



UNIVERSIDADE FEDERAL DO PARANÁ

CURSO DE ENGENHARIA MECÂNICA

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

## 4ª aula

### INSTRUÇÕES

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

### 4.1 OBJETIVOS DA AULA

- 1) Definir variáveis do tipo INTEIRO, CHARACTER e REAL
- 2) Definir formatos para escrevê-las

### 4.2 CRIANDO UM NOVO PROJETO DO TIPO CONSOLE APPLICATION

- 1) Para iniciar o uso do Fortran, execute: Start, Programs, Fortran PowerStation 4.0, Microsoft Developer Studio
- 2) Na “janela principal” do Fortran: File, New
- 3) Na janela “New”: selecione “Project Workspace” e clique sobre o botão “OK”
- 4) Na janela “New Project Workspace”, execute o seguinte:
  - a) Type: selecione “Console Application”
  - b) Location: use o botão BROWSE para indicar um diretório onde você quer criar o seu projeto. Se você já tem conta própria na rede do DEMEC, selecione o seu diretório no “drive” Server1, diretório (pasta) “alunos\_2003\_2”.
  - c) Name: indique o nome do projeto. Para esta aula use “projeto04”. 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

### 4.3 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 “programa04a.f90”. Sempre use ‘f90’ como extensão dos nomes dos arquivos. Ele indica que o programa está ou será escrito na linguagem Fortran 90. Cada arquivo com a extensão “f90” é chamado de “programa fonte”.
  - b) Clique em “OK”
  - c) Na janela “Microsoft Developer Studio”, clique em YES
  - d) 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.

- e) Clique duas vezes sobre o nome do arquivo que foi inserido
- f) Na janela "Microsoft Developer Studio", clique em YES
- g) O arquivo inserido está pronto para ser editado no lado direito da "janela principal"

