



UNIVERSIDADE FEDERAL DO PARANÁ

CURSO DE ENGENHARIA MECÂNICA

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

2ª aula

INSTRUÇÕES

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

2.1 OBJETIVOS DA AULA

1. Utilização das operações matemáticas básicas

2.2 CRIANDO UM NOVO PROJETO DO TIPO CONSOLE APPLICATION

Execute o seguinte:

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. Há pelo menos duas opções. Primeira, se você estiver usando a conta Fortran, já deve estar selecionado o diretório “C:\MSDev\Projects”; se não estiver, selecione este diretório. Segunda, se você já tem conta própria na rede do DEMEC, selecione o seu diretório no “drive” Server1.
 - c) Name: indique o nome do projeto, por exemplo, “projeto2”. 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

2.3 INSERINDO UM ARQUIVO DENTRO DO PROJETO

Execute o seguinte:

1. Na “janela principal” do Fortran: Insert, Files into Project
2. Na janela “Insert Files into Project”:
 - 1) File Name: entre com o nome do arquivo a ser criado, por exemplo, “programa02.f90”. Sempre use ‘f90’ como extensão 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”.

- 2) Clique em “OK”
- 3) Na janela “Microsoft Developer Studio”, clique em YES
- 4) 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.
- 5) Clique duas vezes sobre o nome do arquivo que foi inserido
- 6) Na janela “Microsoft Developer Studio”, clique em YES
- 7) O arquivo inserido está pronto para ser editado no lado direito da “janela principal”

2.4 VERSÃO 1 DO PROGRAMA02.F90

1. Dentro do programa fonte, digite exatamente o seguinte texto:

```
!
! programa02.f90, versão 1
!
! objetivo do programa: operações matemáticas básicas
!
! entrada dos dados diretamente dentro do programa
!
a = 2
b = 3
!
! execução das operações matemáticas básicas
!
c = a + b ! adição (adiciona "a" a "b")
d = a - b ! subtração (subtrai "b" de "a")
e = a * b ! multiplicação (multiplica "a" por "b")
f = a / b ! divisão (divide "a" por "b")
g = a ** b ! potência (eleva "a" à potência "b")
!
! saída dos resultados na janela do DOS
!
write(*,*) 'a + b = ', c
write(*,*) 'a - b = ', d
write(*,*) 'a * b = ', e
write(*,*) 'a / b = ', f
write(*,*) 'a ** b = ', g
!
end
```

2. Depois de digitar este texto, grave-o da seguinte forma: File, Save
3. Comentários sobre este programa:
 - a) Os espaços em branco que estão nas operações matemáticas e nas atribuições não influenciam em nada e não são necessários. Eles têm apenas função estética, facilitando a visualização das operações.
 - b) Note que os valores das variáveis “a” e “b” são atribuídos (fornecidos) a elas diretamente dentro do programa.
 - c) O programa executará as operações matemáticas básicas, atribuindo seus resultados às variáveis “c”, “d”, “e”, “f” e “g”.
 - d) Finalmente, o programa irá “escrever” os resultados na janela do DOS, usando o comando “write”
4. Compile o programa executando: Build, Compile
5. Gere o executável do programa com Build, Build

6. Execute o programa com Build, Execute. O resultado deverá ser:

```
a + b =          5.000000
a - b =         -1.000000
a * b =          6.000000
a / b =         6.666667E-01
a ** b =         8.000000
Press any key to continue
```

7. Confira estes resultados. Você concorda com eles?

2.5 VERSÃO 2 DO PROGRAMA02.f90

1. Partindo da versão 1 do programa02.f90, da seção 2.4, digite exatamente o seguinte texto:

```
!
! programa02.f90, versão 2
!
! objetivo do programa: operações matemáticas básicas
!
! entrada dos dados via teclado
!
! write(*,*) 'Entre com o valor da variavel "a"'
! read(*,*) a
!
! write(*,*) 'Entre com o valor da variavel "b"'
! read(*,*) b
!
! execução das operações matemáticas básicas
!
! c = a + b    ! adição (adiciona "a" a "b")
! d = a - b    ! subtração (subtrai "b" de "a")
! e = a * b    ! multiplicação (multiplica "a" por "b")
! f = a / b    ! divisão (divide "a" por "b")
! g = a ** b   ! potenciação (eleva "a" à potência "b")
!
! saída dos resultados na janela do DOS
!
! write(*,*) 'a + b = ', c
! write(*,*) 'a - b = ', d
! write(*,*) 'a * b = ', e
! write(*,*) 'a / b = ', f
! write(*,*) 'a ** b = ', g
!
end
```

