

# Capítulo 21. Variáveis reais de PRECISÃO QUÁDRUPLA e inteiras de alta magnitude

20 Nov 2024

## OBJETIVOS DO CAPÍTULO

- Conceitos de variáveis inteiras de alta magnitude e variáveis reais de precisão quádrupla
- Funções matemáticas intrínsecas e formatos de edição para variáveis reais de precisão quádrupla
- Comandos novos do FORTRAN: INTEGER\*4, INTEGER\*8 e REAL\*16

Observação: para este capítulo, é necessário usar um compilador compatível com os novos comandos do FORTRAN: INTEGER\*8 e REAL\*16, como o Visual Studio 2008 da Microsoft, versão 9.0 de 2007, o [Simply Fortran](#), ou o [GFortran](#), que é livre do Projeto GNU. O compilador que está sendo usado neste curso, não permite usar estes novos comandos.

## 21.1 programa21a.f90

- 1) Objetivos do programa:
  - a) Definir variáveis inteiras de alta magnitude através do comando INTEGER\*8
  - b) Comparar variáveis inteiras do tipo padrão (INTEGER ou INTEGER\*4) com as de alta magnitude
- 2) No Fortran, **criar um projeto** com o nome **programa21a**
- 3) No Fortran, **criar e inserir** no projeto o programa-fonte **programa21a.f90**
- 4) No Fortran, **copiar** exatamente o texto em vermelho mostrado na **Tabela 21.1**.
- 5) Comentários sobre o programa:
  - a) Para definir uma variável inteira padrão pode-se usar os comandos INTEGER ou INTEGER\*4.
  - b) Para definir uma variável inteira de alta magnitude usa-se o comando INTEGER\*8.
  - c) Também existem variáveis inteiras de altíssima magnitude definidas com o comando INTEGER\*16.  
Porém, neste capítulo, não será possível mostrar exemplos com este comando porque o compilador utilizado não permite.
  - d) A Tabela 21.2 apresenta o intervalo de valores para os três tipos de variáveis inteiras.
  - e) As variáveis Fat e Fat4 são inteiras de magnitude padrão. E a variável Fat8 é de alta magnitude.
- 6) Executar **Build, Compile** para compilar o programa
- 7) Gerar o programa-executável fazendo **Build, Build**.
- 8) Executar o programa através de **Build, Execute com o valor 10**. O resultado deve ser o mostrado na Figura 21.1. As três variáveis fornecem o mesmo resultado, como esperado, porque o valor está dentro do intervalo das duas magnitudes usadas.
- 9) **Executar** novamente o programa **com o valor 13**. O resultado deve ser o mostrado na Figura 21.2. Os resultados de Fat e Fat4 são incorretos porque o fatorial do número 13 ultrapassa o limite de suas magnitudes.

Já o resultado de Fat8 está correto. Para Fat e Fat4, valores corretos do fatorial são obtidos para números até 12, e para Fat8 até 20.

#### 10) Executar novamente o programa com outros valores e analisar os resultados.

Tabela 21.1 Programa21a.f90.

```

integer Fat
INTEGER I, N
integer*4 Fat4
integer*8 Fat8

WRITE(*,*) "Entre com um valor inteiro para calcular o seu fatorial"
READ(*,*) N

IF ( N < 0 ) THEN
    WRITE(*,*) "Nao existe fatorial de ", N
ELSE
    FAT = 1
    fat4 = 1
    fat8 = 1
    DO I = 2, N
        FAT = FAT * I
        FAT4 = FAT4 * I
        FAT8 = FAT8 * I
    END DO
    WRITE(*,*) "O fatorial Fat de", N, " eh = ", FAT
    WRITE(*,*) "O fatorial Fat4 de", N, " eh = ", FAT4
    WRITE(*,*) "O fatorial Fat8 de", N, " eh = ", FAT8
END IF

END