#### 4.4 PROGRAMA04a.f90

- 1) Dentro do espaço de edição de programas do Fortran, digite apenas as linhas abaixo que não iniciam pelo ponto de exclamação (!), mas leia e estude os comentários:

```
! PROGRAMA04a.f90

! Objetivo: definição de variáveis do tipo inteiro e seus formatos

! O comando "INTEGER" é usado para definir variáveis do tipo INTEIRO

! Define "a", "b" e "c" como variáveis do tipo inteiro
integer a, b, c

! Define "d" como variável do tipo inteiro
integer d

! Escreve na janela do DOS (devido ao número 6 no comando WRITE)
! dois comentários que estão entre aspas (") ou apóstrofes (')
write(6,*) "Variaveis do tipo inteiro"
write(6,*) 'Entre com os dados de "a", "b" e "c"'

! Utiliza um único comando READ para ler três variáveis, "a", "b" e "c", através
! do teclado (devido ao número 5 no comando READ)
read(5,*) a, b, c

! Soma os valores das variáveis "a", "b" e "c" e atribui o resultado à variável "d"
d = a + b + c

! Define o número "11" como uma referência ao arquivo "saida4a.txt"
open(11,file='saida4a.txt')

! Escreve no arquivo 11 o valor da variável "a" sem usar qualquer formato de edição
! especificado pelo programador; não existe formato porque há um asterisco no comando
! WRITE; neste caso, a variável "a" será escrita com um formato padrão do Fortran
write(11,*) a

! Aqui tem-se o primeiro exemplo de uso do comando WRITE com a especificação de um
! FORMATO de edição. Em geral, dois números são usados com o comando WRITE: o
! primeiro, define um arquivo, no caso o número 11; o segundo, define um FORMATO de
! edição, no caso o número 8. Portanto, o comando abaixo irá escrever o valor da
! variável "a" no arquivo "11" com o formato especificado pelo número "8".
write(11,8) a

! O comando FORMAT é usado para especificar um formato de edição para escrever ou ler
! uma ou diversas variáveis.
! Os formatos são especificados dentro de parênteses.
! Cada comando FORMAT inicia-se numa linha diferente do arquivo especificado, junto à
! margem esquerda.
! Para variáveis do tipo inteiro, usa-se a letra "i" seguida de um número que
! especifica a quantidade máxima de colunas para a variável.
! O valor da variável é escrito da direita para a esquerda.
! Portanto, o formato abaixo usa duas colunas para uma variável do tipo inteiro.
8 format(i2)

! O comando abaixo irá escrever o valor da variável "a"
! no arquivo "11" com o formato especificado pelo número "12".
write(11,12) a
```

```
! O formato abaixo usa seis colunas para uma variável do tipo inteiro.
12 format(i6)
```

```
! O comando abaixo irá escrever o valor da variável "a"
! no arquivo "11" com o formato especificado pelo número "22".
write(11,22) a
```

```
! Dentro do comando FORMAT pode-se especificar vários formatos
! de edição, incluindo comentários entre aspas ou apóstrofos, desde
! que todos estejam dentro de um único par de parênteses e cada
! formato separado dos demais por vírgula.
! Por exemplo, o comando FORMAT abaixo contém quatro formatos de
! edição.
! O formato "10x" significa que o cursor deixará 10 espaços em branco
! a partir da margem esquerda do arquivo. O número 10 pode ser
! substituído por qualquer outro valor.
! O formato 'valor de a =' , irá reproduzir exatamente o que está entre
! os apóstrofos. Também pode-se usar aspas.
! O formato "i6" usa seis colunas para uma variável do tipo inteiro.
! O formato "3/" fará o cursor descer três linhas no arquivo. O número
! 3 pode ser substituído por qualquer outro valor.
22 format(10x,'valor de a =',i6, 3/)
```

```
! O comando abaixo irá escrever o valor da variável "a"
! no arquivo "11" com o formato especificado pelo número "33".
write(11,33) a
```

```
! O formato abaixo faz exatamente o mesmo que o formato 22, acima,
! mas utiliza aspas para escrever o comentário.
33 format(10x,"valor de a =",i6, 3/)
```

```
! O comando abaixo irá escrever os valores das variáveis "a", "b" e
! "c" no arquivo "11" com o formato especificado pelo número "23".
write(11,23) a, b, c
```

```
! O comando FORMAT, a seguir, contém 11 formatos de edição
! utilizando os quatro tipos já vistos: espaços em branco (x);
! comentários; inteiros (i) e linhas em branco (/).
! Cada variável do comando WRITE está associada a um formato "i6"
! do comando FORMAT, na mesma seqüência, da esquerda para a direita.
23 format(10x, 'a =', i6, 5x, 'b =', i6, 2/, 5x, 'c =', i6, 1/)
```

```
! O comando abaixo irá escrever os valores das variáveis "a", "b" e
! "c" no arquivo "11" com o formato especificado pelo número "24".
write(11,24) a, b, c
```

```
! O formato 24, abaixo, faz exatamente o mesmo que o formato 23,
! acima. A diferença é que, agora, os formatos são escritos ao longo
! de três linhas do programa. Deve-se usar essa forma de edição
! quando um comando é muito longo. Neste caso, usa-se o símbolo "&",
! ao final de cada linha, para informar ao programa que ela continua
! na linha seguinte.
24 format(10x, 'a =', i6,           &
          5x, 'b =', i6, 2/,       &
          5x, 'c =', i6, 1/ )
```

```
! O comando abaixo irá escrever os valores das variáveis "a", "b",
! "c" e "d" no arquivo "11" com o formato especificado pelo número
! "25".
write(11,25) a, b, c, d
```

```
! O comando FORMAT também pode ser usado para repetir formatos.
! Isso é feito usando-se um número para indicar quantas vezes o
! formato, que está entre parênteses, deve ser repetido. No exemplo
! abaixo o formato "i8" é repetido 3 vezes.
25 format ( 3 (i8), 5x, 'd =', i8 )
```

**! O comando CLOSE encerra o uso do arquivo 11**  
**close(11)**

**! O comando END encerra o programa**  
**end**

- 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) Execute o programa com Build, Execute.
- 6) Entre com os valores das variáveis “a”, “b” e “c”, por exemplo, 1, 2 e 3. Isso pode ser feito de várias formas: digitando na mesma linha três valores separados por vírgula ou por espaços em branco; digitando cada valor numa linha e clicando na tecla “Enter”, uma ou várias vezes; ou combinações das duas formas anteriores. Por exemplo:

