



UNIVERSIDADE FEDERAL DO PARANÁ

CURSO DE ENGENHARIA MECÂNICA

Disciplina: TM-102 INFORMÁTICA / PROGRAMAÇÃO**Semestre: 2003/2**

8ª aula

INSTRUÇÕES

- Siga o roteiro abaixo.
- Qualquer dúvida, chame o professor.

8.1 OBJETIVOS DA AULA

- 1) VETORES
- 2) SELECT CASE

8.2 CRIANDO UM NOVO PROJETO DO TIPO QUICKWIN APPLICATION

- 1) Na “janela principal” do Fortran execute: File, New
- 2) Na janela “New”: selecione “Project Workspace” e clique sobre o botão “OK”
- 3) Na janela “New Project Workspace”, execute o seguinte:
 - a) Type: selecione “QuickWin Application”
 - b) Location: use o botão BROWSE para indicar o seu diretório no “drive” Server1, diretório (pasta) “alunos_2003_2”.
 - c) Name: indique o nome do projeto. Use “programa08”. O software criará um diretório com este nome, dentro do diretório indicado no passo anterior.
 - d) Clique em “Create”
 - e) No lado esquerdo da “janela principal” deverá aparecer o nome do projeto criado

8.3 INSERINDO UM ARQUIVO DENTRO DO PROJETO

- 1) Na “janela principal” do Fortran, execute: Insert, Files into Project
- 2) Na janela “Insert Files into Project”:
 - a) File Name: entre com o nome do arquivo a ser criado. Use “programa8a.f90”. Clique em “OK”
 - b) Na janela “Microsoft Developer Studio”, clique em YES
 - c) No lado esquerdo da “janela principal”, clique sobre o sinal + ao lado do nome do projeto; deverá aparecer o nome do arquivo que foi inserido.
 - d) Clique duas vezes sobre o nome do arquivo que foi inserido
 - e) Na janela “Microsoft Developer Studio”, clique em YES
 - f) O arquivo inserido está pronto para ser editado no lado direito da “janela principal”