```

Tabela 21.2 Tipos de variáveis inteiras do FORTRAN e suas magnitudes

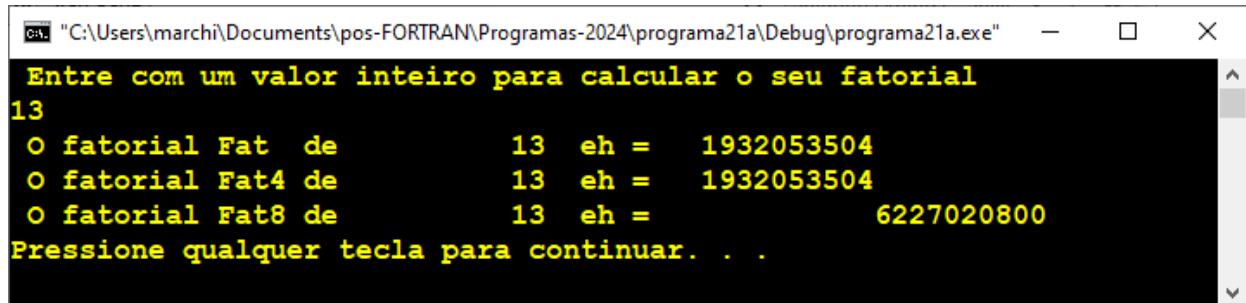
Comando	Intervalo de valores	Valor máximo
integer ou integer*4	-2 <sup>31</sup> a 2 <sup>31</sup> -1	2,147,483,647
integer*8	-2 <sup>63</sup> a 2 <sup>63</sup> -1	9,223,372,036,854,774,807
integer*16	-2 <sup>127</sup> a 2 <sup>127</sup> -1	≈ 1.70141183460469x10 <sup>38</sup>

```

C:\Users\marchi\Documents\pos-FORTRAN\Programas-2024\programa21a\Debug\programa21a.exe
Entre com um valor inteiro para calcular o seu fatorial
10
O fatorial Fat de      10 eh =      3628800
O fatorial Fat4 de     10 eh =      3628800
O fatorial Fat8 de     10 eh =      3628800
Pressione qualquer tecla para continuar. . .

```

Figura 21.1 Resultados da execução do programa21a para valor = 10.



```
Entre com um valor inteiro para calcular o seu fatorial
13
O factorial Fat de      13 eh = 1932053504
O factorial Fat4 de    13 eh = 1932053504
O factorial Fat8 de    13 eh = 6227020800
Pressione qualquer tecla para continuar. . .
```

Figura 21.2 Resultados da execução do programa21a para valor = 13.

## 21.2 programa21b.f90

- 1) Objetivos do programa:
  - a) Definir variáveis reais de precisão quádrupla
  - b) Comparar resultados de variáveis reais de precisões simples, dupla e quádrupla
- 2) No Fortran, **criar um projeto** com o nome **programa21b**
- 3) No Fortran, **criar e inserir** no projeto o programa-fonte **programa21b.f90**
- 4) No Fortran, **copiar** exatamente o texto em vermelho mostrado na **Tabela 21.3**.
- 5) Comentários sobre o programa:
  - a) No capítulo 3 foram abordadas as variáveis do tipo real de precisão simples, definidas com o comando **REAL\*4**, entre outros. Pode-se usar ou não a letra “e” ou “E” para definir o expoente de um número real de precisão simples, dependendo do valor; por exemplo, 340.1 ou 3.401E+2 para  $3.401 \times 10^2$ .
  - b) Já no capítulo 13 foram abordadas as variáveis do tipo real de precisão dupla, definidas com o comando **REAL\*8**, entre outros. Deve-se usar a letra “d” ou “D” para definir o expoente de um número real de precisão dupla; por exemplo, 3.401d+2 para  $3.401 \times 10^2$ .
  - c) Neste capítulo 21 são abordadas as variáveis do tipo real de precisão quádrupla, que podem ser definidas com o comando **REAL\*16**. Deve-se usar a letra “q” ou “Q” para definir o expoente de um número real de precisão quádrupla; por exemplo, 3.401q+2 para  $3.401 \times 10^2$ .
  - d) A Tabela 21.4 apresenta o intervalo de valores para os três tipos de variáveis reais. Os números de dígitos significativos citados são baseados em testes feitos com o programa21b, que permite verificar a precisão prática dos números reais. Porém, o número de dígitos também pode variar com o compilador usado e hardware no qual ele é executado, além da magnitude do valor.
  - e) Em geral, não devem ser usadas variáveis reais de precisão simples em cálculos de engenharia e computação científica, devido à facilidade com que os números são afetados pelos erros de arredondamento nos cálculos.