**Variaveis do tipo inteiro**

**Entre com os dados de "a", "b" e "c"**

**1**

**2,3**

**Press any key to continue**

- 7) A execução do programa04a.f90 deve ter criado um arquivo com o nome “saida4a.txt” dentro do diretório do seu projeto. Você pode ver isso com o programa “Windows NT Explorer”, que pode ser acessado através de: Start, Programs, Windows NT Explorer. Depois, acesse o diretório do seu projeto.
- 8) Para ver o conteúdo do arquivo “saida4a.txt”, que deve estar dentro do diretório do seu projeto, utilize o programa Notepad. Acesse este programa da seguinte forma:
  - a) Start, Programs, Accessories, Notepad.
  - b) File, Open
  - c) Na janela “Open”, em “Look in:”, selecione o diretório do seu projeto. Em seguida, clique sobre o arquivo “saida4a”. Finalmente, clique sobre o botão “Open”. Você verá que o conteúdo deste arquivo é do tipo texto. Você também consegue vê-lo com o programa Word.
  - d) O conteúdo do arquivo “saida4a.txt” deverá ser o seguinte:

**1**

**1**

**1**

**valor de a = 1**

**valor de a = 1**

**a = 1 b = 2**

**c = 3**

**a = 1 b = 2**

**c = 3**

**1 2 3 d = 6**

- 9) Até entender, compare o conteúdo de cada linha do arquivo “saida4a.txt” com os formatos de edição empregados em cada comando WRITE no programa04a.f90.
- 10) Execute novamente o programa com Build, Execute. Entre com novos valores para as variáveis “a”, “b” e “c”, por exemplo, 111, 222 e 333. Na janela do DOS, o resultado da execução do programa deverá ser:

```
Variaveis do tipo inteiro
Entre com os dados de "a", "b" e "c"
111, 222 333
Press any key to continue
```

- 11) O novo conteúdo do arquivo “saida4a.txt” deverá ser o seguinte:

```
111
**
111
    valor de a = 111

    valor de a = 111

    a = 111    b = 222
c = 333
    a = 111    b = 222
c = 333
111    222    333    d = 666
```

- 12) Note que o formato (i2), usado no segundo comando WRITE do programa, não conseguiu escrever corretamente o valor da variável “a”, que foi representado por dois asteriscos. Isso ocorre sempre que o formato é incompatível com o valor da variável. Portanto, deve-se sempre usar um formato superdimensionado para evitar esse tipo de problema.
- 13) Altere os formatos usados no programa04a.f90 e verifique o que ocorre.