2. Depois de digitar este texto, grave-o da seguinte forma: File, Save
3. Comentários sobre este programa:
- Note que agora não são atribuídos (fornecidos) valores às variáveis “a” e “b” dentro do programa.
 - A palavra “read” é outro comando do Fortran, assim como “write” e “end”. O comando “read” é usado para atribuir (fornecer) dados ao programa. Neste caso, os dados serão fornecidos através do teclado, na janela do DOS, durante a execução do programa. Após cada comando “read”, é necessário clicar na tecla “Enter”. Portanto, após entrar com o

primeiro valor (a), você deverá clicar na tecla “Enter”, e após entrar com o segundo valor (b), você deverá novamente clicar na tecla “Enter”.

- c) O programa executará as operações matemáticas básicas, atribuindo seus resultados às variáveis “c”, “d”, “e”, “f” e “g”.
- d) Finalmente, o programa irá “escrever” os resultados na janela do DOS, usando o comando “write”
4. Compile o programa executando: Build, Compile
 5. Gere o executável do programa com Build, Build
 6. Execute o programa com Build, Execute
 7. Entre com o primeiro valor (a); use o valor 2; depois clique na tecla “Enter”
 8. Entre com o segundo valor (b); use o valor 3; depois clique na tecla “Enter”
 9. O resultado deverá ser:

```
Entre com o valor da variavel "a"
2
Entre com o valor da variavel "b"
3
a + b =          5.000000
a - b =         -1.000000
a * b =          6.000000
a / b =         6.666667E-01
a ** b =         8.000000
Press any key to continue
```

10. Confira estes resultados. Você concorda com eles?

11. Execute novamente o programa utilizando $a = 2$ e $b = 0.5$. Números reais devem usar ponto para separar a parte inteira da parte decimal. Se você usar vírgula no lugar do ponto, o programa entenderá que são dois números diferentes, um antes e outro depois da vírgula. O novo resultado deverá ser:

```
Entre com o valor da variavel "a"
2
Entre com o valor da variavel "b"
0.5
a + b =          2.500000
a - b =          1.500000
a * b =          1.000000
a / b =          4.000000
a ** b =         1.414214
Press any key to continue
```

12. Confira estes resultados. Você concorda com eles?

13. Execute novamente o programa utilizando os valores que você quiser para “a” e “b”

14. Execute novamente o programa utilizando $a = 2$ e $b = 0$. O novo resultado deverá ser:

```
Entre com o valor da variavel "a"
2
Entre com o valor da variavel "b"
0

run-time error M6103: MATH
- floating-point error: divide by zero
Press any key to continue
```

Neste caso note que nenhum resultado foi apresentado porque o programa não consegue dividir 2 por 0, gerando um erro e interrompendo a execução do programa. Você sempre

deve se prevenir evitando esse tipo de erro. Por exemplo, se o usuário entrar com um valor nulo, informe que não é possível e peça para entrar com um outro valor diferente de zero.

2.6 VERSÃO 3 DO PROGRAMA02.f90

1. Partindo da versão 2 do programa02.f90, da seção 2.5, digite exatamente o seguinte texto:

```
!  
! programa02.f90, versão 3  
!  
! objetivo do programa: operações matemáticas básicas  
!  
! entrada dos dados via teclado  
!  
! write(*,*) 'Entre com o valor da variavel "a"  
! read(*,*) a  
!  
! write(*,*) 'Entre com o valor da variavel "b"  
! read(*,*) b  
!  
! execução das operações matemáticas básicas  
! utilizando variáveis do tipo "inteira"  
!  
! i = a + b  
! j = a - b  
! k = a * b  
! l = a / b  
! m = a ** b  
! n = a  
!  
! saída dos resultados na janela do DOS  
!  
! write(*,*) 'a + b = ', i  
! write(*,*) 'a - b = ', j  
! write(*,*) 'a * b = ', k  
! write(*,*) 'a / b = ', l  
! write(*,*) 'a ** b = ', m  
! write(*,*) 'a = ', n  
!  
! end
```