Tabela 21.3 Programa21b.f90.

```

implicit none
integer N
REAL*4 A, C
REAL*8 B, D, E, F
REAL*16 G, H, I, J

WRITE(*,*) "Entre com o valor de N (inteiro)"
READ(*,*) N

WRITE(*,*) "Entre com o valor de A (real simples)"
READ(*,*) A

WRITE(*,*) "Entre com o valor de B (real dupla)"
READ(*,*) B

WRITE(*,*) "Entre com o valor de G (real quadrupla)"
READ(*,*) G

C = 2/3.0
D = 2/3.0
E = 2/3.0e0
F = 2/3.0d0
H = 2/3.0e0
I = 2/3.0d0
J = 2/3.0q0

WRITE(*,*)
WRITE(*,*) "reais simples"
WRITE(*,*) " A =", A
WRITE(*,*) " C =", C
WRITE(*,*) "N/A =", N/A
WRITE(*,7) " N/A =", N/A
write(*,*) " 1234567890"
7 format(A, 1pe28.20)

WRITE(*,*)
WRITE(*,*) "reais duplas"
WRITE(*,*) " B =", B
WRITE(*,*) " D =", D
WRITE(*,*) " E =", E
WRITE(*,*) " F =", F
WRITE(*,*) "N/B =", N/B
WRITE(*,8) " N/B =", N/B
write(*,*) " 12345678901234567890"
8 format(A, 1pe38.30)

WRITE(*,*)
WRITE(*,*) "reais quadruplas"
WRITE(*,*) " G =", G
WRITE(*,*) " H =", H
WRITE(*,*) " I =", I
WRITE(*,*) " J =", J
WRITE(*,*) "N/G =", N/G
WRITE(*,9) " N/G =", N/G
write(*,*) " 12345678901234567890123456789012345"
9 format(A, 1pe48.40)

END

```

f) Em geral, variáveis reais de precisão dupla são mais adequadas em cálculos de engenharia e computação científica. Porém, elas exigem o dobro da memória computacional em relação às variáveis reais de precisão simples. Já as variáveis reais de precisão quádrupla são menos habituais e exigem o quádruplo da memória computacional em relação às variáveis reais de precisão simples; porém, há certos tipos de aplicações e problemas em que são necessárias, por exemplo, na área de Verificação em Dinâmica dos Fluidos Computacional.

Tabela 21.4 Tipos de variáveis reais do FORTRAN

Precisão	Comando	Dígitos significativos	Intervalo aproximado de valores
Simples	<code>real*4</code>	7 ou 8	$1.18 \times 10^{-38}$ a $3.40 \times 10^{38}$
Dupla	<code>real*8</code>	15 ou 16	$2.23 \times 10^{-308}$ a $1.80 \times 10^{308}$
Quádrupla	<code>real*16</code>	33 ou 34	$3.36 \times 10^{-4932}$ a $1.20 \times 10^{4932}$

- 6) Executar **Build, Compile** para compilar o programa
  - 7) Gerar o programa-executável fazendo **Build, Build**.
  - 8) Executar o programa através de **Build, Execute com os valores N = 2 e A = B = C = 3**. Os resultados devem ser os mostrados na Figura 21.3. Deve-se notar o seguinte:

Figura 21.3 Resultados da execução do programa 21b.

- a) Reais simples: o número de algarismos significativos é 7 para as variáveis A, C e N/A sem formatação. E, ao se usar formatação para N/A, mesmo com 21 dígitos significativos, verifica-se que apenas os 7 primeiros dígitos são corretos para 2/3; os demais são “lixo da memória”, ou seja, dígitos incorretos.
- b) Reais duplas: o número de algarismos significativos é 15 para as variáveis B, D, E, F e N/B sem formatação. E, ao se usar formatação para N/B, mesmo com 31 dígitos significativos, verifica-se que apenas os 16 primeiros dígitos são corretos para 2/3; os demais são “lixo da memória”, ou seja, dígitos incorretos. Além disso, deve-se notar que as variáveis D e E apresentam apenas os 7 primeiros dígitos corretos; os demais estão “contaminados” com a precisão simples devido à forma como a razão 2/3 foi calculada, pois o correto é usar 3.0d0 em vez de 3.0e0 ou 3.0.
- c) Reais quádruplas: o número de algarismos significativos é 33 para as variáveis G, H, I, J e N/G sem formatação. E, ao se usar formatação para N/G, mesmo com 41 dígitos significativos, verifica-se que apenas os 34 primeiros dígitos são corretos para 2/3; os demais são “lixo da memória”, ou seja, dígitos incorretos. Além disso, deve-se notar que as variáveis H e I apresentam respectivamente apenas os 7 e 16 primeiros dígitos corretos; os demais estão “contaminados” respectivamente com as precisões simples e dupla devido à forma como a razão 2/3 foi calculada, pois o correto é usar 3.0q0 em vez de 3.0e0 ou 3.0d0.
- d) Deve-se sempre tomar cuidado com os números reais, em cálculos, definições e dados, para não ocorrer “contaminação” da precisão que realmente se deseja usar.