#### **4.5 PROGRAMA04b.f90**

- 1) Clique sobre o nome do “programa04a.f90”, no lado esquerda da janela principal do Fortran. Em seguida, execute: Edit, Cut. Com isso, o programa04a.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 4.3 deste tutorial, insira no projeto04 o “programa04b.f90”.
- 3) Dentro do espaço de edição de programas do Fortran, digite apenas as linhas abaixo que não iniciam pelo ponto de exclamação (!), mas leia e estude os comentários:

```
! PROGRAMA04b.f90
```

```
! Objetivo: definição de variáveis do tipo character e seus formatos
```

```
! O comando "CHARACTER" é usado para definir variáveis do
! tipo CHARACTER, que inclui letras, números e símbolos

! Define "b" como variável do tipo character
character b

! Define "a" e "c" como variáveis do tipo character, cada uma podendo
! ter até 10 caracteres alfanuméricos
character*10 a, c

! Define "d" como variável do tipo character, podendo
! ter até 30 caracteres alfanuméricos
character*30 d

! Escreve na janela do DOS (devido ao número 6 no comando WRITE)
! dois comentários que estão entre aspas (") ou apóstrofos (')
write(6,*) "variaveis do tipo character"
write(6,*) 'Entre com o conteúdo de "a", "b" e "c"'

! Utiliza um único comando READ para ler três variáveis,
! "a", "b" e "c", através do teclado (devido ao número 5 no
! comando READ).
! O conteúdo de cada variável do tipo character deve estar entre
! aspas ou apóstrofos.
read(5,*) a, b, c

! Quando se usa duas barras (//) entre duas variáveis do tipo
! character, o resultado é uma outra variável que junta o conteúdo das
! duas na seqüência da esquerda para a direita.
! Portanto, na linha abaixo, junta-se o conteúdo da variável "a" à
! variável "b", ao comentário "teste ", e ao conteúdo da variável "c";
! e o resultado é atribuído à variável "d"
d = a // b // 'teste ' // c

! Define o número "11" como uma referência ao arquivo
! "saida4b.txt"
open(11,file='saida4b.txt')

! Escreve no arquivo 11 o conteúdo da variável "a" sem usar
! qualquer formato de edição especificado pelo programador;
! não existe formato porque há um asterisco no comando WRITE;
! neste caso, a variável "a" será escrita com um formato padrão
! do Fortran
write(11,*) a

! O comando abaixo irá escrever o conteúdo da variável "a"
! no arquivo "11" com o formato especificado pelo número "8".
write(11,8) a

! Para variáveis do tipo character, usa-se a letra "a" seguida de um
! número que especifica a quantidade máxima de colunas para a
! variável.
! O conteúdo da variável é escrito da esquerda para a direita.
! Portanto, o formato abaixo usa duas colunas para uma variável
! do tipo character.
8 format(a2)

! O comando abaixo irá escrever o conteúdo da variável "a"
! no arquivo "11" com o formato especificado pelo número "12".
write(11,12) a

! O formato abaixo usa seis colunas para uma variável
! do tipo character.
12 format(a6)

! O comando abaixo irá escrever o conteúdo da variável "a"
```

```

! no arquivo "11" com o formato especificado pelo número "22".
write(11,22) a

! O comando FORMAT abaixo contém quatro formatos de edição.
! O formato "10x" significa que o cursor deixará 10 espaços em branco
! a partir da margem esquerda do arquivo. O número 10 pode ser
! substituído por qualquer outro valor.
! O formato 'conteúdo de a =' irá reproduzir exatamente o que está
! entre os apóstrofos. Também pode-se usar aspas.
! O formato "a10" usa dez colunas para uma variável do tipo caracter.
! O formato "1/" fará o cursor descer uma linha no arquivo. O número
! 1 pode ser substituído por qualquer outro valor.
22 format(10x,'conteúdo de a =',a10, 1/)

! O comando abaixo irá escrever os conteúdos das variáveis "a", "b",
! "c" e "d" no arquivo "11" sem usar formatação, por haver um
! asterisco no comando WRITE.
write(11,*) a, b, c, d

! O comando abaixo irá escrever os conteúdos das variáveis
! "a", "b", "c" e "d" no arquivo "11" com o formato especificado
! pelo número "24".
write(11,24) a, b, c, d

! No comando FORMAT com número 24, abaixo, os formatos são escritos
! ao longo de quatro linhas do programa. Deve-se usar essa forma de
! edição quando um comando é muito longo. Neste caso, usa-se o
! símbolo "&", ao final de cada linha, para informar ao programa que
! ela continua na linha seguinte.
24 format(10x, 'a =', a10,          &
          5x, 'b =', a10, 2/,      &
          5x, 'c =', a10, 1/,      &
          5x, 'd =', a30, 1/ )

! O comando abaixo irá escrever os conteúdos das variáveis
! "a", "b", "c" e "d" no arquivo "11" com o formato especificado
! pelo número "25".
write(11,25) a, b, c, d

! O comando FORMAT também pode ser usado para repetir formatos.
! Isso é feito usando-se um número para indicar quantas vezes o
! formato, que está entre parênteses, deve ser repetido. No exemplo
! abaixo o formato "a10" é repetido 3 vezes, seguido pelo
! formato "a30".
25 format ( 3 (a10), a30 )

! O comando CLOSE encerra o uso do arquivo 11
close(11)

! O comando END encerra o programa
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) Execute o programa com Build, Execute.
- 8) Entre com os conteúdos das variáveis "a", "b" e "c", por exemplo, 'variavel a', 'variavel b', 'variavel c'. Isso pode ser feito de várias formas: digitando na mesma linha o conteúdo das três variáveis separados por vírgula ou por espaços em branco; digitando o conteúdo de cada variável numa linha e clicando na tecla "Enter", uma ou várias vezes; ou combinações das duas formas anteriores. Mas lembre-se que o conteúdo de cada variável deve estar entre aspas ou apóstrofos. Por exemplo:
 

```

variaveis do tipo caracter
Entre com o conteudo de "a", "b" e "c"

```

```
'variavel a' 'variavel b'
  'variavel c'
Press any key to continue
```

9) O conteúdo do arquivo “saida4b.txt” deverá ser o seguinte:

```
variavel a
va
variav
      conteúdo de a =variavel a

variavel avvariavel cvariavel avteste variavel c
      a =variavel a      b =          v

      c =variavel c
      d =variavel avteste variavel c

variavel a      vvariavel cvariavel avteste variavel c
```

- 10) Até entender, compare o conteúdo de cada linha do arquivo “saida4b.txt” com os formatos de edição empregados em cada comando WRITE no programa04b.f90.
- 11) Note que o formato (a2), usado no primeiro comando FORMAT do programa, não conseguiu escrever todo o conteúdo da variável “a”. Foram escritas apenas as duas primeiras colunas da variável “a”, a partir da esquerda. O resto do conteúdo de “a” é desprezado devido ao formato empregado. Portanto, deve-se sempre usar um formato superdimensionado para evitar esse tipo de problema.
- 12) Altere os formatos usados no programa04b.f90 e verifique o que ocorre.

#### 4.6 PROGRAMA04c.f90

- 1) Clique sobre o nome do “programa04b.f90”, no lado esquerda da janela principal do Fortran. Em seguida, execute: Edit, Cut. Com isso, o programa04b.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 4.3 deste tutorial, insira no projeto04 o “programa04c.f90”.
- 3) Dentro do espaço de edição de programas do Fortran, digite apenas as linhas abaixo que não iniciam pelo ponto de exclamação (!), mas leia e estude os comentários:

```
! PROGRAMA04c.f90

! Objetivo: definição de variáveis do tipo real simples
!           e seus formatos

! O comando "REAL" é usado para definir variáveis do tipo REAL

! Define "a" como variável do tipo real simples
real*4 a

! Define "b", "c" e "d" como variáveis do tipo real simples
real*4 b, c, d

! Escreve na janela do DOS (devido ao número 6 no comando WRITE)
! dois comentários que estão entre aspas (") ou apóstrofos (')
```



```
write(6,*) "variaveis do tipo real"
write(6,*) 'Entre com os dados de "a", "b" e "c"'

! Utiliza um único comando READ para ler três variáveis,
! "a", "b" e "c", através do teclado (devido ao número 5 no
! comando READ)
read(5,*) a, b, c

! Soma os valores das variáveis "a", "b" e "c" e atribui o
! resultado à variável "d"
d = a + b + c

! Define o número "11" como uma referência ao arquivo
! "saida4c.txt"
open(11,file='saida4c.txt')

! Escreve no arquivo 11 o valor da variável "a" sem usar
! qualquer formato de edição especificado pelo programador;
! não existe formato porque há um asterisco no comando WRITE;
! neste caso, a variável "a" será escrita com um formato padrão
! do Fortran
write(11,*) a

! Escreve no arquivo 11 o valor da variável "a" seis vezes usando em
! cada vez um formato de edição diferente especificado pelo número 21.
write(11,21) a, a, a, a, a, a

! Para variáveis do tipo real, há vários formatos disponíveis.

! O formato de ponto flutuante é denotado por "fx.y": "x" é um número
! que equivale à quantidade total de colunas usadas para representar
! o valor da variável, incluindo seu sinal e o ponto decimal; "y" é
! outro número que equivale à quantidade de algarismos usados para
! representar a parte decimal do valor da variável.

! O formato exponencial é denotado por "ex.y": "x" é um número
! que equivale à quantidade total de colunas usadas para representar
! o valor da variável, incluindo seu sinal, o ponto decimal, a letra
! "E", o sinal do expoente e dois algarismos para o valor do expoente.
! -1.2E+03 significa -1.2 elevado à potência +3, ou seja, -1200.0
! -1.2E-04 significa -1.2 elevado à potência -4, ou seja, -0.00012

! O formato denotado por "zpex.y" é uma variante de "ex.y", onde
! "z" é um número que equivale à quantidade total de algarismos que se
! deseja antes do ponto decimal, ou seja, a parte inteira do número.

! O formato denotado por "enx.y" é uma variante de "ex.y" chamada de
! notação de engenharia.

! O formato denotado por "esx.y" é uma variante de "ex.y" chamada de
! notação científica.

! Em qualquer formato, o valor da variável é escrito da direita
! para a esquerda.

21 format('"a" no formato f5.2    =', f5.2,    1/,    &
        '"a" no formato f5.0    =', f5.0,    1/,    &
        '"a" no formato e15.3   =', e15.3,   1/,    &
        '"a" no formato 2pe15.3 =', 2pe15.3, 1/,    &
        '"a" no formato en15.3  =', en15.3,  1/,    &
        '"a" no formato es15.3  =', es15.3,  1/      )

! O comando abaixo irá escrever os valores das variáveis "a", "b",
! "c" e "d" no arquivo "11" com o formato especificado pelo número "24".
write(11,24) a, b, c, d

! No comando FORMAT especificado pelo número 24, abaixo, os formatos
```

! são escritos ao longo de quatro linhas do programa. Deve-se usar  
! essa forma de edição quando um comando é muito longo. Neste caso,  
! usa-se o símbolo "&", ao final de cada linha, para informar ao  
! programa que ela continua na linha seguinte.

! O formato "tx" representa uma tabulação, isto é, o cursor é movido  
! para a coluna indicada pelo valor de "x".