2. Depois de digitar este texto, grave-o da seguinte forma: File, Save
3. Comentários sobre este programa:
 - a) Na versão 2 do programa02.f90, da seção 2.5, os resultados dos cálculos foram atribuídos às variáveis “c”, “d”, “e”, “f” e “g”. Nesta versão 3 do programa02.f90, os mesmos cálculos são atribuídos às variáveis “i”, “j”, “k”, “l” e “m”.
 - b) Por default, isto é, se nada for especificado explicitamente em contrário, variáveis que iniciam pelas letras “i”, “j”, “k”, “l”, “m” e “n” são do tipo INTEIRA. Variáveis do tipo inteira só conseguem armazenar na memória do computador valores INTEIROS, ou seja, o conjunto dos números inteiros.
 - c) Por default, isto é, se nada for especificado explicitamente em contrário, variáveis que NÃO iniciam pelas letras “i”, “j”, “k”, “l”, “m” e “n” são do tipo REAL. Variáveis do tipo real só conseguem armazenar na memória do computador valores REAIS, ou seja, o conjunto dos números reais.

- d) Note que o programa executará as operações matemáticas básicas com as variáveis “a” e “b”, que são do tipo real, mas os resultados serão atribuídos às variáveis “i”, “j”, “k”, “l”, “m” e “n”, que são do tipo inteira.
- e) Finalmente, o programa irá “escrever” os resultados na janela do DOS, usando o comando “write”
4. Compile o programa executando: Build, Compile
 5. Gere o executável do programa com Build, Build
 6. Execute o programa com Build, Execute
 7. Entre com o primeiro valor (a); use o valor 2; depois clique na tecla “Enter”
 8. Entre com o segundo valor (b); use o valor 3; depois clique na tecla “Enter”
 9. O resultado deverá ser:
Entre com o valor da variavel "a"
2
Entre com o valor da variavel "b"
3
a + b = 5
a - b = -1
a * b = 6
a / b = 0
a ** b = 8
a = 2
Press any key to continue
 10. Utilizando a sua calculadora, refaça estes cálculos e os compare com os resultados fornecidos pelo programa02, no item anterior, principalmente de “a” dividido por “b”. Você concorda com eles?
 11. Note que todos os resultados são números inteiros, ao contrário das versões 1 e 2 do programa02.f90 que apresentavam números reais, com casas decimais.
 12. Execute novamente o programa utilizando a = 1.5 e b = 0.4. O novo resultado deverá ser:
Entre com o valor da variavel "a"
1.5
Entre com o valor da variavel "b"
0.4
a + b = 1
a - b = 1
a * b = 0
a / b = 3
a ** b = 1
a = 1
Press any key to continue
 13. Utilizando a sua calculadora, refaça estes cálculos e os compare com os resultados fornecidos pelo programa02, no item anterior. Você concorda com eles?
 14. Execute novamente o programa utilizando os valores que você quiser para “a” e “b”

2.7 VERSÃO 4 DO PROGRAMA02.f90

1. Partindo da versão 3 do programa02.f90, da seção 2.6, digite exatamente o seguinte texto:
!
! programa02.f90, versão 4
!
! objetivo do programa: operações matemáticas básicas

```
!  
! entrada dos dados via teclado  
!  
write(*,*) 'Entre com o valor da variavel "a"  
read(*,*) a  
!  
write(*,*) 'Entre com o valor da variavel "b"  
read(*,*) b  
!  
write(*,*) 'Entre com o valor da variavel "a2"  
read(*,*) a2  
!  
write(*,*) 'Entre com o valor da variavel "b2"  
read(*,*) b2  
!  
! execução de operações matemáticas básicas  
!  
h1 = a + b - a2  
h2 = a + b / a2  
h3 = (a + b) / a2  
!  
write(*,*) 'h1 =',h1  
write(*,*) 'h2 =',h2  
write(*,*) 'h3 =',h3  
!  
h4 = a + b / a2 * b2  
h5 = a + b / (a2 * b2)  
h6 = a + b * a2 ** b2  
h7 = a + (b * (a2 ** b2))  
!  
write(*,*) 'h4 =',h4  
write(*,*) 'h5 =',h5  
write(*,*) 'h6 =',h6  
write(*,*) 'h7 =',h7  
!  
h8 = a / b  
i = a / b  
!  
write(*,*) 'h8 =',h8  
write(*,*) 'i =',i  
!  
end
```

