x(P-4)*x(P-1)*x(P+1)*x(P+2)) &
 + (x(P-4)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-3)*x(P-2)*x(P-1)*x(P+1)) +
  (x(P-3)*x(P-2)*x(P-1)*x(P+2)) &
 + (x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-3)*x(P-2)*x(P) *x(P+2)) &
+ (x(P-3)*x(P-2)*x(P+1)*x(P+2)) &
 + (x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-3)*x(P-1)*x(P) *x(P+2)) &
+ (x(P-3)*x(P-1)*x(P+1)*x(P+2)) &
 + (x(P-3)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-2)*x(P-1)*x(P) *x(P+2)) &
+ (x(P-2)*x(P-1)*x(P+1)*x(P+2)) &
 + (x(P-2)*x(P) *x(P+1)*x(P+2)) &
 + (x(P-1)*x(P) *x(P+1)*x(P+2))
fx5I = -((x(P-5)*x(P-4)*x(P-3)) + (x(P-5)*x(P-4)*x(P-2)) + (x(P-5)*x(P-4)*x(P-1)) + (x(P-5)*x(P-4)*x(P)) + (x(P-5)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)
 4) *x (P+1) + (x (P-1))
 5) *x (P-4) *x (P+2)  &
+(x(P-5)*x(P-3)*x(P-2)) + (x(P-5)*x(P-3)*x(P-1)) + (x(P-5)*x(P-3)*x(P)) + (x(P-5)*x(P-3)*x(P-3)) + (x(P-5)*x(P-3)) + (
5) *x (P-3) *x (P+2)  &
 + (x(P-5)*x(P-2)*x(P-1)) + (x(P-5)*x(P-2)*x(P)) + (x(P-5)*x(P-2)*x(P+1)) + (x(P-5)*x(P-2)*x(P-2)*x(P-2)*x(P-2) + (x(P-5)*x(P-2)*x(P-2) + (x(P-5)*x(P-2)*x(P-2)) + (x(P-5)*x(P-2)*x(P-2) + (x(P-5)*x(P-2)*x(P-2) + (x(P-5)*x(P-2)*x(P-2)) + (x(P-5)*x(P-2)*x(P-2) + (x(P-5)*x(P-2)*x(P
5)*x(P-2)*x(P+2)) &
 + (x(P-5)*x(P-1)*x(P)) + (x(P-5)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
5) *x (P-1) *x (P+2)  &
+ (x(P-5)*x(P) *x(P+1)) + (x(P-5)*x(P) *x(P+2)) &
 + (x(P-5)*x(P+1)*x(P+2)) &
 +(x(P-4)*x(P-3)*x(P-3)) + (x(P-4)*x(P-3)*x(P-1)) + (x(P-4)*x(P-3)*x(P)) + (x(P-4)*x(P-3)*x(P-3)) + (x(P-4)*x(P-3)) + (x(P-4)*x(P-3)*x(P-3)) + (x(P-4)*x(P-3)) + (x(P-4)*x(P-
4) *x (P-3) *x (P+2)) &
 + (x(P-4)*x(P-2)*x(P-1)) + (x(P-4)*x(P-2)*x(P)) + (x(P-4)*x(P-2)*x(P+1)) + (x(P-4)*x(P-2)*x(P-1)) + (x(P-4)*x(P-1)) + (x(P-4)*x(P-1
 4) *x (P-2) *x (P+2)  &
 + (x(P-4)*x(P-1)*x(P)) + (x(P-4)*x(P-1)*x(P+1)) + (x(P-4)*x(P-1)*x(P-1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
 4) *x (P-1) *x (P+2)  &
```

```
+ (x(P-4)*x(P) *x(P+1)) + (x(P-4)*x(P) *x(P+2)) &
+ (x(P-4)*x(P+1)*x(P+2)) &
+ (x(P-3)*x(P-2)*x(P-1)) + (x(P-3)*x(P-2)*x(P)) + (x(P-3)*x(P-2)*x(P+1)) + (x(P-3)*x(P-2)*x(P-2)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x
3) *x (P-2) *x (P+2)) &
+ (x(P-3)*x(P-1)*x(P)) + (x(P-3)*x(P-1)*x(P+1)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
3) \times \times (P-1) \times \times (P+2) &
+ (x(P-3)*x(P) *x(P+1)) + (x(P-3)*x(P) *x(P+2)) &
+ (x(P-3)*x(P+1)*x(P+2)) &
+ (x(P-2)*x(P-1)*x(P)) + (x(P-2)*x(P-1)*x(P+1)) + (x(P-1)*x(P+1)) + (x(P-1)*x(P-1)) + (x(P-1)*x(P-1)) + (x(P-1)*x(P-1)*x(P-1)) + (x(P-1)*x(P-1)) + (x(P-1)*x(P-1)) + (x(P-1)*x(P-1)) + (x(P-1)*x(P-1)) + (x(P-1)
2) *x (P-1) *x (P+2)  &
+ (x(P-2)*x(P) *x(P+1)) + (x(P-2)*x(P) *x(P+2)) &
+ (x(P-2)*x(P+1)*x(P+2)) &
+ (x(P-1)*x(P) *x(P+1)) + (x(P-1)*x(P) *x(P+2)) &
+ (x(P-1)*x(P+1)*x(P+2)) &
+ (x(P) *x(P+1) *x(P+2))
fx6I = + ((x(P-5)*x(P-4)) + (x(P-5)*x(P-3)) + (x(P-5)*x(P-2)) + (x(P-5)*x(P-1)) + (x(P-5)*x(P)) + (x(P-5)*x(P+1)) + (x
5) *x (P+2)) &
+(x(P-4)*x(P-3)) + (x(P-4)*x(P-2)) + (x(P-4)*x(P-1)) + (x(P-4)*x(P)) + (x(P-4)*x(P+1)) + (x(P-4)*x(P
+ (x(P-3)*x(P-2)) + (x(P-3)*x(P-1)) + (x(P-3)*x(P)) + (x(P-3)*x(P+1)) + (x(P-3)*x(P+2)) &
+ (x(P-2)*x(P-1)) + (x(P-2)*x(P)) + (x(P-2)*x(P+1)) + (x(P-2)*x(P+2)) &
+ (x(P-1)*x(P)) + (x(P-1)*x(P+1)) + (x(P-1)*x(P+2)) &
+ (x(P) * x(P+1)) + (x(P) *x(P+2)) &
+ (x(P+1)*x(P+2))
fx7I = -(x(P-5) + x(P-4) + x(P-3) + x(P-2) + x(P-1) + x(P) + x(P+1) + x(P+2))
fx8I=1
!9- Escrevendo (x-x0)(x-x1)(x-x2)(x-x3)(x-x4)(x-x5)(x-x6)(x-x7)(x-x8)
fx0J = -(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)*x(P+1)*x(P+1)*x(P+3))
fx1J = +((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+3) *x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+3)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P+3)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-5)*x(P-4)*x(P+3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+ (x(P-5)*x(P+3)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+ (x(P+3)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2))
fx2J = -((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) &
```

```
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3))&
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3))
+(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+3))&
+(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+3))&
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3))&
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3))&
+(x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3))&
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3))
+(x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3))&
+(x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+(x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3))&
+(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) &
+(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+(x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3))&
+(x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+(x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+(x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3))
fx3J = +((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3
3) *x (P-2) *x (P-1) *x
(P+2) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
(P+2) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x
(P+2) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+3)) & (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+3)) & (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+3)) & (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(
+ (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+3)) &
```

```
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x
(P+2) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P-1)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P
(P+2) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P-1)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P
(P+2) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
```

```
+ (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
 + (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
 + (x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
 + (x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
 + (x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)))
fx4J = -((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-4)*x(P-3)*x(P-4)*x(P-4)*x(P-3)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4
2) *x (P+1) + (x (P-5) *x (P-1)) + (x (P-1) *x (P
 4) \times (P-3) \times (P-2) \times (P+2) + (x(P-5) \times (P-4) \times (P-3) \times (P-2) \times (P+3)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
 4) \times (P-3) \times (P-1) \times (P+2) + (\times (P-5) \times (P-4) \times (P-3) \times (P-1) \times (P+3)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)) + (x(P-5)*x(P-1))
 4) \times (P-3) \times (P) \times (P+2) + (x(P-5) \times (P-4) \times (P-3) \times (P) \times (P+3)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+3)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P+2)*x(P+3)) &
 + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
 4) \times (P-2) \times (P-1) \times (P+2) + (x(P-5) \times (P-4) \times (P-2) \times (P-1) \times (P+3)) &
 + (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-1))
4) * x (P-2) * x (P) * x (P+2)) + (x (P-5) * x (P-4) * x (P-2) * x (P) * x (P+3)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+3)) &
 + (x(P-5)*x(P-4)*x(P-2)*x(P+2)*x(P+3)) &
 + (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*
 4) \times \times (P-1) \times \times (P) \times \times (P+2) + (\times (P-5) \times \times (P-4) \times \times (P-1) \times \times (P) \times \times (P+3)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+3)) &
 + (x(P-5)*x(P-4)*x(P-1)*x(P+2)*x(P+3)) &
 + (x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+3)) &
 + (x(P-5)*x(P-4)*x(P) *x(P+2)*x(P+3)) &
 + (x(P-5)*x(P-4)*x(P+1)*x(P+2)*x(P+3)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) + (x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3
3) \times (P-2) \times (P-1) \times (P+2) + (x(P-5) \times (P-3) \times (P-2) \times (P-1) \times (P+3) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-1))
3) *x (P-2) *x (P) *x (P+2)) + (x (P-5) *x (P-3) *x (P-2) *x (P) *x (P+3)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+3)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P+2)*x(P+3)) &
 + (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
3) \times (P-1) \times (P) \times (P+2) + (x(P-5) \times (P-3) \times (P-1) \times (P) \times (P+3)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+3)) &
 + (x(P-5)*x(P-3)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+3)) &
 + (x(P-5)*x(P-3)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
2) * x (P-1) * x (P) * x (P+2)) + (x (P-5) * x (P-2) * x (P-1) * x (P) * x (P+3)) &
```