8.4 PROGRAMA8a.f90

- 1) Dentro do espaço de edição de programas do Fortran, escreva o programa abaixo (para o programa funcionar, basta digitar as linhas em vermelho, as demais são comentários que você deve ler e entender):

```
! -----
! programa8a.f90
! -----
! Objetivo: definir dois vetores com dimensão alocável, com elementos do
!           tipo real, realizar operações aritméticas com eles e
!           utilizar o comando SELECT CASE
! -----
! Parte do programa onde são incluídas bibliotecas (módulos) do Fortran

use msflib
! Inclui no programa a biblioteca do Fortran que permite usar os comandos
! "systemqq" e "setexitqq"

use msfwin
! Inclui no programa a biblioteca do Fortran que permite usar o
! comando "setwindowtext"

! -----
! Parte do programa onde são definidas as variáveis do programa

implicit none
! Ao se usar este comando, todas as variáveis do programa têm que ser definidas
! explicitamente com os comandos real, character, integer e logical. Assim,
! são evitados erros de lógica no programa, captados ao se fazer a compilação.

logical dos
! Declaração de variável do tipo "lógica" para usar o comando "systemqq"

logical janela
! Declaração de variável do tipo "lógica" para usar o comando "setwindowtext"

integer*4 fecha
! Declaração de variável do tipo "inteiro" para usar o comando "setexitqq"

real*8,dimension(:),allocatable :: vetor_a, vetor_b
! Declaração das variáveis chamadas "vetor_a" e "vetor_b" com as seguintes
! características:
! (1) comando REAL*8 as define como do tipo real dupla;
! (2) comando DIMENSION(:) as define com dimensão alocável ou aberta, conforme o
!     comando ALLOCATABLE, isto é, suas dimensões serão definidas dentro do
!     programa em função de alguma outra variável.

integer elemento
! Número da componente ou elemento dos vetores

integer quantidade_elementos
! Número de componentes ou elementos dos vetores

integer tipo_de_calculo
! Tipo de cálculo a ser realizado com o vetor_a para gerar o vetor_b

! -----
janela = setwindowtext(gethwndqq(qwin$framewindow), &
    "Programa8a.f90 - DOIS VETORES COM NÚMERO INDEFINIDO DE ELEMENTOS"C)
! Esta instrução define o nome da janela do programa.
! O nome deve ficar entre aspas, conforme o exemplo acima.

! -----
! Parte do programa onde são lidos os dados

dos = systemqq('Notepad dados8a.txt')
! Mostra o conteúdo do arquivo "dados8a.txt" com o programa Notepad
```

```
open(7,file='dados8a.txt')
```

```
! Define o número "7" como uma referência ao arquivo "dados8a.txt"
```

```
read(7,*) quantidade_elementos
```

```
! O comando READ é usado para ler a quantidade de elementos dos vetores no  
! arquivo 7, que corresponde ao arquivo "dados8a.txt", conforme definido no  
! comando OPEN acima.
```

```
allocate ( vetor_a(quantidade_elementos), vetor_b(quantidade_elementos) )
```

```
! Com este comando ALLOCATE define-se quantos elementos as variáveis "vetor_a" e  
! "vetor_b" terão. Note que esta definição ocorre em função da variável  
! "quantidade_elementos", que é um dado do programa. Portanto, a alocação de  
! memória para as variáveis "vetor_a" e "vetor_b" não é pré-definida, ela ocorre  
! em função de dados do programa.
```

```
do elemento = 1, quantidade_elementos
```

```
! Este ciclo DO inicia com o valor 1 e termina com o valor da variável  
! "quantidade_elementos", tendo como passo o valor unitário (por não ter sido  
! especificado).  
! É um ciclo do tipo progressivo, definido por variáveis inteiras.  
! A variável "elemento" começará com o valor 1.  
! Quando a execução do programa atingir o comando END DO, haverá um retorno ao  
! comando DO e o valor da variável "elemento" passará a ser a soma da variável  
! "elemento" com "passo". Se o novo valor da variável "elemento" for menor ou  
! igual à variável "quantidade_elementos", o programa voltará a executar o  
! ciclo DO a END DO. Caso contrário, a execução do programa prosseguirá a partir  
! do comando END DO.
```

```
read(7,*) vetor_a(elemento)
```

```
! O comando READ é usado para ler cada "elemento" da variável "vetor_a" do  
! arquivo 7, que corresponde ao arquivo "dados8a.txt", conforme definido no  
! comando OPEN acima.  
! Como há um asterisco, o conteúdo de cada variável é lido sem qualquer  
! formato, e apenas uma variável por linha no arquivo "dados8a.txt".  
! Note que só há um comando READ. Mas como ele está dentro de um ciclo, ele  
! será suficiente para ler todos os elementos do vetor.
```

```
end do
```

```
! Encerra o ciclo DO
```

```
read(7,*) tipo_de_calculo
```

```
! O comando READ é usado para ler o tipo de cálculo que será realizado com o  
! "vetor_a" para gerar o "vetor_b". A leitura é feita no arquivo 7, que  
! corresponde ao arquivo "dados8a.txt", conforme definido no comando OPEN acima.
```

```
close(7)
```

```
! O comando CLOSE encerra o uso do arquivo 7
```

```
! -----
```

```
! Parte do programa onde são escritos os dados no arquivo de resultados
```

```
open(10,file='resultados8a.txt')
```

```
! Define o número "10" como uma referência ao arquivo "resultados8a.txt"
```

```
write(10,20)
```

```
! Escreve um comentário no arquivo 10 com o formato especificado pelo número 20.
```

```
do elemento = 1, quantidade_elementos
```

```
write(10,30) elemento, vetor_a(elemento)
```

```
! Escreve os valores das variáveis "elemento" e "vetor_a" no arquivo "10" com  
! o formato especificado pelo número "30".
```

```
end do
```

```
! -----
```

```
! Parte do programa onde são realizadas operações matemáticas com o "vetor_a"  
! para obter o "vetor_b" em função do "tipo_de_calculo" definido nos dados
```

select case (tipo_de_calculo)