## 9) Executar novamente o programa com outros dados e analisar os resultados.

### 21.3 programa21c.f90

- 1) Objetivo do programa: realizar cálculos elementares envolvendo variáveis inteira e reais de precisões simples, dupla e quádrupla
- 2) No Fortran, **criar um projeto** com o nome **programa21c**
- 3) No Fortran, **criar e inserir** no projeto o programa-fonte **programa21c.f90**
- 4) No Fortran, **copiar** exatamente o texto em vermelho mostrado na **Tabela 21.5**.
- 5) Comentários sobre o programa:
  - a) Os tipos de cálculos das variáveis I e R4B já foram abordados no programa03d.f90 do capítulo 3, seção 3.2, para explicar as consequências de misturar variáveis dos tipos inteiro e real simples em cálculos.
  - b) Os tipos de cálculos das variáveis R8A e R8B já foram abordados no programa13b.f90 do capítulo 13, seção 13.2, para explicar as consequências de misturar variáveis reais simples e dupla em cálculos.
  - c) As variáveis R16A e R16B são equivalentes às variáveis R8A e R8B mas com precisão quádrupla.
  - d) As variáveis R4C, R8C e R16C exemplificam cálculos envolvendo números com magnitudes muito diferentes, quando o número de magnitude menor pode ser desprezado em relação ao número de magnitude maior dependendo da diferença de ordem entre eles e da precisão dos cálculos.

- e) Quando se atribui uma variável real de precisão menor a outra de precisão maior, esta tem sua precisão deteriorada para a de precisão menor.
- f) Se o objetivo é usar precisão dupla no programa, todas as variáveis, dados e cálculos devem envolver precisão dupla; o mesmo vale para o caso da precisão quádrupla.

Tabela 21.5 Programa21c.f90.

```

implicit none

INTEGER I

REAL*4 R4A, R4B, R4C

REAL*8 R8A, R8B, R8C

REAL*16 R16A, R16B, R16C

WRITE(*,*) "Entre com o valor de R4A (real simples)"
READ(*,*) R4A

I = R4A

R4B = 1.0 / I

R8A = R4A
R8B = 1.0d0 / I

R16A = R4A
R16B = 1.0q0 / I

R4C = 1 + 1.0e-10
R8C = 1 + 1.0d-10
R16C = 1 + 1.0q-30

WRITE(*,*)
WRITE(*,*) "inteiro"
WRITE(*,*) "I = ", I

WRITE(*,*)
WRITE(*,*) "reais simples"
WRITE(*,*) "R4A = ", R4A
WRITE(*,*) "R4B = ", R4B
WRITE(*,*) "R4C = ", R4C

WRITE(*,*)
WRITE(*,*) "reais duplas"
WRITE(*,*) "R8A = ", R8A
WRITE(*,*) "R8B = ", R8B
WRITE(*,*) "R8C = ", R8C

WRITE(*,*)
WRITE(*,*) "reais quadruplas"
WRITE(*,*) "R16A = ", R16A
WRITE(*,*) "R16B = ", R16B
WRITE(*,*) "R16C = ", R16C

END

```

- 6) Executar **Build, Compile** para compilar o programa
- 7) Gerar o programa-executável fazendo **Build, Build**.
- 8) Executar o programa através de **Build, Execute com o valor 6.9 para a variável A.**

- 9) Analisar os resultados mostrados na Figura 21.4 considerando-se os comentários do item 5, acima.  
 10) Executar novamente o programa com outros valores para a variável R4A e analisar os resultados.

```

C:\ "C:\Users\marchi\Documents\pos-FORTRAN\Programas-2024\programa21c\Debug\programa21c.exe"
Entre com o valor de R4A (real simples)
6.9

inteiro
I = 6

reais simples
R4A = 6.900000
R4B = 0.1666667
R4C = 1.000000

reais duplas
R8A = 6.90000009536743
R8B = 0.1666666666666667
R8C = 1.00000000010000

reais quadruplas
R16A = 6.90000009536743164062500000000000000000
R16B = 0.1666666666666666666666666666666666666667
R16C = 1.00000000000000000000000000000000000000100
Pressione qualquer tecla para continuar. . .
  