```
+ (x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) + (x(P-4)*x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
3) \times (P-2) \times (P-1) \times (P+2) + (\times (P-4) \times (P-3) \times (P-2) \times (P-1) \times (P+3) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
3) \times (P-2) \times (P) \times (P+2) + (\times (P-4) \times (P-3) \times (P-2) \times (P) \times (P+3)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
3) *x (P-1) *x (P) *x (P+2)) + (x (P-4) *x (P-3) *x (P-1) *x (P) *x (P+3)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
2) *x (P-1) *x (P) *x (P+2) + (x (P-4) *x (P-2) *x (P-1) *x (P) *x (P+3)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) &
+ \ (x (P-4) * x (P-2) * x (P) \ * x (P+1) * x (P+2)) \ + \ (x (P-4) * x (P-2) * x (P) \ * x (P+1) * x (P+3)) \ \&
+ (x(P-4)*x(P-2)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
2)*x(P-1)*x(P) *x(P+2) + (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
 + (x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
```

```
+ (x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) & (P+3)*x(P+1)*x(P+3) & (P+3)*x(P+1)*x(P+3) & (P+3)*x(P+1)*x(P+3) & (P+3)*x(P+1)*x(P+3) & (P+3)*x(P+3) & (P+3)*x(P+3)*x(P+3) & (P+3)*x(P+3) & (P+3)*x(P+3) & (P+3)*x(P+3
+ (x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)))
 fx5J = + ((x(P-5)*x(P-4)*x(P-3)*x(P-2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(
4) *x (P-3) *x (P+1) +
 (x(P-5)*x(P-4)*x(P-3)*x(P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P+3)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-2)*x(P)) + (x(P-5)*x(P-4)*x(P-2)*x(P+1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)
 (x(P-5)*x(P-4)*x(P-2)*x(P+2)) + (x(P-5)*x(P-4)*x(P-2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-1)*x(P+1)) +
 (x(P-5)*x(P-4)*x(P-1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-1)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P) *x(P+2)) + (x(P-5)*x(P-4)*x(P) *x(P+3)) & (x(P-5)*x(P-4)*x(P) & (x(P-5)*x(P-4)*x(P-4)*x(P) & (x(P-5)*x(P-4)*x(P-4)*x(P) & (x(P-5)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(
+ (x(P-5)*x(P-4)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P+2)*x(P+3)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-3)*x(P-2)*x(P)) + (x(P-5)*x(P-3)*x(P-2)*x(P+1)) + (x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(
 (x(P-5)*x(P-3)*x(P-2)*x(P+2)) + (x(P-5)*x(P-3)*x(P-2)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P)) + (x(P-5)*x(P-3)*x(P-1)*x(P+1)) +
 (x(P-5)*x(P-3)*x(P-1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-1)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P) *x(P+1)) + (x(P-5)*x(P-3)*x(P) *x(P+2)) + (x(P-5)*x(P-3)*x(P) *x(P+3)) &
+ (x(P-5)*x(P-3)*x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-3)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-2)*x(P-1)*x(P+1)) +
 (x(P-5)*x(P-2)*x(P-1)*x(P+2)) + (x(P-5)*x(P-2)*x(P-1)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-2)*x(P) *x(P+2)) + (x(P-5)*x(P-2)*x(P) *x(P+3)) &
+ (x(P-5)*x(P-2)*x(P+1)*x(P+2)) + (x(P-5)*x(P-2)*x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-2)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P) *x(P+2)) + (x(P-5)*x(P-1)*x(P) *x(P+3)) &
+ (x(P-5)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-5)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-5)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-5)*x(P+1)*x(P+2)*x(P+3)) &
+(x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-4)*x(P-3)*x(P-2)*x(P)) + (x(P-4)*x(P-3)*x(P-2)*x(P+1)) +
 (x(P-4)*x(P-3)*x(P-2)*x(P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P)) + (x(P-4)*x(P-3)*x(P-1)*x(P+1)) +
 (x(P-4)*x(P-3)*x(P-1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-1)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+1)) + (x(P-4)*x(P-3)*x(P) *x(P+2)) + (x(P-4)*x(P-3)*x(P) *x(P+3)) &
+ (x(P-4)*x(P-3)*x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-3)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P)) + (x(P-4)*x(P-2)*x(P-1)*x(P+1)) +
 (x(P-4)*x(P-2)*x(P-1)*x(P+2)) + (x(P-4)*x(P-2)*x(P-1)*x(P+3)) &
```

```
+ (x(P-4)*x(P-2)*x(P) *x(P+1)) + (x(P-4)*x(P-2)*x(P) *x(P+2)) + (x(P-4)*x(P-2)*x(P) *x(P+3)) &
+ (x(P-4)*x(P-2)*x(P+1)*x(P+2)) + (x(P-4)*x(P-2)*x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-2)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P) *x(P+2)) + (x(P-4)*x(P-1)*x(P) *x(P+3)) &
+ (x(P-4)*x(P-1)*x(P+1)*x(P+2)) + (x(P-4)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-4)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-4)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-4)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-4)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-3)*x(P-2)*x(P-1)*x(P+1)) +
 (x(P-3)*x(P-2)*x(P-1)*x(P+2)) + (x(P-3)*x(P-2)*x(P-1)*x(P+3)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-3)*x(P-2)*x(P) *x(P+2)) + (x(P-3)*x(P-2)*x(P) *x(P+3)) &
+ (x(P-3)*x(P-2)*x(P+1)*x(P+2)) + (x(P-3)*x(P-2)*x(P+1)*x(P+3)) &
+ (x(P-3)*x(P-2)*x(P+2)*x(P+3)) &
+ (x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-3)*x(P-1)*x(P) *x(P+2)) + (x(P-3)*x(P-1)*x(P) *x(P+3)) &
+ (x(P-3)*x(P-1)*x(P+1)*x(P+2)) + (x(P-3)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-3)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-3)*x(P) *x(P+1)*x(P+2)) + (x(P-3)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-3)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-3)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-2)*x(P-1)*x(P) *x(P+2)) + (x(P-2)*x(P-1)*x(P) *x(P+3)) &
+ (x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-2)*x(P-1)*x(P+1)*x(P+3)) &
+ (x(P-2)*x(P-1)*x(P+2)*x(P+3)) &
+ (x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-2)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-2)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-2)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-1)*x(P) *x(P+1)*x(P+3)) &
+ (x(P-1)*x(P) *x(P+2)*x(P+3)) &
+ (x(P-1)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P) *x(P+1) *x(P+2) *x(P+3)))
fx6J = -((x(P-5)*x(P-4)*x(P-3)) + (x(P-5)*x(P-4)*x(P-2)) + (x(P-5)*x(P-4)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
4) *x (P+1) + (x (P-1))
5) *x (P-4) *x (P+2)) + (x (P-5) *x (P-4) *x (P+3)) &
+(x(P-5)*x(P-3)*x(P-2)) + (x(P-5)*x(P-3)*x(P-1)) + (x(P-5)*x(P-3)*x(P)) + (x(P-5)*x(P-3)*x(P-3)) + (x(P-5)*x(P-3)) + (
5) *x (P-3) *x (P+2)) + (x (P-5) *x (P-3) *x (P+3)) &
+(x(P-5)*x(P-2)*x(P-1)) + (x(P-5)*x(P-2)*x(P)) + (x(P-5)*x(P-2)*x(P+1)) + (x(P-5)*x(P-2)*x(P-1)) + (x(P-5)*x(P-1)) + (
5) *x (P-2) *x (P+2)) + (x (P-5) *x (P-2) *x (P+3)) &
+ (x(P-5)*x(P-1)*x(P)) + (x(P-5)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
5) *x (P-1) *x (P+2)) + (x (P-5) *x (P-1) *x (P+3)) &
+ (x(P-5)*x(P) *x(P+1)) + (x(P-5)*x(P) *x(P+2)) + (x(P-5)*x(P) *x(P+3)) &
+ (x(P-5)*x(P+1)*x(P+2)) + (x(P-5)*x(P+1)*x(P+3)) &
+ (x(P-5)*x(P+2)*x(P+3)) &
+(x(P-4)*x(P-3)*x(P-3)) + (x(P-4)*x(P-3)*x(P-1)) + (x(P-4)*x(P-3)*x(P)) + (x(P-4)*x(P-3)*x(P-3)) + (x(P-4)*x(P-3)) + (x(P-4)*x(P-3)*x(P-3)) + (x(P-4)*x(P-3)) + (x(P-4)*x(P-
```