! O comando SELECT CASE é usado para escolher uma opção entre várias e executar apenas o bloco de comandos da opção escolhida.
! O bloco pode conter uma única linha de programa ou várias.
! Para usar o comando SELECT CASE, além dele, deve-se colocar entre parênteses uma variável que será usada para definir uma entre as várias opções. Neste exemplo, a variável de escolha é "tipo_de_calculo", que é do tipo inteiro.
! Mas também pode-se usar variáveis do tipo character.
! Veja abaixo que cada opção realiza cálculos diferentes.

case (1)**vetor_b = vetor_a + 10**

! A opção 1 adiciona o valor 10 a cada elemento do "vetor_a" e atribui o resultado a cada elemento do "vetor_b"

write(10,101)**case (2)****vetor_b = vetor_a - 10**

! A opção 2 subtrai o valor 10 de cada elemento do "vetor_a" e atribui o resultado a cada elemento do "vetor_b"

write(10,102)**case (3)****vetor_b = 2 * vetor_a**

! A opção 3 multiplica por 2 cada elemento do "vetor_a" e atribui o resultado a cada elemento do "vetor_b"

write(10,103)**case (4)****vetor_b = vetor_a / 5**

! A opção 4 divide por 5 cada elemento do "vetor_a" e atribui o resultado a cada elemento do "vetor_b"

write(10,104)**case (5)****vetor_b = vetor_a ** 2**

! A opção 5 eleva ao quadrado cada elemento do "vetor_a" e atribui o resultado a cada elemento do "vetor_b"

write(10,105)**case default****write(10,40)**

! Se nenhuma das opções pré-definidas for escolhida, o programa executa a opção CASE DEFAULT.

! Neste exemplo, se não forem escolhidos os números 1, 2, 3, 4 ou 5, o CASE DEFAULT será executado, escrevendo um aviso no arquivo 10 que é definido pelo formato número 40.

end select

! O comando END SELECT é usado para definir o fim do comando SELECT CASE

! -----

! Parte do programa onde são escritos os valores do "vetor_b" no arquivo de resultados

write(10,50)

! Escreve um comentário no arquivo 10 com o formato especificado pelo número 50.

do elemento = 1, quantidade_elementos**write(10,60) elemento, vetor_b(elemento)**

! Escreve os valores das variáveis "elemento" e "vetor_b" no arquivo "10" com o formato especificado pelo número "60".

end do

! -----

close(10)

! O comando CLOSE encerra o uso do arquivo 10

```

dos = systemqq('Notepad resultados8a.txt')
! Mostra o conteúdo do arquivo "resultados8a.txt" com o programa Notepad

fecha = setexitqq(qwin$exitnopersist)
! A instrução acima fecha automaticamente a janela do programa ao final de sua
! execução

! -----

! Parte do programa onde são definidos os formatos utilizados

20 format('***** DADOS - VETOR A *****')

30 format( 3x, 'vetor_a(', i1, ') =', 1pe15.5 )

40 format(1/, 5x, 'VOCÊ NÃO ESCOLHEU UMA OPÇÃO DAS EXISTENTES', 1/, &
          5x, 'TENTE NOVAMENTE')

50 format('***** RESULTADOS - VETOR B *****')

60 format( 3x, 'vetor_b(', i1, ') =', 1pe15.5 )

101 format('Você escolheu a opção 1: adiciona 10')
102 format('Você escolheu a opção 2: subtrai 10')
103 format('Você escolheu a opção 3: multiplica por 2')
104 format('Você escolheu a opção 4: divide por 5')
105 format('Você escolheu a opção 5: eleva ao quadrado')

! -----

end
! O comando END indica o final do programa