```

Figura 21.4 Resultados da execução do programa21c com R4A = 6.9.

## 21.4 programa21d.f90

- 1) Objetivos do programa:
  - a) Usar funções matemáticas intrínsecas do FORTRAN para variáveis reais de precisão quádrupla
  - b) Usar formatos de edição para variáveis reais de precisão quádrupla
- 2) No Fortran, **criar um projeto** com o nome **programa21d**
- 3) No Fortran, **criar e inserir** no projeto o programa-fonte **programa21d.f90**
- 4) No Fortran, **copiar** exatamente o texto em vermelho mostrado na **Tabela 21.6**.
- 5) Comentários sobre o programa:
  - a) Os tipos de funções matemáticas intrínsecas do FORTRAN são os mesmos para variáveis reais duplas d quádruplas. Mas os nomes são distintos para preservar a precisão das variáveis; em geral, é necessário apenas trocar a letra D à frente do comando usado com variáveis do tipo real dupla pela letra Q para variáveis reais quádruplas., apresentadas nas Tabelas 13.4 e 13.5 do Capítulo 13. Por exemplo: a função DABS(X), usada para calcular o módulo da variável X do tipo real dupla passa a ser QABS quando X é do tipo real quádrupla.
  - b) Os tipos de formatos de edição usados com variáveis do tipo real simples e dupla também podem ser usados com variáveis do tipo real quádrupla. Estes formatos foram vistos no Capítulo 6, seção 6.3, e no Capítulo 13 na seção 13.4.

- c) Por default, quando não se usa o comando FORMAT e não se especifica o número de algarismos do expoente, as variáveis reais simples, duplas e quádruplas são apresentadas respectivamente com dois, três e quatro algarismos.

Tabela 21.6 Programa21d.f90.

```

USE PORTLIB
IMPLICIT NONE
REAL A4, B4, C4, Pi4
REAL*8 A8, B8, C8, Pi8
REAL*16 A16, B16, C16, Pi16
INTEGER VER

WRITE(*,*) "Entre com o valor de A16 (real quadrupla)"
READ(*,*) A16

OPEN(9,file="saida21d.txt")

A4 = A16
A8 = A16
WRITE(9,*) "A4 = A16 =", A4
WRITE(9,*) "A8 = A16 =", A8

B4 = ABS(A4)
B8 = DABS(A8)
B16 = QABS(A16)
WRITE(9,*) "B4 = Módulo com ABS(A4) =", B4
WRITE(9,*) "B8 = Módulo com DABS(A8) =", B8
WRITE(9,*) "B16 = Módulo com QABS(A16) =", B16

C4 = LOG10(B4)
C8 = DLOG10(B8)
C16 = qLOG10(B16)
WRITE(9,*) "C4 = Logaritmo decimal com LOG10(B4) =", C4
WRITE(9,*) "C8 = Logaritmo decimal com DLOG10(B8) =", C8
WRITE(9,*) "C16 = Logaritmo decimal com qLOG10(B16) =", C16

Pi4 = ACOS(-1.0e0)
Pi8 = DACOS(-1.0d0)
Pi16 = qACOS(-1.0q0)
WRITE(9,*) "Pi com ACOS(-1.0e0) =", Pi4
WRITE(9,*) "Pi com DACOS(-1.0d0) =", Pi8
WRITE(9,*) "Pi com qACOS(-1.0q0) =", Pi16

WRITE(9,*), A16, "= A16 sem formato"

WRITE(9,21) A16, A16, A16, A16, a16
21 FORMAT ("A16 no formato F8.2      =", F8.2,      1/, &
           "A16 no formato F5.0      =", F5.0,      1/, &
           "A16 no formato E15.3E1   =", E15.3E1,   1/, &
           "A16 no formato 1PE15.2   =", 1PE15.2,   1/, &
           "A16 no formato 1PE15.2E4 =", 1PE15.2E4, 1/, &
           "A16 no formato 1PE50.32E4 =", 1PE50.32E4, 1/)

CLOSE(9)

VER = SYSTEM("Notepad saida21d.txt")
END

```

- d) Porém, por default, quando se usa o comando FORMAT mas não se especifica o número de algarismos do expoente, todos os tipos de variáveis reais são apresentados com dois algarismos no expoente.