```
4) *x (P-3) *x (P+2) + (x (P-4) *x (P-3) *x (P+3)) &
  + (x(P-4)*x(P-2)*x(P-1)) + (x(P-4)*x(P-2)*x(P)) + (x(P-4)*x(P-2)*x(P+1)) + (x(P-4)*x(P-2)*x(P+1)) + (x(P-4)*x(P-2)*x(P-1)) + (x(P-4)*x(P-1)) + (x(P-4)*x(P
   4) *x (P-2) *x (P+2)) + (x (P-4) *x (P-2) *x (P+3)) &
  + (x(P-4)*x(P-1)*x(P)) + (x(P-4)*x(P-1)*x(P+1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)) + (x(P-4)*x(P-1)*x(P-1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
  4) *x (P-1) *x (P+2) + (x (P-4) *x (P-1) *x (P+3)) &
  + (x(P-4)*x(P) *x(P+1)) + (x(P-4)*x(P) *x(P+2)) + (x(P-4)*x(P) *x(P+3)) &
  + (x(P-4)*x(P+1)*x(P+2)) + (x(P-4)*x(P+1)*x(P+3)) &
  + (x(P-4)*x(P+2)*x(P+3)) &
  + (x(P-3)*x(P-2)*x(P-1)) + (x(P-3)*x(P-2)*x(P)) + (x(P-3)*x(P-2)*x(P+1)) + (x(P-3)*x(P-2)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(
 3) \times x (P-2) \times x (P+2) + (x (P-3) \times x (P-2) \times x (P+3)) &
  + (x(P-3)*x(P-1)*x(P)) + (x(P-3)*x(P-1)*x(P+1)) + (x(P-3)*x(P-1)*x(P+1)) + (x(P-3)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
 3) \times x (P-1) \times x (P+2) + (x (P-3) \times x (P-1) \times x (P+3)) &
  + (x(P-3)*x(P) *x(P+1)) + (x(P-3)*x(P) *x(P+2)) + (x(P-3)*x(P) *x(P+3)) &
  + (x(P-3)*x(P+1)*x(P+2)) + (x(P-3)*x(P+1)*x(P+3)) &
  + (x(P-3)*x(P+2)*x(P+3)) &
  + (x(P-2)*x(P-1)*x(P)) + (x(P-2)*x(P-1)*x(P+1)) + (x(P-1)*x(P-1)*x(P-1)) + (x(P-1)*x(P-1)*x(P-1) + (x(P-1)*x(P-1)*x(P-1)) + (x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)
 2) *x (P-1) *x (P+2) + (x (P-2) *x (P-1) *x (P+3)) &
  + (x(P-2)*x(P) *x(P+1)) + (x(P-2)*x(P) *x(P+2)) + (x(P-2)*x(P) *x(P+3)) &
 + (x(P-2)*x(P+1)*x(P+2)) + (x(P-2)*x(P+1)*x(P+3)) &
  + (x(P-2)*x(P+2)*x(P+3)) &
  + (x(P-1)*x(P) *x(P+1)) + (x(P-1)*x(P) *x(P+2)) + (x(P-1)*x(P) *x(P+3)) &
  + (x(P-1)*x(P+1)*x(P+2)) + (x(P-1)*x(P+1)*x(P+3)) &
  + (x(P-1)*x(P+2)*x(P+3)) &
 + (x(P) *x(P+1) *x(P+2)) + (x(P) *x(P+1) *x(P+3)) &
  + (x(P) *x(P+2) *x(P+3)) &
  + (x(P+1)*x(P+2)*x(P+3))
 fx7J = + ((x(P-5)*x(P-4)) + (x(P-5)*x(P-3)) + (x(P-5)*x(P-2)) + (x(P-5)*x(P-1)) + (x(P-5)*x(P)) + (x(P-5)*x(P-1)) + (x
  5) *x(P+2)) + (x(P+2))
 -5)*x(P+3)) &
  +(x(P-4)*x(P-3)) + (x(P-4)*x(P-2)) + (x(P-4)*x(P-1)) + (x(P-4)*x(P)) + (x(P-4)*x(P+1)) + (x(P-4)*x(P+2)) + (x(P-4)*x(P
  4) *x(P+3)) &
  +(x(P-3)*x(P-2)) + (x(P-3)*x(P-1)) + (x(P-3)*x(P)) + (x(P-3)*x(P+1)) + (x(P-3)*x(P+2)) + (x(P-3)*x(P
 3) *x(P+3)) &
  + (x(P-2)*x(P-1)) + (x(P-2)*x(P)) + (x(P-2)*x(P+1)) + (x(P-2)*x(P+2)) + (x(P-2)*x(P+2))
 2) *x(P+3)) &
  + (x(P-1)*x(P)) + (x(P-1)*x(P+1)) + (x(P-1)*x(P+2)) + (x(P-1)*x(P+2))
1) *x(P+3)) &
 + (x(P) * x(P+1)) + (x(P) * x(P+2)) + (x(P) * x(P+3)) &
 + (x(P+1)*x(P+2)) + (x(P+1)*x(P+3)) &
  + (x(P+2)*x(P+3))
 fx8J = -(x(P-5) + x(P-4) + x(P-3) + x(P-2) + x(P-1) + x(P) + x(P+1) + x(P+2) + x(P+3)
 fx9J=1
   !10- Escrevendo (x-x0)(x-x1)(x-x2)(x-x3)(x-x4)(x-x5)(x-x6)(x-x7)(x-x8)(x-x9)
```

```
fx0K = + (x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+3) * x(P+4))
fx1K = -((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3))&
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+4)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+4)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+4)*x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+4)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P+4)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P-4)*x(P+4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P-5)*x(P+4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) &
+ (x(P+4)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)))
fx2K = +((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1) *x(P+2)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1) *x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2) *x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2) *x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+3)*x(P+4))
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2) *x(P+3)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3)*x(P+4))
+(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2) *x(P+3))
+(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2) *x(P+3)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3) *x(P+4))
+(x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2) *x(P+3)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2) *x(P+4)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3) *x(P+4)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3) *x(P+4)) &
```

```
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)*x(P+4))
 +(x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
 +(x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
 +(x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2) *x(P+3)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2) *x(P+4)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3) *x(P+4)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3) *x(P+4)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3) *x(P+4))
 +(x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
 +(x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
 +(x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4)) &
 +(x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3) *x(P+4))
 fx3K = -(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)*x(P+1) + x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) + x(P+2) + x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P
  4) *x (P-3) *x (P-
 2) \times (P-1) \times (P) \times (P+3) + x(P-5) \times (P-4) \times (P-3) \times (P-2) \times (P-1) \times (P) \times (P+4) & 
 + x (P-5) *x (P-4) *x (P-3) *x (P-2) *x (P-1) *x (P+1) *x (P+2) + x (P-5) *x (P-4) *x (P-3) *x (P-3) *x (P-1) *x (P-1)
 2) \times (P-1) \times (P+1) \times (P+3) + x(P-5) \times (P-4) \times (P-3) \times (P-2) \times (P-1) \times (P+1) \times (P+1) \times (P+4) & 
 + x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P+2) * x(P+3) + x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P+2) * x(P+4) & (P-5) * x(P-4) * x(P-3) * 
 + x(P-5) *x(P-4) *x(P-3) *x(P-2) *x(P-1) *x(P+3) *x(P+4) &
 + x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P) * x(P+1) * x(P+2) + x(P-5) * x(P-4) * x(P-3) * x
 2) \times x(P) \times x(P+1) \times x(P+3) + x(P-5) \times x(P-4) \times x(P-3) \times x(P-2) \times x(P) \times x(P+1) \times x(P+4) &
 + x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P) * x(P+2) * x(P+3) + x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P) * x(P+2) * x(P+4) & 
 + \times (P-5) \times (P-4) \times (P-3) \times (P-2) \times (P) \times (P+3) \times (P+4)  &
 + x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P+1) * x(P+2) * x(P+3) + x(P-5) * x(P-4) * x(P-3) * x(P-2) * x(P+1) * x(P+2) * x(P+4) & (P+2) * x(P+3) * 
 + x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+3)*x(P+4) &
 + x(P-5) *x(P-4) *x(P-3) *x(P-2) *x(P+2) *x(P+3) *x(P+4) &
 + x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2) + x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-
+ x(P-5) * x(P-4) * x(P-3) * x(P-1) * x(P) * x(P+2) * x(P+3) + x(P-5) * x(P-4) * x(P-3) * x(P-1) * x(P) * x(P+2) * x(P+4) & 
 + x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+3)*x(P+4) &
 + \ x (P-5) * x (P-4) * x (P-3) * x (P-1) * x (P+1) * x (P+2) * x (P+3) \\ + \ x (P-5) * x (P-4) * x (P-3) * x (P-1) * x (P+1) * x (P+2) * x (P+4) \\ & \& (P-5) * x (P-4) * x (P-3) * x (P-1) * x (P+1) * x (P+2) * x (P+4) \\ & \& (P-5) * x (P-4) * x (P-3) * x (P-1) * x (P+2) * x (P+4) \\ & \& (P-5) * x (P-4) * x (P-3) * x (P-1) * x (P+2) * x (P+4) \\ & \& (P-5) * x (P-4) * x (P-3) * x (P-4) * x (P+2) * x (P+4) \\ & \& (P-5) * x (P-4) * x (P-3) * x (P-4) * x (P+2) * x (P+4) \\ & \& (P-5) * x (P-4) * x (P-3) * x (P-4) * x (P+2) * x (P+4) \\ & \& (P-5) * x (P-4) * x (P-3) * x (P-4) * x (P-4) * x (P+4) * x (P+4) \\ & \& (P-5) * x (P-4) * 
 + \times (P-5) \times (P-4) \times (P-3) \times (P-1) \times (P+1) \times (P+3) \times (P+4) &
 + \times (P-5) \times (P-4) \times (P-3) \times (P-1) \times (P+2) \times (P+3) \times (P+4) &
 + x(P-5) * x(P-4) * x(P-3) * x(P) * x(P+1) * x(P+2) * x(P+3) + x(P-5) * x(P-4) * x(P-3) * x(P) * x(P+1) * x(P+2) * x(P+4) &
 + \times (P-5) \times (P-4) \times (P-3) \times (P) \times (P+1) \times (P+3) \times (P+4) &
 + x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+2)*x(P+3)*x(P+4) &
  + x(P-5) * x(P-4) * x(P-3) * x(P+1) * x(P+2) * x(P+3) * x(P+4) &
 + x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2) + x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-
1) * x (P) * x (P+1) * x (P+3) + x (P-5) * x (P-4) * x (P-2) * x (P-1) * x (P) * x (P+1) * x (P+4) & (P+3) * x (P+1) * x (P+2) * x (P+1) * x (P+3) * x (P+
 + x(P-5) * x(P-4) * x(P-2) * x(P-1) * x(P) * x(P+2) * x(P+3) + x(P-5) * x(P-4) * x(P-2) * x(P-1) * x(P) * x(P+2) * x(P+4) &
 + x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+3)*x(P+4) &
  + x(P-5) * x(P-4) * x(P-2) * x(P-1) * x(P+1) * x(P+2) * x(P+3) + x(P-5) * x(P-4) * x(P-2) * x(P-1) * x(P+1) * x(P+2) * x(P+4) & (P+3) * x(P+3) *
```