```

- 2) Depois de digitar o programa, grave-o da seguinte forma: File, Save
- 3) Compile o programa executando: Build, Compile
- 4) Gere o executável do programa com Build, Build
- 5) Antes de executar este programa, você precisa criar o arquivo “dados8a.txt”, inserindo nele a quantidade de elementos do vetor_a e os valores dos seus elementos, além do tipo de cálculo desejado. Para fazer isso, execute o seguinte:
 - a) Start, Programs, Accessories, Notepad
 - b) Dentro do espaço de edição do Notepad, edite por exemplo:

```

4      quantidade de elementos do vetor
1.0    elemento 1
5.0    elemento 2
-3.0   elemento 3
2.0    elemento 4
3      tipo de cálculo

```

Descrição dos tipos de cálculo disponíveis:

```

1: adiciona o valor 10 a cada elemento
2: subtrai o valor 10 de cada elemento
3: multiplica por 2 cada elemento
4: divide por 5 cada elemento
5: eleva ao quadrado cada elemento

```
 - c) File, Save
 - d) Na janela “Save As”, em “File Name”, coloque “dados8a.txt”; em “Save in:”, selecione o diretório do seu projeto. Em seguida, clique sobre o botão “Save”. Finalmente, selecione File, Exit.
- 6) Execute o programa com Build, Execute. O resultado deverá ser:
 - a) A abertura de uma janela do DOS.

- b) A abertura de uma janela do WINDOWS que terá como título “Programa8a.f90 - DOIS VETORES COM NÚMERO INDEFINIDO DE ELEMENTOS”, conforme foi especificado dentro do programa8a.f90.
- c) A abertura do arquivo “dados8a.txt” com o programa Notepad. Se o arquivo não existir no diretório do projeto, ocorrerá um erro. Se você quiser, altere o conteúdo das variáveis no arquivo “dados8a.txt”. Depois, grave o novo valor com File, Save. Finalmente, feche este arquivo com File, Exit.
- d) Em seguida, ocorrerá a abertura do arquivo “resultados8a.txt” com o programa Notepad. O conteúdo deverá ser:

```

***** DADOS - VETOR A *****
vetor_a(1) = 1.00000E+00
vetor_a(2) = 5.00000E+00
vetor_a(3) = -3.00000E+00
vetor_a(4) = 2.00000E+00
Você escolheu a opção 3: multiplica por 2
***** RESULTADOS - VETOR B *****
vetor_b(1) = 2.00000E+00
vetor_b(2) = 1.00000E+01
vetor_b(3) = -6.00000E+00
vetor_b(4) = 4.00000E+00

```

- e) Até entender, compare o conteúdo de cada linha do arquivo “resultados8a.txt” com os comandos do programa8a.f90 e seus respectivos formatos de edição.
- f) Feche o arquivo “resultados8a.txt” com File, Exit.
- g) Em seguida, tanto a janela do DOS quanto a do WINDOWS serão fechadas automaticamente.
- 7) Execute novamente o programa com Build, Execute. Utilize as outras opções (1, 2, 4 e 5) de cálculo e altere o número de elementos do vetor_a, nos dados. Analise os resultados.
- 8) Execute novamente o programa com Build, Execute. No arquivo de dados, utilize para a variável “tipo de cálculo” algum valor diferente das opções disponíveis (1 a 5). Tente, por exemplo, 0 ou 10. Analise os resultados.

8.5 PROGRAMA8b.f90

- 1) Clique dentro do espaço de edição do seu “programa8a.f90”. Depois execute: File, Close. Clique sobre o nome do “programa8a.f90”, no lado esquerdo da janela principal do Fortran. Em seguida, execute: Edit, Cut. Com isso, o “programa8a.f90” será eliminado do seu projeto, mas o arquivo continuará a existir dentro do diretório do seu projeto, podendo ser incluído nele quando desejado.
- 2) Seguindo os passos da seção 8.3 deste tutorial, insira no projeto “programa08” o arquivo “programa8b.f90”.
- 3) Dentro do espaço de edição de programas do Fortran, escreva o programa abaixo (para o programa funcionar, basta digitar as linhas em vermelho, as demais são comentários que você deve ler e entender):

```

! -----
! programa8b.f90
! -----
! Objetivo: aplicar diversas funções matemáticas pré-definidas no Fortran junto
!           com vetores e o comando SELECT CASE
! -----
! Parte do programa onde são incluídas bibliotecas (módulos) do Fortran
use msflib
use msfwin
! -----
! Parte do programa onde são definidas as variáveis do programa
implicit none