2. Depois de digitar este texto, 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 o primeiro valor (a); use o valor 1; depois clique na tecla "Enter"
7. Entre com o segundo valor (b); use o valor 2; depois clique na tecla "Enter"
8. Entre com o primeiro valor (a2); use o valor 1.5; depois clique na tecla "Enter"
9. Entre com o segundo valor (b2); use o valor 2.5; depois clique na tecla "Enter"
10. O resultado deverá ser:
Entre com o valor da variavel "a"
1
Entre com o valor da variavel "b"

```
2
Entre com o valor da variavel "a2"
1.5
Entre com o valor da variavel "b2"
2.5
h1 =      1.500000
h2 =      2.333333
h3 =      2.000000
h4 =      4.333333
h5 =      1.533333
h6 =      6.511352
h7 =      6.511352
h8 =     5.000000E-01
i =       0
Press any key to continue
```

11. O objetivo da versão 4 do programa02.f90 é testar a ESCALA DE PRIORIDADES que é adotada no Fortran para executar operações matemáticas com diversos termos numa mesma expressão. As prioridades são:
- 1º Potenciação (**)
 - 2º Multiplicação (*) ou divisão (/)
 - 3º Adição (+) ou subtração (-)
- Outras normas adotadas:
- a) No caso de operações que tenham a mesma prioridade, como a adição e a subtração, primeiro é executado o cálculo mais à esquerda.
 - b) Para que os cálculos sejam executados na ordem que se deseja, deve-se usar parênteses. Neste caso, a ordem de execução é do parênteses mais interno para o mais externo.
 - c) Havendo duas potenciações consecutivas, o cálculo é feito da direita para a esquerda.
 - d) Uma operação envolvendo dois números inteiros resulta num número inteiro.
 - e) Uma operação envolvendo um número inteiro e um número real resulta num número real.
12. Utilizando a sua calculadora e as normas descritas no item acima, obtenha os valores das variáveis “h1” a “h8” e de “i” e os compare com os resultados fornecidos pelo programa02. Você concorda com eles?
13. Execute novamente o programa utilizando os valores que você quiser para “a”, “b”, “a2” e “b2” e verifique os resultados do programa com a sua calculadora.

2.8 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

2.9 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 2ª AULA

(Para ser entregue na próxima aula)

Exercício 2.1

Em que ordem são calculadas as seguintes expressões aritméticas?

1. $(R+U/T)**(T+V/G)$
2. $A/B/(A/B/C)$
3. $X/Y*Z+P*Q/R*S/T$
4. $A+(B+C)/D-E*F+G**H$
5. $B/((A-B)*C)+A**2$

Exercício 2.2

Determine o resultado das expressões aritméticas abaixo considerando $A=2.0$, $B=12.0$, $I=6$ e $J=3$.

1. B/A
2. $A + B + I$
3. J / I
4. $B/(I-J)$
5. $1.0/J$
6. $1/J$

Exercício 2.3

Escreva as seguintes expressões algébricas na forma de expressões aritméticas do FORTRAN.

1. A^{I+J}
2. $\frac{1}{\frac{1}{A} + \frac{1}{B}}$
3. $\frac{1}{2}\sqrt{A^2 + B^2}$
4. $\frac{A(A-1)}{2}$

Exercício 2.4

1. Acesse o site da disciplina no endereço:

ftp://ftp.demec.ufpr.br/Disciplinas/Tm102/Marchi/semestre_2003_2/

2. Estude o capítulo 2 da “apostila_logica_programacao.pdf”
3. Faça os exercícios do capítulo 2
4. Entregue na próxima aula a solução dos exercícios do capítulo 2