```
+ \times (P-5) \times (P-4) \times (P-2) \times (P-1) \times (P+1) \times (P+3) \times (P+4) &
+ x(P-5) *x(P-4) *x(P-2) *x(P-1) *x(P+2) *x(P+3) *x(P+4) &
+ x(P-5) * x(P-4) * x(P-2) * x(P) * x(P+1) * x(P+2) * x(P+3) + x(P-5) * x(P-4) * x(P-2) * x(P) * x(P+1) * x(P+2) * x(P+4) & 
+ x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+3)*x(P+4) &
+ x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+2)*x(P+3)*x(P+4) &
+ x(P-5) *x(P-4) *x(P-2) *x(P+1) *x(P+2) *x(P+3) *x(P+4) &
+ \times (P-5) \times (P-4) \times (P-1) \times (P) \times (P+1) \times (P+1) \times (P+2) \times (P+3) + \times (P-5) \times (P-4) \times (P-4) \times (P) \times (P+1) \times (P+1) \times (P+2) \times (P+4) & (P+3) \times (P
+ x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+3)*x(P+4) &
+ x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+2)*x(P+3)*x(P+4) &
+ \times (P-5) \times (P-4) \times (P-1) \times (P+1) \times (P+2) \times (P+3) \times (P+4) &
+ x (P-5) *x (P-4) *x (P) *x (P+1) *x (P+2) *x (P+3) *x (P+4) &
+ x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2) + x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-
1) \times \times (P) \times \times (P+1) \times \times (P+3) + \times (P-5) \times \times (P-3) \times \times (P-2) \times \times (P-1) \times \times (P) \times \times (P+1) \times \times (P+4) &
+ x(P-5) * x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+2) * x(P+3) + x(P-5) * x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+2) * x(P+4) &
+ x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3)*x(P+4) &
+ x(P-5) * x(P-3) * x(P-2) * x(P-1) * x(P+1) * x(P+2) * x(P+3) + x(P-5) * x(P-3) * x(P-2) * x(P-1) * x(P+1) * x(P+2) * x(P+4) & (P+3) * x(P+1) * x(P+2) * x(P+3) * 
+ x(P-5) *x(P-3) *x(P-2) *x(P-1) *x(P+1) *x(P+3) *x(P+4) &
+ x(P-5) *x(P-3) *x(P-2) *x(P-1) *x(P+2) *x(P+3) *x(P+4) &
+ x(P-5) * x(P-3) * x(P-2) * x(P) * x(P+1) * x(P+2) * x(P+3) + x(P-5) * x(P-3) * x(P-2) * x(P) * x(P+1) * x(P+2) * x(P+4) &
+ x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3)*x(P+4) &
+ x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3)*x(P+4) &
+ x(P-5) *x(P-3) *x(P-2) *x(P+1) *x(P+2) *x(P+3) *x(P+4) &
+ x(P-5) * x(P-3) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+3) + x(P-5) * x(P-3) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+4) &
+ x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3)*x(P+4) &
+ \times (P-5) \times (P-3) \times (P-1) \times (P) \times (P+2) \times (P+3) \times (P+4) &
+ x(P-5) *x(P-3) *x(P-1) *x(P+1) *x(P+2) *x(P+3) *x(P+4) &
+ x (P-5) *x (P-3) *x (P) *x (P+1) *x (P+2) *x (P+3) *x (P+4) &
+ x(P-5) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+3) + x(P-5) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+4) & 
+ x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)*x(P+4) &
+ x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)*x(P+4) &
+ \times (P-5) \times (P-2) \times (P-1) \times (P+1) \times (P+2) \times (P+3) \times (P+4) &
+ x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)*x(P+4) &
+ x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)*x(P+4) &
+ x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+2) + x(P-4) * x(P-3) * x(P-2) * x(P-2) * x(P-1) * x
+ x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+2) * x(P+3) + x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+2) * x(P+4) &
+ x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+3) * x(P+4) &
+ x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P+1) * x(P+2) * x(P+3) + x(P-4) * x(P-3) * x(P-2) * x(P-1) * x(P+1) * x(P+2) * x(P+4) & (P+3) * x(P+3) * 
+ x(P-4) *x(P-3) *x(P-2) *x(P-1) *x(P+1) *x(P+3) *x(P+4) &
+ x(P-4) *x(P-3) *x(P-2) *x(P-1) *x(P+2) *x(P+3) *x(P+4) &
+ x(P-4) * x(P-3) * x(P-2) * x(P) * x(P+1) * x(P+2) * x(P+3) + x(P-4) * x(P-3) * x(P-2) * x(P) * x(P+1) * x(P+2) * x(P+4) &
+ x(P-4) * x(P-3) * x(P-2) * x(P) * x(P+1) * x(P+3) * x(P+4) &
 + x(P-4) * x(P-3) * x(P-2) * x(P) * x(P+2) * x(P+3) * x(P+4) &
```

```
+ x(P-4) *x(P-3) *x(P-2) *x(P+1) *x(P+2) *x(P+3) *x(P+4) &
 + \times (P-4) \times (P-3) \times (P-1) \times (P) \times (P+1) \times (P+2) \times (P+3) + \times (P-4) \times (P-3) \times (P-1) \times (P) \times (P+1) \times (P+2) \times (P+4) & (P+3) \times (P
 + x(P-4) * x(P-3) * x(P-1) * x(P) * x(P+1) * x(P+3) * x(P+4) &
 + x(P-4) *x(P-3) *x(P-1) *x(P) *x(P+2) *x(P+3) *x(P+4) &
 + \times (P-4) \times (P-3) \times (P-1) \times (P+1) \times (P+2) \times (P+3) \times (P+4) &
 + x(P-4) * x(P-3) * x(P) * x(P+1) * x(P+2) * x(P+3) * x(P+4) &
 + x(P-4) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+3) + x(P-4) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+4) &
 + x(P-4) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+3) * x(P+4) &
 + x(P-4) * x(P-2) * x(P-1) * x(P) * x(P+2) * x(P+3) * x(P+4) &
 + \times (P-4) \times (P-2) \times (P-1) \times (P+1) \times (P+2) \times (P+3) \times (P+4) &
 + x(P-4) * x(P-2) * x(P) * x(P+1) * x(P+2) * x(P+3) * x(P+4) &
 + x(P-4) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+3) * x(P+4) &
 + x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+3) + x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+2) * x(P+4) & (P+3) * x(P+1) * x(P+2) * x(P+3) * x(P+
 + x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+1) * x(P+3) * x(P+4) &
 + x(P-3) * x(P-2) * x(P-1) * x(P) * x(P+2) * x(P+3) * x(P+4) &
 + \times (P-3) \times (P-2) \times (P-1) \times (P+1) \times (P+2) \times (P+3) \times (P+4) &
 + x(P-3) * x(P-2) * x(P) * x(P+1) * x(P+2) * x(P+3) * x(P+4) &
  + x(P-3) *x(P-1) *x(P) *x(P+1) *x(P+2) *x(P+3) *x(P+4) &
 + x(P-2) *x(P-1) *x(P) *x(P+1) *x(P+2) *x(P+3) *x(P+4)
 fx4K = +((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4
 3) *x (P-2) *x (P-1) *x
   (P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x
  (P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+4)) &
  +(x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P-4)*x(P-3)*x(P-3)*x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-
 2) *x (P+1) *x (P+
 4)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+2)*x(P+4)) &
  + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P+3)*x(P+4)) &
  + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x
    (P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+4)) & \\ (P+2) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+4)) & \\ (P+2) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+4)) & \\ (P+2) + (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P-3)*x(P-1)*x(P) *x(P-3)*x(P-1)*x(P) *x(P-3)*x(P-1)*x(P-3)*x(P-1)*x(P-3)*x(P-1)*x(P-3)*x(P-1)*x(P-3)*x(P-3)*x(P-1)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*
  +(x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*
1) *x(P+1) *x(P+
  4)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+3)*x(P+4)) &
  +(x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-3)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*
  *x(P+1)*x(P+
 4)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+3)*x(P+4)) &
  + (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+4)) & (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+4)) & (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+3)) & (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+3)) & (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+3)) & (x(P-5)*x(P-4)*x(P-3)*x(P+3)*x(P+3)) & (x(P-5)*x(P-3)*x(P-3)*x(P+3)*x(P+3) & (x(P-5)*x(P-3)*x(P-3)*x(P+3) & (x(P-5)*x(P-3)*x(P-3)*x(P+3)) & (x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x
   + (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+3)*x(P+4)) &
```

```
+ (x(P-5)*x(P-4)*x(P-3)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
  (P+2)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+4)) &
 +(x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P-1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P-1)*x(P+3)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P
1) *x (P+1) *x (P+
 4)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+3)*x(P+4)) &
 +(x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P)
*x(P+1)*x(P+
4)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-2)*x(P+2)*x(P+3)*x(P+4)) &
 +(x(P-5)*x(P-4)*x(P-1)*x(P)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-1)*x(P)*x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-1)*x(P)
 *x(P+1)*x(P+
4)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-4)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x
 (P+2)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+4)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+3)) + (x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*
1) *x (P+1) *x (P+
4)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+3)*x(P+4)) &
 +(x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*
*x(P+1)*x(P+
 4)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P-2)*x(P+2)*x(P+3)*x(P+4)) &
```

```
+(x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-3)*x(P-3)*x(P-1)*x(P-3) *x(P-1)*x(P-3) *x(P-1)*x(P-3) *x(P-3)*x(P-3) *x(P-3)*x(P-3) *x(P-3) *x(
 *x(P+1)*x(P+
 4)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
 +(x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-1)*x(P+3)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1) + (x(P-5)*x(P-
 *x(P+1)*x(P+
4)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
 + \ (x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) \ + \ (x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) \ \& \ (x(P-5)*x(P-1)*x(P+1)*x(P+1)*x(P+2)*x(P+4)) \ \& \ (x(P-5)*x(P-1)*x(P+1)*x(P+1)*x(P+2)*x(P+3)) \ \& \ (x(P-5)*x(P-1)*x(P+1)*x(P+1)*x(P+2)*x(P+3)) \ \& \ (x(P-5)*x(P-1)*x(P+1)*x(P+1)*x(P+2)*x(P+3)) \ \& \ (x(P-5)*x(P-1)*x(P+1)*x(P+3)) \ \& \ (x(P-5)*x(P-1)*x(P+1)*x(P+3)) \ \& \ (x(P-5)*x(P-1)*x(P+1)*x(P+3)) \ \& \ (x(P-5)*x(P-1)*x(P+3)) \ \& \ (x(P-5)*
 + (x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-2)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-2)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-1)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P-1)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
 + (x(P-5)*x(P) *x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
 + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(
   (P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+4)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+3)) + (x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*
1) *x(P+1) *x(P+
 4)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+4)) &
 + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+3)*x(P+4)) &
 +(x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P)
 *x(P+1)*x(P+
4)) &
 + (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+4)) &
 + (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+3)*x(P+4)) &
 + (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+4)) & (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+4)) & (x(P-4)*x(P-3)*x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) & (x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(
  + (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+3)*x(P+4)) &
```