```

```
logical dos
logical janela

integer*4 fecha

real*8,dimension(19) :: x, y
! Declaração das variáveis chamadas "x" e "y" com as seguintes
! características:
! (1) comando REAL*8 as define como do tipo real dupla;
! (2) comando DIMENSION(19) as define com dimensão fixa de 19 elementos cada

integer elemento
integer tipo_de_calculo

real*8 pi
! Constante matemática

! -----

janela = setwindowtext(gethwnDqq(qwin$framewindow), &
    "Programa8b.f90 - FUNÇÕES"C)

! -----

! Parte do programa onde são lidos os dados

dos = systemqq('Notepad dados8b.txt')

open(7,file='dados8b.txt')

read(7,*) tipo_de_calculo

close(7)

! -----

! Parte do programa onde é gerado o vetor "x" básico

do elemento = 1, 19

    x(elemento) = (elemento - 1) * 20
    ! Gera o vetor "x" a partir da variável "elemento".
    ! "x" variará de 0 a 360, com passo de 20

end do

! -----

! Parte do programa onde são realizadas operações matemáticas com o vetor "x"
! para obter o vetor "y" em função do "tipo_de_calculo" definido nos dados

open(10,file='resultados8b.txt')

pi = acos(-1.0)
! Cálculo da constante matemática "pi" utilizando a função arco cosseno (acos)

select case ( tipo_de_calculo )

! Em cada opção abaixo, é realizada alguma operação matemática com cada elemento
! do vetor x e atribuído o resultado a cada elemento do vetor y

    case ( 1 )
        x = x - 180
        y = abs(x)
        ! Opção 1: módulo (abs)
        write(10,101)

    case ( 2 )
        x = x - 180
        y = exp(x)
        ! Opção 2: exponencial (exp)
```

```
write(10,102)

case ( 3 )
  x = x + 1.0e-10
  y = log(x)
  ! Opção 3: logaritmo natural (log)
  write(10,103)

case ( 4 )
  x = x + 1.0e-10
  y = log10(x)
  ! Opção 4: logaritmo decimal (log10)
  write(10,104)

case ( 5 )
  x = x * pi / 180
  y = sin(x)
  ! Opção 5: seno (sin)
  write(10,105)

case ( 6 )
  x = (x / 180) - 1.0
  y = asin(x)
  ! Opção 6: arco seno (asin)
  write(10,106)

case ( 7 )
  x = x / 360
  y = sinh(x)
  ! Opção 7: seno hiperbólico (sinh)
  write(10,107)

case ( 8 )
  x = x * pi / 180
  y = cos(x)
  ! Opção 8: cosseno (cos)
  write(10,108)

case ( 9 )
  x = (x / 180) - 1.0
  y = acos(x)
  ! Opção 9: arco cosseno (acos)
  write(10,109)

case ( 10 )
  x = x / 360
  y = cosh(x)
  ! Opção 10: cosseno hiperbólico (cosh)
  write(10,110)

case ( 11 )
  x = x * pi / 180
  y = tan(x)
  ! Opção 11: tangente (tan)
  write(10,111)

case ( 12 )
  x = (x / 180) - 1.0
  y = atan(x)
  ! Opção 12: arco tangente (atan)
  write(10,112)

case ( 13 )
  x = x / 360
  y = tanh(x)
  ! Opção 13: tangente hiperbólica (tanh)
  write(10,113)

case default
  write(10,40)
```



```
! Se nenhuma das opções pré-definidas for escolhida, o programa executa
! a opção CASE DEFAULT.
! Neste exemplo, se não forem escolhidos os números 1 a 13,
! o CASE DEFAULT será executado, escrevendo um aviso no arquivo 10 que é
! definido pelo formato número 40.
```

```
end select
```

```
! -----
```

```
! Parte do programa onde são escritos os valores dos vetores "x" e "y"
```

```
write(10,20)
```

```
do elemento = 1, 19
```

```
    write(10,60) x(elemento), y(elemento)
```

```
    ! Escreve os valores dos vetores "x" e "y" no arquivo "10" com
    ! o formato especificado pelo número "60".
```

```
end do
```

```
! -----
```

```
close(10)
```

```
dos = systemqq('Notepad resultados8b.txt')
```

```
fecha = setexitqq(qwin$exitnopersist)
```

```
! -----
```

```
! Parte do programa onde são definidos os formatos utilizados
```

```
20 format(1/, t5, 'X', t20, 'Y', 1/)
```

```
40 format(1/, 5x, 'VOCÊ NÃO ESCOLHEU UMA OPÇÃO DAS EXISTENTES', 1/, &
    5x, 'TENDE NOVAMENTE')
```

```
60 format( 2 ( 1pe15.5 ) )
```

```
101 format('Opção 1: y = MÓDULO de x')
```

```
102 format('Opção 2: y = EXPONENCIAL de x')
```

```
103 format('Opção 3: y = LOGARITMO NATURAL de x')
```

```
104 format('Opção 4: y = LOGARITMO DECIMAL de x')
```

```
105 format('Opção 5: y = SENO de x em radianos')
```

```
106 format('Opção 6: y (radianos) = ARCO SENO de x')
```

```
107 format('Opção 7: y = SENO HIPERBÓLICO de x')
```

```
108 format('Opção 8: y = COSSENO de x em radianos')
```

```
109 format('Opção 9: y (radianos) = ARCO COSSENO de x')
```

```
110 format('Opção 10: y = COSSENO HIPERBÓLICO de x')
```

```
111 format('Opção 11: y = TANGENTE de x em radianos')
```

```
112 format('Opção 12: y (radianos) = ARCO TANGENTE de x')
```

```
113 format('Opção 13: y = TANGENTE HIPERBÓLICA de x')
```

```
! -----
```

```
end
```