```
24 format( t5, 'a =', 1pe15.3,      &
           t25, 'b =', 1pe15.3, 1/,  &
           t25, 'c =', 1pe15.3, 1/,  &
           t25, 'd =', 1pe15.3 )
```

! O comando CLOSE encerra o uso do arquivo 11  
close(11)

! O comando END encerra o programa  
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) Execute o programa com Build, Execute.
- 8) Entre com os conteúdos das variáveis "a", "b" e "c", por exemplo, -1.234567890, 2.222222222, 3.333333333. Isso pode ser feito de várias formas: digitando na mesma linha o conteúdo das três variáveis separados por vírgula ou por espaços em branco; digitando o conteúdo de cada variável numa linha e clicando na tecla "Enter", uma ou várias vezes; ou combinações das duas formas anteriores. Por exemplo:

```
variaveis do tipo real
Entre com os dados de "a", "b" e "c"
-1.234567890
 2.222222222
 3.333333333
Press any key to continue
```

- 9) O conteúdo do arquivo "saida4c.txt" deverá ser o seguinte:

```
-1.234568
"a" no formato f5.2    ==-1.23
"a" no formato f5.0    =  -1.
"a" no formato e15.3   =    -.123E+01
"a" no formato 2pe15.3 =    -12.35E-01
"a" no formato en15.3  =    -1.235E+00
"a" no formato es15.3  =    -1.235E+00

a =    -1.235E+00  b =    2.222E+00
                   c =    3.333E+00
                   d =    4.321E+00
```

- 10) Até entender, compare o conteúdo de cada linha do arquivo "saida4c.txt" com os formatos de edição empregados em cada comando WRITE no programa04c.f90.
- 11) Note que o último algarismo é arredondado.
- 12) Note que um número real simples, sem formato de edição (primeiro resultado do programa), é representado com sete algarismos.
- 13) Altere os formatos usados no programa04c.f90 e verifique o que ocorre.

## 4.7 FECHANDO UM PROJETO E O FORTRAN

- 1) Dentro da “janela principal” do Fortran, para fechar um projeto que está aberto, basta executar o seguinte: File, Close Workspace
- 2) Para fechar o programa Fortran, basta executar: File, Exit

#### **4.8 FECHANDO O WINDOWS**

- 1) Execute: Start, Shut Down
- 2) Na janela “Shut Down Windows”, escolha a opção “Close all programs and log on as a different user?”. Depois clique no botão “Yes”.
- 3) Depois que o computador sair do Windows, por favor, apenas desligue o monitor do computador.

## **LISTA DE EXERCÍCIOS DA 4ª AULA**

**(Para ser entregue na próxima aula)**

### **Exercício 4.1**

- 1) Implemente (escreva, edite) um programa em Fortran 90 para calcular a soma dos termos de uma progressão aritmética (P.A.) com o seguinte algoritmo:
  - a) Ler os dados do arquivo “dados4.txt”; os dados são seu nome completo (caracter), o valor (real) do primeiro termo da P.A., a diferença (real) entre dois elementos subseqüentes e o número (inteiro) de elementos da P.A.
  - b) Calcular os resultados: o valor (real) do último termo da P.A. e a soma (real) de todos os elementos
  - c) Escrever no arquivo “saida4.txt” os dados lidos do arquivo “dados4.txt” e os dois resultados, incluindo comentários para informar ao que corresponde cada valor
- 2) Compile o programa
- 3) Gere o executável do programa
- 4) Execute o programa
- 5) Entregue na próxima aula uma impressão do seu programa Fortran 90, do arquivo “dados4.txt” e do arquivo “saida4.txt”

### **Exercício 4.2**

- 1) Acesse o site da disciplina no endereço:  
[ftp://ftp.demec.ufpr.br/Disciplinas/Tm102/Marchi/semestre\\_2003\\_2/](ftp://ftp.demec.ufpr.br/Disciplinas/Tm102/Marchi/semestre_2003_2/)
- 2) Estude o capítulo 4 da “apostila\_logica\_programacao.pdf”
- 3) Faça os exercícios do capítulo 4
- 4) Entregue na próxima aula a solução dos exercícios do capítulo 4