```
+ (x(P-4)*x(P-3)*x(P-2)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P-1)*x(P) *x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
*x(P+1)*x(P+
4)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P
*x(P+1)*x(P+
4)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+4)) & (x(P-4)*x(P-2)*x(P) & (x(P-4)*x(P-2)*x(P) & (x(P-4)*x(P-2)*x(P) & (x(P-4)*x(P-2)*x(P-2)*x(P) & (x(P-4)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x
+ (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P) *x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-3)*x(P-1)*x(P+1)*x(P+3)) + (x(P-3)*x(P-1)*x(P+3)) + (x(P-3)*x(P-1)*x(P-1)*x(P+3)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-
*x(P+1)*x(P+
4)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
+(x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
 +(x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
```

```
+ (x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)*x(P+4))
fx5K = -((x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P-2)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3
2) *x (P+1) + (x (P-5) *x (P-
4) *x (P-3) *x (P-2) *x (P+2)) + (x (P-5) *x (P-4) *x (P-3) *x (P-2) *x (P+3)) + (x (P-5) *x (P-4) *x (P-3) *x (P-2) *x (P+4)) & (x (P-5) *x (P-4) *x (P-3) *x (P-2) *x (P-4) *x (P-3) *x (P-4) *x (P-4
+ (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
4) *x (P-3) *x (P-1) *x (P+2) + (x (P-5) *x (P-4) *x (P-3) *x (P-1) *x (P+3)) + (x (P-5) *x (P-4) *x (P-3) *x (P-1) *x (P+4)) & 
+ (x(P-5)*x(P-4)*x(P-3)*x(P) *x(P+1)) + (x(P-5)*x(P-1))
4) *x (P-3) *x (P) *x (P+2) + (x (P-5) *x (P-4) *x (P-3) *x (P) *x (P+3)) + (x (P-5) *x (P-4) *x (P-3) *x (P) *x (P+4)) &
+(x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-3)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
4) * x (P-2) * x (P-1) * x (P+2)) + (x (P-5) * x (P-4) * x (P-2) * x (P-1) * x (P+3)) + (x (P-5) * x (P-4) * x (P-2) * x (P-1) * x (P+4)) & \\
+ (x(P-5)*x(P-4)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-1))
4) *x (P-2) *x (P) *x (P+2)) + (x (P-5) *x (P-4) *x (P-2) *x (P) *x (P+3)) + (x (P-5) *x (P-4) *x (P-2) *x (P) *x (P+4)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
4) *x (P-1) *x (P) *x (P+2) + (x (P-5) *x (P-4) *x (P-1) *x (P) *x (P+3)) + (x (P-5) *x (P-4) *x (P-1) *x (P) *x (P+4)) &
+(x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P+3)*x(P+4)) &
+(x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
3) *x (P-2) *x (P-1) *x (P+2)) + (x (P-5) *x (P-3) *x (P-2) *x (P-1) *x (P+3)) + (x (P-5) *x (P-3) *x (P-1) *x (P+4)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-1))
3) *x (P-2) *x (P) *x (P+2)) + (x (P-5) *x (P-3) *x (P-2) *x (P) *x (P+3)) + (x (P-5) *x (P-3) *x (P-2) *x (P) *x (P+4)) &
+(x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+3)) + (x(P-5)*x(P-3)*x(P-2)*x(P+1)*x(P+4)) &
```

```
+ (x(P-5)*x(P-3)*x(P-2)*x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P-2)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
3) \times (P-1) \times (P) \times (P+2) + (x(P-5) \times (P-3) \times (P-3) \times (P-1) \times (P) \times (P+3) + (x(P-5) \times (P-3) \times (P-1) \times (P) \times (P+4)) 
+(x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+3)) + (x(P-5)*x(P-3)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P+3)*x(P+4)) &
+(x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-3)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-1)*x(P-1)*x(P-1)) + (x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*
2) *x (P-1) *x (P) *x (P+2)) + (x (P-5) *x (P-2) *x (P-1) *x (P) *x (P+3)) + (x (P-5) *x (P-2) *x (P-1) *x (P) *x (P+4)) &
+(x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) + (x(P-5)*x(P-2)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-2)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P+3)*x(P+4)) &
+(x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-2)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-2)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-2)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P-1)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-4)*x(P-3)*x(P-2)*x(P-1)*x(P+1)) + (x(P-4)*x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
3) *x (P-2) *x (P-1) *x (P+2)) + (x (P-4) *x (P-3) *x (P-2) *x (P-1) *x (P+3)) + (x (P-4) *x (P-3) *x (P-2) *x (P-1) *x (P+4)) & 
+ (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x
3)*x(P-2)*x(P) *x(P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P) *x(P+4)) &
+(x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P+1)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
3) * x (P-1) * x (P) * x (P+2)) + (x (P-4) * x (P-3) * x (P-1) * x (P) * x (P+3)) + (x (P-4) * x (P-3) * x (P-1) * x (P) * x (P+4)) &
```

```
+(x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+3)) + (x(P-4)*x(P-3)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+3)) + (x(P-4)*x(P-3)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P-1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-
2) *x (P-1) *x (P) *x (P+2)) + (x (P-4) *x (P-2) *x (P-1) *x (P) *x (P+3)) + (x (P-4) *x (P-2) *x (P-1) *x (P) *x (P+4)) &
+(x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) + (x(P-4)*x(P-2)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-2)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+3)) + (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+4)) & (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+4)) & (x(P-4)*x(P-2)*x(P) *x(P+1)*x(P+3)) & (x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-
+ (x(P-4)*x(P-2)*x(P) *x(P+2)*x(P+3)) + (x(P-4)*x(P-2)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-2)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-4)*x(P-1)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-4)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x
2) * x (P-1) * x (P) * x (P+2)) + (x (P-3) * x (P-2) * x (P-1) * x (P) * x (P+3)) + (x (P-3) * x (P-2) * x (P-1) * x (P) * x (P+4)) & (P-1) * x 
+(x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+3)) + (x(P-3)*x(P-2)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+3)) + (x(P-3)*x(P-2)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P+3)*x(P+4)) &
+(x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+3)) + (x(P-3)*x(P-2)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+3)) + (x(P-3)*x(P-2)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+3)) + (x(P-3)*x(P-2)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-3)*x(P-1)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-3)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
```

```
+ (x(P-3)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) & (x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) & (x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) & (x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) & (x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) & (x(P-3)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) & (x(P-3)*x(P-1)*x(P+3)) & (x(P-3)*x(P-1)*x(P-3)) & (x(P-3)*x(P-3)*x(P-3)) & (x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)) & (x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)) & (x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)) & (x(P-3)*x(P-3)*x(P-3)*x(P-3) & (x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*
+ (x(P-3)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-3)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-3)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-2)*x(P-1)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-2)*x(P-1)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-2)*x(P-1)*x(P+1)*x(P+2)*x(P+4)) & (x(P-2)*x(P+1)*x(P+2)*x(P+4)) & (x(P-2)*x(P+1)*x(P+2)*x(P+4)) & (x(P-2)*x(P+1)*x(P+2)*x(P+4)) & (x(P-2)*x(P+4)) & (x(P-
+ (x(P-2)*x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-2)*x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-2)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P-1)*x(P) *x(P+1)*x(P+2)*x(P+4)) & (x(P-1)*x(P+1)*x(P+2)*x(P+3)) & (x(P-1)*x(P+3)) & (x(P-1)*
+ (x(P-1)*x(P) *x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-1)*x(P) *x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-1)*x(P+1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P) *x(P+1) *x(P+2) *x(P+3) *x(P+4))
fx6K = + ((x(P-5)*x(P-4)*x(P-3)*x(P-2)) + (x(P-5)*x(P-4)*x(P-3)*x(P-1)) + (x(P-5)*x(P-4)*x(P-3)*x(P)) + (x(P-5)*x(P-3)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P)) + (x(P-5)*x(P-4)*x(P-3)*x(P-3)*x(P)) + (x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-
4) *x (P-3) *x (P+1) +
  (x(P-5)*x(P-4)*x(P-3)*x(P+2)) + (x(P-5)*x(P-4)*x(P-3)*x(P+3)) + (x(P-5)*x(P-4)*x(P-3)*x(P+4)) &
+(x(P-5)*x(P-4)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-2)*x(P)) + (x(P-5)*x(P-4)*x(P-2)*x(P+1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-1)) + (x(P-5)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-2)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)
 (x(P-5)*x(P-4)*x(P-2)*x(P+2)) + (x(P-5)*x(P-4)*x(P-2)*x(P+3)) + (x(P-5)*x(P-4)*x(P-2)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P-1)*x(P)) + (x(P-5)*x(P-4)*x(P-1)*x(P+1)) +
 (x(P-5)*x(P-4)*x(P-1)*x(P+2)) + (x(P-5)*x(P-4)*x(P-1)*x(P+3)) + (x(P-5)*x(P-4)*x(P-1)*x(P+4)) &
+(x(P-5)*x(P-4)*x(P) *x(P+1)) + (x(P-5)*x(P-4)*x(P) *x(P+2)) + (x(P-5)*x(P-4)*x(P) *x(P+3)) + (x(P-5)*x(P+3)) + (x(P-5)
*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P+1)*x(P+2)) + (x(P-5)*x(P-4)*x(P+1)*x(P+3)) + (x(P-5)*x(P-4)*x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P+2)*x(P+3)) + (x(P-5)*x(P-4)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-4)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P-2)*x(P-1)) + (x(P-5)*x(P-3)*x(P-2)*x(P)) + (x(P-5)*x(P-3)*x(P-2)*x(P+1)) + (x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x
 (x(P-5)*x(P-3)*x(P-2)*x(P+2)) + (x(P-5)*x(P-3)*x(P-2)*x(P+3)) + (x(P-5)*x(P-3)*x(P-2)*x(P+4)) &
+ (x(P-5)*x(P-3)*x(P-1)*x(P)) + (x(P-5)*x(P-3)*x(P-1)*x(P+1)) +
  (x(P-5)*x(P-3)*x(P-1)*x(P+2)) + (x(P-5)*x(P-3)*x(P-1)*x(P+3)) + (x(P-5)*x(P-3)*x(P-1)*x(P+4)) &
+(x(P-5)*x(P-3)*x(P) *x(P+1)) + (x(P-5)*x(P-3)*x(P) *x(P+2)) + (x(P-5)*x(P-3)*x(P) *x(P+3)) + (x(P-5)*x(P-3)*x(P) *x(P+3)) + (x(P-5)*x(P-3)*x(P) *x(P-5)*x(P-3)*x(P) *x(P-5)*x(P-3)*x(P) *x(P-5)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P
*x(P+4)) &
+(x(P-5)*x(P-3)*x(P+1)*x(P+2)) + (x(P-5)*x(P-3)*x(P+1)*x(P+3)) + (x(P-5)*x(P-3)*x(P+1)*x(P+4)) &
 + (x(P-5)*x(P-3)*x(P+2)*x(P+3)) + (x(P-5)*x(P-3)*x(P+2)*x(P+4)) &
```