- 4) Depois de digitar o programa, grave-o da seguinte forma: File, Save
- 5) Compile o programa executando: Build, Compile
- 6) Gere o executável do programa com Build, Build
- 7) Antes de executar este programa, você precisa criar o arquivo “dados8b.txt”, inserindo nele o tipo de cálculo desejado. Para fazer isso, execute o seguinte:
 - a) Start, Programs, Accessories, Notepad
 - b) Dentro do espaço de edição do Notepad, edite por exemplo:

4 tipo de cálculo

Descrição dos tipos de cálculo disponíveis:

- 1: módulo
 2: exponencial
 3: logaritmo natural
 4: logaritmo decimal
 5: seno
 6: arco seno
 7: seno hiperbólico
 8: cosseno
 9: arco cosseno
 10: cosseno hiperbólico
 11: tangente
 12: arco tangente
 13: tangente hiperbólica

- c) File, Save
 - d) Na janela “Save As”, em “File Name”, coloque “dados8b.txt”; em “Save in:”, selecione o diretório do seu projeto. Em seguida, clique sobre o botão “Save”. Finalmente, selecione File, Exit.
- 8) Execute o programa com Build, Execute. O resultado deverá ser:
 - a) A abertura de uma janela do DOS.
 - b) A abertura de uma janela do WINDOWS que terá como título “Programa8b.f90 - FUNÇÕES”, conforme foi especificado dentro do programa8b.f90.
 - c) A abertura do arquivo “dados8b.txt” com o programa Notepad. Se o arquivo não existir no diretório do projeto, ocorrerá um erro. Se você quiser, altere o conteúdo das variáveis no arquivo “dados8b.txt”. Depois, grave o novo valor com File, Save. Finalmente, feche este arquivo com File, Exit.
 - d) Em seguida, ocorrerá a abertura do arquivo “resultados8b.txt” com o programa Notepad. O conteúdo deverá ser:

Opção 4: $y = \text{LOGARITMO DECIMAL de } x$

X	Y
1.00000E-10	-1.00000E+01
2.00000E+01	1.30103E+00
4.00000E+01	1.60206E+00
6.00000E+01	1.77815E+00
8.00000E+01	1.90309E+00
1.00000E+02	2.00000E+00
1.20000E+02	2.07918E+00
1.40000E+02	2.14613E+00
1.60000E+02	2.20412E+00
1.80000E+02	2.25527E+00
2.00000E+02	2.30103E+00
2.20000E+02	2.34242E+00
2.40000E+02	2.38021E+00
2.60000E+02	2.41497E+00
2.80000E+02	2.44716E+00
3.00000E+02	2.47712E+00
3.20000E+02	2.50515E+00
3.40000E+02	2.53148E+00
3.60000E+02	2.55630E+00

- e) Até entender, compare o conteúdo de cada linha do arquivo “resultados8b.txt” com os comandos do programa8b.f90 e seus respectivos formatos de edição.
- f) Feche o arquivo “resultados8b.txt” com File, Exit.
- g) Em seguida, tanto a janela do DOS quanto a do WINDOWS serão fechadas automaticamente.

- 9) Execute novamente o programa com Build, Execute. Utilize as outras opções (1, 2, 3, e 5 a 13) de cálculo. Analise os resultados.
- 10) Execute novamente o programa com Build, Execute. No arquivo de dados, utilize para a variável “tipo de cálculo” algum valor diferente das opções disponíveis (1 a 13). Tente, por exemplo, 0 ou 20. Analise os resultados.

8.6 ENCERRANDO A AULA

- 1) Dentro da “janela principal” do Fortran, feche o seu projeto executando o seguinte: File, Close Workspace
- 2) Na janela “Microsoft Developer Studio” clique sobre o botão “Yes”.
- 3) Para fechar o programa Fortran, basta executar: File, Exit
- 4) Para fechar o Windows, execute: Start, Shut Down
- 5) Na janela “Shut Down Windows”, escolha a opção “Close all programs and log on as a different user?”. Depois clique no botão “Yes”.
- 6) Depois que o computador fechar o Windows, por favor, apenas desligue o monitor do computador.

LISTA DE EXERCÍCIOS DA 8ª AULA

Exercício 8.1

- 1) Implemente um programa em Fortran para calcular a altitude de um projétil lançado verticalmente para cima. Utilize o seguinte algoritmo:
 - a) Ler do arquivo “dados8.txt”:
 - yo = a altitude inicial do projétil (metros); use = 0.0;
 - vo = a velocidade inicial do projétil (metros/segundo); use = 50.0;
 - g = a aceleração gravitacional (metros/segundo²); use = 9.8;
 - dt = o intervalo (passo) de tempo (segundos); use = 0.5;
 - b) Calcular a altitude (y) do projétil, para instantes de tempo consecutivos cujo intervalo entre eles é igual ao passo de tempo, até que a altitude volte a yo.
 - c) Escrever no arquivo “resultados8.txt”:
 - Todos os valores lidos do arquivo “dados8.txt”; e
 - Uma lista contendo cada instante de tempo e a respectiva altitude do projétil.
- 2) Compile o programa
- 3) Gere o executável do programa
- 4) Execute o programa
- 5) Entregue na próxima aula:
 - Uma impressão do seu programa fonte em Fortran;
 - Uma impressão do arquivo “resultados8.txt”; e
 - Um gráfico da altitude versus tempo do projétil.

Exercício 8.2

- 1) Acesse o site da disciplina no endereço:
ftp://ftp.demec.ufpr.br/Disciplinas/Tm102/Marchi/semestre_2003_2/
- 2) Estude as páginas 29 e 30 do capítulo 7 da “apostila_logica_programacao.pdf”.