e) Nos argumentos de funções intrínsecas para variáveis do tipo real quádrupla deve-se definir os números usando-se a letra Q (ou q), em vez de D (ou d) das variáveis reais duplas, para representar o expoente. Em cálculos ou expressões que envolvam números reais, também deve-se fazer isso. Se estas recomendações não forem seguidas ocorrerá erro de compilação ou perda de precisão nos cálculos. Por exemplo, no programa 21d.f90 usa-se -1.0e0 como argumento da função real simples ACOS para calcular o valor de  $\pi$ , -1.0d0 como argumento da função real dupla DACOS e -1.0q0 como argumento da função real quádrupla QACOS.

f) A escala de prioridades entre os operadores matemáticos e as regras adotadas nas operações matemáticas, descritas nas Tabelas 3.4 e 3.5 também se aplicam às variáveis do tipo real de precisão quádrupla, onde pertinente.

- 6) Executar **Build, Compile** para compilar o programa
  - 7) Gerar o programa-executável fazendo **Build, Build**.
  - 8) Executar o programa através de **Build, Execute com o valor -1.2345678901234567890 para a variável A16.**
  - 9) **Analizar os resultados mostrados na Figura 21.5** considerando-se os comentários do item 5, acima.

Figura 21.5 Resultado da execução do programa21d no arquivo saída21d.txt com A16 = -1.2345678901234567890.

- 10) Executar novamente o programa com outros valores para a variável A16 e analisar os resultados; por exemplo,  $A16 = 1e+3, 1.0d-3, 1.2q-30$ . Notar que para este último número, o valor não é apresentado para o formato E15.3E1, sendo apresentados apenas asteriscos que indicam o uso de um formato inadequado; ver a Figura 21.6.

11) Executar novamente o programa com outros valores para a variável A16 e analisar os resultados.

Figura 21.6 Resultado da execução do programa21d no arquivo saída21d.txt com A16 = 1.2q-30.

## 21.5 EXERCÍCIOS

## Exercício 21.1

Confirmar se o fatorial do número 12 é correto ao ser calculado com uma variável integer\*4.

## Exercício 21.2

Confirmar se o fatorial do número 20 é correto ao ser calculado com uma variável integer\*8; e se com 21 já é incorreto.

### Exercício 21.3

Para testar as operações matemáticas básicas, refazer o programa03e.f90, do capítulo 3, seção 3.3, usando variáveis do tipo real quádrupla em vez de real simples.

## Exercício 21.4

Para testar o uso da escala de prioridades entre os operadores matemáticos e as regras adotadas nas operações matemáticas, descritas nas Tabelas 3.4 e 3.5, refazer o programa 03f.f90, do capítulo 3, seção 3.5, usando variáveis do tipo real quádrupla em vez de real simples, escrever os resultados H1 a H12 em arquivo tipo texto, e abrir automaticamente o arquivo de resultados com o aplicativo Bloco de Notas.

Comparar os resultados com aqueles da Figura 3.7.

### Exercício 21.5

Escrever as expressões em linguagem FORTRAN que correspondem às seguintes expressões algébricas, onde  $A$ ,  $B$  e  $L$  são números reais de precisão quádrupla, e  $I$  e  $J$  são números inteiros:

$$a) \quad A = \frac{\sqrt{B} + 5}{100}$$

$$b) \quad A = \frac{1}{10} + \sqrt[3]{B}$$

$$c) \quad L = \frac{1}{3-I} \sqrt{I^2 + \frac{1}{J^3}}$$

$$d) \quad L = \frac{1}{1 + \frac{1}{I}} \left( I^2 + \frac{J}{I^3 - 2} \right)^{\frac{1}{3}}$$

Para verificar se a resposta de cada item está correta, basta implementar a expressão em FORTRAN em um programa e comparar seu resultado com o valor obtido através de uma calculadora para a expressão algébrica correspondente. Os dois resultados devem ser iguais. Isso também vale para o próximo exercício.

### Exercício 21.6

Executar o programa 21d.f90 com  $A16 = 1e50$ , que significa  $1 \times 10^{50}$ . Notar o que ocorre com algumas variáveis reais de precisão simples. Por que isso ocorre?

### Exercício 21.7

Executar o programa 21d.f90 com  $A16 = 1e350$ , que significa  $1 \times 10^{350}$ . Notar o que ocorre com algumas variáveis reais de precisões simples e dupla. Por que isso ocorre?