```
+ (x(P-5)*x(P-3)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P-1)*x(P)) + (x(P-5)*x(P-2)*x(P-1)*x(P+1)) +
  (x(P-5)*x(P-2)*x(P-1)*x(P+2)) + (x(P-5)*x(P-2)*x(P-1)*x(P+3)) + (x(P-5)*x(P-2)*x(P-1)*x(P+4)) &
+(x(P-5)*x(P-2)*x(P) *x(P+1)) + (x(P-5)*x(P-2)*x(P) *x(P+2)) + (x(P-5)*x(P-2)*x(P) *x(P+3)) + (x(P-5)*x(P-2)*x(P) *x(P-1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)) + (x(P-5)*x(P-
*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P+1)*x(P+2)) + (x(P-5)*x(P-2)*x(P+1)*x(P+3)) + (x(P-5)*x(P-2)*x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P+2)*x(P+3)) + (x(P-5)*x(P-2)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-2)*x(P+3)*x(P+4)) &
+(x(P-5)*x(P-1)*x(P) *x(P+1)) + (x(P-5)*x(P-1)*x(P) *x(P+2)) + (x(P-5)*x(P-1)*x(P) *x(P+3)) + (x(P-5)*x(P-1)*x(P) *x(P-1)*x(P) *x(P-1
*x(P+4)) &
+(x(P-5)*x(P-1)*x(P+1)*x(P+2)) + (x(P-5)*x(P-1)*x(P+1)*x(P+3)) + (x(P-5)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-5)*x(P-1)*x(P+2)*x(P+3)) + (x(P-5)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P-1)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P) *x(P+1)*x(P+2)) + (x(P-5)*x(P) *x(P+1)*x(P+3)) + (x(P-5)*x(P) *x(P+1)*x(P+4)) & (x(P-5)*x(P) *x(P+1)*x(P+2)) & (x(P-5)*x(P) *x(P+2)) & (x(P-5)*x(P) & (
+ (x(P-5)*x(P) *x(P+2)*x(P+3)) + (x(P-5)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-5)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-5)*x(P+1)*x(P+2)*x(P+3)) + (x(P-5)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-5)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-5)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-4)*x(P-3)*x(P-2)*x(P-1)) + (x(P-4)*x(P-3)*x(P-2)*x(P)) + (x(P-4)*x(P-3)*x(P-2)*x(P+1)) + (x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(
 (x(P-4)*x(P-3)*x(P-2)*x(P+2)) + (x(P-4)*x(P-3)*x(P-2)*x(P+3)) + (x(P-4)*x(P-3)*x(P-2)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P-1)*x(P)) + (x(P-4)*x(P-3)*x(P-1)*x(P+1)) +
  (x(P-4)*x(P-3)*x(P-1)*x(P+2)) + (x(P-4)*x(P-3)*x(P-1)*x(P+3)) + (x(P-4)*x(P-3)*x(P-1)*x(P+4)) &
+(x(P-4)*x(P-3)*x(P) *x(P+1)) + (x(P-4)*x(P-3)*x(P) *x(P+2)) + (x(P-4)*x(P-3)*x(P) *x(P+3)) + (x(P-4)*x(P-3)*x(P) *x(P-4)*x(P-3)*x(P) + (x(P-4)*x(P-3)*x(P) *x(P-3)*x(P) + (x(P-4)*x(P-3)*x(P) *x(P-3)*x(P) + (x(P-4)*x(P-3)*x(P) + (x(P-4)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-3)*x(P-
*x(P+4)) &
+(x(P-4)*x(P-3)*x(P+1)*x(P+2)) + (x(P-4)*x(P-3)*x(P+1)*x(P+3)) + (x(P-4)*x(P-3)*x(P+1)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P+2)*x(P+3)) + (x(P-4)*x(P-3)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-3)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P-1)*x(P)) + (x(P-4)*x(P-2)*x(P-1)*x(P+1)) +
 (x(P-4)*x(P-2)*x(P-1)*x(P+2)) + (x(P-4)*x(P-2)*x(P-1)*x(P+3)) + (x(P-4)*x(P-2)*x(P-1)*x(P+4)) &
+(x(P-4)*x(P-2)*x(P) *x(P+1)) + (x(P-4)*x(P-2)*x(P) *x(P+2)) + (x(P-4)*x(P-2)*x(P) *x(P+3)) + (x(P-4)*x(P-2)*x(P)) + (x(P-4)*x(P-2)*x(P-2)*x(P)) + (x(P-4)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x
*x(P+4)) &
+(x(P-4)*x(P-2)*x(P+1)*x(P+2)) + (x(P-4)*x(P-2)*x(P+1)*x(P+3)) + (x(P-4)*x(P-2)*x(P+1)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P+2)*x(P+3)) + (x(P-4)*x(P-2)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-2)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P) *x(P+1)) + (x(P-4)*x(P-1)*x(P) *x(P+2)) + (x(P-4)*x(P-1)*x(P) *x(P+3)) + (x(P-4)*x(P-1)*x(P) *x(P-1)*x(P) *x(P-
*x(P+4)) &
+(x(P-4)*x(P-1)*x(P+1)*x(P+2)) + (x(P-4)*x(P-1)*x(P+1)*x(P+3)) + (x(P-4)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P+2)*x(P+3)) + (x(P-4)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P-1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P) *x(P+1)*x(P+2)) + (x(P-4)*x(P) *x(P+1)*x(P+3)) + (x(P-4)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-4)*x(P) *x(P+2)*x(P+3)) + (x(P-4)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-4)*x(P) *x(P+3)*x(P+4)) &
```

```
+ (x(P-4)*x(P+1)*x(P+2)*x(P+3)) + (x(P-4)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-4)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-4)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P-1)*x(P)) + (x(P-3)*x(P-2)*x(P-1)*x(P+1)) +
 (x(P-3)*x(P-2)*x(P-1)*x(P+2)) + (x(P-3)*x(P-2)*x(P-1)*x(P+3)) + (x(P-3)*x(P-2)*x(P-1)*x(P+4)) &
+(x(P-3)*x(P-2)*x(P) *x(P+1)) + (x(P-3)*x(P-2)*x(P) *x(P+2)) + (x(P-3)*x(P-2)*x(P) *x(P+3)) + (x(P-3)*x(P-2)*x(P) *x(P-3)*x(P-2)*x(P) + (x(P-3)*x(P-2)*x(P) *x(P-3)*x(P-2)*x(P) + (x(P-3)*x(P-2)*x(P) *x(P-3)*x(P-2)*x(P) + (x(P-3)*x(P-2)*x(P) *x(P-3)*x(P-3)*x(P-2)*x(P) + (x(P-3)*x(P-2)*x(P) *x(P-3)*x(P-2)*x(P) + (x(P-3)*x(P-2)*x(P) *x(P-3)*x(P-2)*x(P) + (x(P-3)*x(P-2)*x(P) *x(P-3)*x(P-2)*x(P) + (x(P-3)*x(P-2)*x(P) *x(P-3)*x(P-2)*x(P) + (x(P-3)*x(P-2)*x(P) + (x(P-2)*x(P-2)*x(P) + (x(P-2)*x(P) + (x(P-2)*
*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P+1)*x(P+2)) + (x(P-3)*x(P-2)*x(P+1)*x(P+3)) + (x(P-3)*x(P-2)*x(P+1)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P+2)*x(P+3)) + (x(P-3)*x(P-2)*x(P+2)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P+3)*x(P+4)) &
+(x(P-3)*x(P-1)*x(P) *x(P+1)) + (x(P-3)*x(P-1)*x(P) *x(P+2)) + (x(P-3)*x(P-1)*x(P) *x(P+3)) + (x(P-3)*x(P-1)*x(P)) + (x(P-3)*x(P-1)*x(P-1)*x(P)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-
*x(P+4)) &
+ (x(P-3)*x(P-1)*x(P+1)*x(P+2)) + (x(P-3)*x(P-1)*x(P+1)*x(P+3)) + (x(P-3)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-3)*x(P-1)*x(P+2)*x(P+3)) + (x(P-3)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-3)*x(P-1)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P) *x(P+1)*x(P+2)) + (x(P-3)*x(P) *x(P+1)*x(P+3)) + (x(P-3)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-3)*x(P) *x(P+2)*x(P+3)) + (x(P-3)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-3)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-3)*x(P+1)*x(P+2)*x(P+3)) + (x(P-3)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-3)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P+2)*x(P+3)*x(P+4)) &
+(x(P-2)*x(P-1)*x(P) *x(P+1)) + (x(P-2)*x(P-1)*x(P) *x(P+2)) + (x(P-2)*x(P-1)*x(P) *x(P+3)) + (x(P-2)*x(P-1)*x(P)) + (x(P-2)*x(P-1)*x(P-1)*x(P)) + (x(P-2)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1
*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P+1)*x(P+2)) + (x(P-2)*x(P-1)*x(P+1)*x(P+3)) + (x(P-2)*x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P+2)*x(P+3)) + (x(P-2)*x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-2)*x(P-1)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P) *x(P+1)*x(P+2)) + (x(P-2)*x(P) *x(P+1)*x(P+3)) + (x(P-2)*x(P) *x(P+1)*x(P+4)) & (x(P-2)*x(P) *x(P+4)) & (x(P-2)*x(P+4)) & 
+ (x(P-2)*x(P) *x(P+2)*x(P+3)) + (x(P-2)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-2)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-2)*x(P+1)*x(P+2)*x(P+3)) + (x(P-2)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-2)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-2)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P-1)*x(P) *x(P+1)*x(P+2)) + (x(P-1)*x(P) *x(P+1)*x(P+3)) + (x(P-1)*x(P) *x(P+1)*x(P+4)) &
+ (x(P-1)*x(P) *x(P+2)*x(P+3)) + (x(P-1)*x(P) *x(P+2)*x(P+4)) &
+ (x(P-1)*x(P) *x(P+3)*x(P+4)) &
+ (x(P-1)*x(P+1)*x(P+2)*x(P+3)) + (x(P-1)*x(P+1)*x(P+2)*x(P+4)) &
+ (x(P-1)*x(P+1)*x(P+3)*x(P+4)) &
+ (x(P-1)*x(P+2)*x(P+3)*x(P+4)) &
+ (x(P) *x(P+1)*x(P+2)*x(P+3)) + (x(P) *x(P+1)*x(P+2)*x(P+4)) &
+ (x(P) *x(P+1) *x(P+3) *x(P+4)) &
+ (x(P) *x(P+2) *x(P+3) *x(P+4)) &
+ (x(P+1)*x(P+2)*x(P+3)*x(P+4))
```

```
fx7K = -((x(P-5)*x(P-4)*x(P-3)) + (x(P-5)*x(P-4)*x(P-2)) + (x(P-5)*x(P-4)*x(P-1)) + (x(P-5)*x(P-4)*x(P)) + (x(P-5)*x(P-4)*x(P)) + (x(P-5)*x(P-4)*x(P-1)) + (x(P-5)*x(P-4)*x(P-4)*x(P-4)*x(P-4)) + (x(P-5)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)*x(P-4)
 4) *x (P+1) + (x (P-
 5) \times (P-4) \times (P+2) + (x(P-5) \times (P-4) \times (P+3)) + (x(P-5) \times (P-4) \times (P+4)) &
+(x(P-5)*x(P-3)*x(P-2)) + (x(P-5)*x(P-3)*x(P-1)) + (x(P-5)*x(P-3)*x(P)) + (x(P-5)*x(P-3)*x(P-1)) + (x(P-5)*x(P-1)) + (
5) *x (P-3) *x (P+2) + (x (P-5) *x (P-3) *x (P+3)) + (x (P-5) *x (P-3) *x (P+4)) &
 +(x(P-5)*x(P-2)*x(P-1)) + (x(P-5)*x(P-2)*x(P)) + (x(P-5)*x(P-2)*x(P+1)) + (x(P-5)*x(P-2)*x(P-1)) + (x(P-5)*x(P-1)) + (
 5) \times (P-2) \times (P+2) + (x(P-5) \times (P-2) \times (P+3)) + (x(P-5) \times (P-2) \times (P+4)) &
 + (x(P-5)*x(P-1)*x(P)) + (x(P-5)*x(P-1)*x(P+1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1) + (x(P-5)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-
5) *x (P-1) *x (P+2) + (x (P-5) *x (P-1) *x (P+3)) + (x (P-5) *x (P-1) *x (P+4)) &
 + (x(P-5)*x(P) *x(P+1)) + (x(P-5)*x(P) *x(P+2)) + (x(P-5)*x(P) *x(P+3)) + (x(P-5)*x(P) *x(P+4)) & (x(P-5)*x(P) & (x(P-5)*x(P)) & (x(P-5)*x(P)) & (x(P-5)*x(P) & (x(P-5)*x(P)) & (x(P-5)*x(P)) & (x(P-5)*x(
+ (x(P-5)*x(P+1)*x(P+2)) + (x(P-5)*x(P+1)*x(P+3)) + (x(P-5)*x(P+1)*x(P+4)) &
+ (x(P-5)*x(P+2)*x(P+3)) + (x(P-5)*x(P+2)*x(P+4)) &
 + (x(P-5)*x(P+3)*x(P+4)) &
+(x(P-4)*x(P-3)*x(P-2)) + (x(P-4)*x(P-3)*x(P-1)) + (x(P-4)*x(P-3)*x(P)) + (x(P-4)*x(P-3)*x(P-3)) + (x(P-4)*x(P-3)*x(P-1)) + (x(P-4)*x(P-3)*x(P-3)) + (x(P-4)*x(P-3)) + (x(P-
4) * x (P-3) * x (P+2) + (x (P-4) * x (P-3) * x (P+3)) + (x (P-4) * x (P-3) * x (P+4)) &
 + (x(P-4)*x(P-2)*x(P-1)) + (x(P-4)*x(P-2)*x(P)) + (x(P-4)*x(P-2)*x(P+1)) + (x(P-4)*x(P-2)*x(P-2) + (x(P-4)*x(P-2)*x(P-2)) + (x(P-4)*x(P-2)*x(P-2)*x(P-2) + (x(P-4)*x(P-2)*x(P-2) + (x(P-4)*x(P-2)*x(P-2)) + (x(P-4)*x(P-2)*x(P-2) + (x(P-4)*x(P-2)*x(P-2)) + (x(P-4)*x(P-2)*x(P-2) + (x(P-4)*x(P-2)*x(P-2)) + (x(P-4)*x(P-2)*x(P-2) + (x(P-4)*x(P-2)*x(P-2)) + (x(P-4)*x(P-2)*x(P-2) + (x(P-
4) * x (P-2) * x (P+2) + (x (P-4) * x (P-2) * x (P+3)) + (x (P-4) * x (P-2) * x (P+4)) &
 + (x(P-4)*x(P-1)*x(P)) + (x(P-4)*x(P-1)*x(P+1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1) + (x(P-4)*x(P-1)*x(P-1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)) + (x(P-4)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P
4) * x (P-1) * x (P+2) + (x (P-4) * x (P-1) * x (P+3)) + (x (P-4) * x (P-1) * x (P+4)) &
+(x(P-4)*x(P) *x(P+1)) + (x(P-4)*x(P) *x(P+2)) + (x(P-4)*x(P) *x(P+3)) + (x(P-4)*x(P) *x(P+4)) &
+ (x(P-4)*x(P+1)*x(P+2)) + (x(P-4)*x(P+1)*x(P+3)) + (x(P-4)*x(P+1)*x(P+4)) &
 + (x(P-4)*x(P+2)*x(P+3)) + (x(P-4)*x(P+2)*x(P+4)) &
 + (x(P-4)*x(P+3)*x(P+4)) &
+ (x(P-3)*x(P-2)*x(P-1)) + (x(P-3)*x(P-2)*x(P)) + (x(P-3)*x(P-2)*x(P+1)) + (x(P-3)*x(P-2)*x(P-2)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2)*x(P-2)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2)) + (x(P-3)*x(P-2)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P-2) + (x(P-3)*x(P-2)*x(P
3) \times (P-2) \times (P+2) + (x(P-3) \times (P-2) \times (P+3)) + (x(P-3) \times (P-2) \times (P+4)) &
 + (x(P-3)*x(P-1)*x(P)) + (x(P-3)*x(P-1)*x(P+1)) + (x(P-3)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*
3) \times (P-1) \times (P+2) + (x(P-3) \times (P-1) \times (P+3)) + (x(P-3) \times (P-1) \times (P+4)) &
+ (x(P-3)*x(P) *x(P+1)) + (x(P-3)*x(P) *x(P+2)) + (x(P-3)*x(P) *x(P+3)) + (x(P-3)*x(P) *x(P+4)) &
+ (x(P-3)*x(P+1)*x(P+2)) + (x(P-3)*x(P+1)*x(P+3)) + (x(P-3)*x(P+1)*x(P+4)) &
 + (x(P-3)*x(P+2)*x(P+3)) + (x(P-3)*x(P+2)*x(P+4)) &
+ (x(P-3)*x(P+3)*x(P+4)) &
 + (x(P-2)*x(P-1)*x(P)) + (x(P-2)*x(P-1)*x(P+1)) + (x(P-1)*x(P-1)*x(P-1)) + (x(P-1)*x(P-1)*x(P-1) + (x(P-1)*x(P-1)*x(P-1)) + (x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)*x(P-1)
2) *x (P-1) *x (P+2) + (x (P-2) *x (P-1) *x (P+3)) + (x (P-2) *x (P-1) *x (P+4)) &
+(x(P-2)*x(P) *x(P+1)) + (x(P-2)*x(P) *x(P+2)) + (x(P-2)*x(P) *x(P+3)) + (x(P-2)*x(P) *x(P+4)) &
 + (x(P-2)*x(P+1)*x(P+2)) + (x(P-2)*x(P+1)*x(P+3)) + (x(P-2)*x(P+1)*x(P+4)) &
+ (x(P-2)*x(P+2)*x(P+3)) + (x(P-2)*x(P+2)*x(P+4)) &
+ (x(P-2)*x(P+3)*x(P+4)) &
 +(x(P-1)*x(P) *x(P+1)) + (x(P-1)*x(P) *x(P+2)) + (x(P-1)*x(P) *x(P+3)) + (x(P-1)*x(P) *x(P+4)) &
+ (x(P-1)*x(P+1)*x(P+2)) + (x(P-1)*x(P+1)*x(P+3)) + (x(P-1)*x(P+1)*x(P+4)) &
+ (x(P-1)*x(P+2)*x(P+3)) + (x(P-1)*x(P+2)*x(P+4)) &
+ (x(P-1)*x(P+3)*x(P+4)) &
+ (x(P) *x(P+1) *x(P+2)) + (x(P) *x(P+1) *x(P+3)) + (x(P) *x(P+1) *x(P+4)) &
 + (x(P) *x(P+2) *x(P+3)) + (x(P) *x(P+2) *x(P+4)) &
```

```
+ (x(P) *x(P+3) *x(P+4)) &
 + (x(P+1)*x(P+2)*x(P+3)) + (x(P+1)*x(P+2)*x(P+4)) &
 + (x(P+1)*x(P+3)*x(P+4)) &
 + (x(P+2)*x(P+3)*x(P+4))
fx8K = + ((x(P-5)*x(P-4)) + (x(P-5)*x(P-3)) + (x(P-5)*x(P-2)) + (x(P-5)*x(P-1)) + (x(P-5)*x(P)) + (x(P-5)*x(P-1)) + (x
 5) *x(P+2)) + (x(P+2))
-5)*x(P+3)) + (x(P-5)*x(P+4))&
 +(x(P-4)*x(P-3)) + (x(P-4)*x(P-2)) + (x(P-4)*x(P-1)) + (x(P-4)*x(P)) + (x(P-4)*x(P+1)) + (x(P-4)*x(P+2)) + (x(P-4)*x(P
-4) *x (P+3) + (x (P-4) *x (P+4)) &
+ (x(P-3)*x(P-2)) + (x(P-3)*x(P-1)) + (x(P-3)*x(P)) + (x(P-3)*x(P+1)) + (x(P-3)*x(P+2)) + (x(P-3)*x(P-2)) + (x(P-3)*x(P-2)) + (x(P-3)*x(P-2)) + (x(P-3)*x(P-2)) + (x(P-3)*x(P-2)) + (x(P-3)*x(
-3) *x (P+3) + (x (P-3) *x (P+4)) &
 + (x(P-2)*x(P-1)) + (x(P-2)*x(P)) + (x(P-2)*x(P+1)) + (x(P-2)*x(P+2)) + (x(P-2)*x(P+2))
-2)*x(P+3)) + (x(P-2)*x(P+4))&
+ (x(P-1)*x(P)) + (x(P-1)*x(P+1)) + (x(P-1)*x(P+2)) + (x(P-1)*x(P+2))
-1)*x(P+3)) + (x(P-1)*x(P+4))&
+ (x(P) * x(P+1)) + (x(P) *x(P+2)) + (x(P) *x(P+3)) + (x(P) *x(P+4)) &
+ (x(P+1)*x(P+2)) + (x(P+1)*x(P+3)) + (x(P+1)*x(P+4)) &
 + (x(P+2)*x(P+3)) + (x(P+2)*x(P+4)) &
 + (x(P+3)*x(P+4))
 fx9K = -(x(P-5)+x(P-4)+x(P-3)+x(P-2)+x(P-1)+x(P)+x(P+1)+x(P+2)+x(P+3)+x(P+4))
 f \times 10 K = 1
  1-----
   !Polinômio P10
  fx0 = (fxA0*fx0A) + (fxA1*fx0B) + (fxA2*fx0C) + (fxA3*fx0D) + (fxA4*fx0E) + (fxA5*fx0F) + (fxA6*fx0G) + (fxA7*fx0H) + (fxA7*fx
   (fxA8*fx0I) + (fxA9*
  fx0J) + (fxA10*fx0K)
  fx1 = (fxA1*fx1B) + (fxA2*fx1C) + (fxA3*fx1D) + (fxA4*fx1E) + (fxA5*fx1F) + (fxA6*fx1G) + (fxA7*fx1H) + (fxA8*fx1I) + (fxA6*fx1G) + (fxA7*fx1H) + (fxA8*fx1F) + (fxA7*fx1H) + (fxA8*fx1F) + (fxA7*fx1H) + (fxA8*fx1F) + (fxA7*fx1H) + (fxA7*fx1H) + (fxA8*fx1F) + (fxA7*fx1H) + (fxA7*fx1H) + (fxA8*fx1F) + (fxA7*fx1H) + (fxA7*fx
   (fxA9*
 fx1J) + (fxA10*fx1K)
  fx2 = (fxA2*fx2C) + (fxA3*fx2D) + (fxA4*fx2E) + (fxA5*fx2F) + (fxA6*fx2G) + (fxA7*fx2H) + (fxA8*fx2I) + (fxA9*fx2F) + (fxA9*fx
 fx2J) + (fxA10*fx2K)
  fx3 = (fxA3*fx3D) + (fxA4*fx3E) + (fxA5*fx3F) + (fxA6*fx3G) + (fxA7*fx3H) + (fxA8*fx3I) + (fxA9*fx3F)
 fx3J) + (fxA10*fx3K)
  fx4 = (fxA4*fx4E) + (fxA5*fx4F) + (fxA6*fx4G) + (fxA7*fx4H) + (fxA8*fx4I) + (fxA9*fx4F) + (fxA9*fx
 fx4J) + (fxA10*fx4K)
 fx5 = (fxA5*fx5F) + (fxA6*fx5G) + (fxA7*fx5H) + (fxA8*fx5I) + (fxA9*fx5F)
  fx5J) + (fxA10*fx5K)
 fx6 = (fxA6*fx6G) + (fxA7*fx6H) + (fxA8*fx6I) + (fxA9*fx6F) + (fxA9*fx
 fx6J) + (fxA10*fx6K)
 fx7 = (fxA7*fx7H) + (fxA8*fx7I) + (fxA9*
 fx7J) + (fxA10*fx7K)
 fx8=(fxA8*fx8I) + (fxA9*
 fx8J) + (fxA10*fx8K)
```

```
fx9=(fxA9*
  fx9J) + (fxA10*fx9K)
  fx10=
   (fxA10*fx10K)
 E10= fx10
E9 = fx9
 E8 = fx8
E7 = fx7
E6 = fx6
E5 = fx5
E4 = fx4
 E3 = fx3
E2 = fx2
E1 = fx1
 E0 = fx0
   !método de newton rapson
   !DELTAX=1.0q12
   !ERRO=1.0q-16
  !Z = x(P-3)
  DELTAX=1.0q12
 ERRO=1.0q-12
  Z= x(P-3)
    (DO WHILE QABS (DELTAX) > ERRO)
 *3)) + (3*E3 * (Z**2)) + (2*E2* Z) + (E1)
 wb = (90 \times E10 \times (Z^{**}8)) + (72 \times E9 \times (Z^{**}7)) + (56 \times E8 \times (Z^{**}6)) + (42 \times E7 \times (Z^{**}5)) + (30 \times E6 \times (Z^{**}4)) + (20 \times E5 \times (Z^{**}3)) + (30 \times E6 \times (Z^{**}4)) + (30 \times (Z^{**}4)) + (30 \times E6 \times (Z^{**}4)) + (30
   (12*E4* (Z*
  *2)) + (6*E3 *(Z)) + (2*E2)
 wc = (720 \times E10 \times (Z^{**7})) + (504 \times E9 \times (Z^{**6})) + (336 \times E8 \times (Z^{**5})) + (210 \times E7 \times (Z^{**4})) + (120 \times E6 \times (Z^{**3})) + (60 \times E5 \times (Z^{**2})) + (120 \times E7 \times (Z^{**4})) + (120 \times E7 \times (Z^{**3})) + (120 \times E7 \times (Z^{**4})) + (120 \times E7 \times (Z^{**3})) + (120 \times E7 \times (Z^{**4})) + (120 \times E7 \times (Z^{**3})) + (120 \times E7 \times (Z^{**4})) + (120 \times (Z^{**4})) + (120 \times E7 \times (Z^{**4})) + (120 \times E7 \times (Z^{**4})) + (120 \times (Z^{**4})) + (120
    (24*E4*
   (Z)) + (6*E3)
 DELTAX = (wa*wb) / ((wb**2) - (wa*wc))
  Z=Z-DELTAX
 ENDDO
 R = Z
  fx = (E10* (R**10)) + (E9* (R**9)) + (E8* (R**8)) + (E7* (R**7)) + (E6* (R**6)) + (E5* (R**5)) + (E4* (R**4)) + (E3* (R**4)) + (E4* (R**4))
 R^{**}3)) + (E2*(R^{**})
 2)) + (E1*R) + E0
  open(1, file="saida.txt")
  10 format (1pe50.40)
```

```
20 format (1pe50.40)
Write(1,*) "TESTE DE POLINÔMIO"
Write(1,*) "LEANDRO ALBERTO NOVAK"
Write(1,*) "h="
Write(1,10) h
Write(1,*) "Função : POLINOMIO GRAU 10 "
Write(1,*) "P4 n= "
Write(1,20) n
Write (1, *)
Write(1,*) "-----"
Write(1,*) "DADOS DE ENTRADA:"
Write(1,*) "x(0)="
Write(1,10) x(0)
Write (1, *) "x (1) ="
Write (1, 10) x (1)
Write (1, *) "x (2=)"
Write(1,10) x(2)
Write(1,*) "x(3)="
Write(1,10) x(3)
Write (1, *) "x (4) ="
Write(1,10) x(4)
Write(1,*) "x(5)="
Write(1,10) x(5)
Write (1, *) "x (6) ="
Write(1,10) x(6)
Write (1, *) "x (7=)"
Write(1,10) x(7)
Write(1,*) "x(8)="
Write(1,10) x(8)
Write (1, *) "x (9) ="
Write(1,10) x(9)
Write(1,*) "x(10)="
Write(1,10) x(10)
Write(1,*)
Write(1,*) "-----"
Write(1,*) "T(0)="
Write(1,10) T(0)
Write (1, *) "T (1) ="
Write (1,10) T(1)
Write(1,*) "T(2)="
Write(1,10) T(2)
Write(1,*) "T(3)="
Write(1,10) T(3)
```

```
Write(1,*) "T(4)="
Write(1,10) T(4)
Write(1,*) "T(5)="
Write(1,10) T(5)
Write(1,*) "T(6)="
Write(1,10) T(6)
Write(1,*) "T(7)="
Write(1,10) T(7)
Write(1,*) "T(8)="
Write(1,10) T(8)
Write(1,*) "T(9)="
Write(1,10) T(9)
Write (1, *) "T (10) ="
Write(1,10) T(10)
Write(1,*) "-----"
Write(1,*) "coeficientes A, B, C, D, E:"
Write(1,*) "K"
Write (1, 10) fx0
Write(1,*) "J"
Write(1,10) fx1
Write(1,*) "I"
Write(1,10) fx2
Write(1,*) "H"
Write(1,10) fx3
Write(1,*) "G"
Write(1,10) fx4
Write(1,*) "F"
Write(1,10) fx5
Write(1,*) "E"
Write(1,10) fx6
Write(1,*) "D"
Write(1,10) fx7
Write(1,*) "C"
Write (1, 10) fx8
Write(1,*) "B"
Write(1,10) fx9
Write (1, *) "A"
Write(1,10) fx10
Write(1,*) "-----"
Write(1,*) "Ponto crítico e coordenada:"
Write(1,*) "Xmax"
Write(1,10) Z
Write(1,*) "Tmax"
```

```
Write(1,10) fx
close(1)
VER = SYSTEM("Notepad saida.txt")
end program DDivididas
```