

# ***Programação de Computadores III***

## **Aula 7**

**Professor Leandro Augusto Frata Fernandes**  
laffernandes@ic.uff.br

Material disponível em  
<http://www.ic.uff.br/~laffernandes/teaching/2011.1/tcc-03.063>

---

### ***Roteiro da Aula de Hoje***

- Introdução ao FORTRAN (Parte I)
  - Organização de programas
  - Tipos de dados
  - Variáveis

## ***Introdução ao FORTRAN***

- Cinco aspectos precisam ser considerados antes de escrever o primeiro programa em FORTRAN
  - Como um programa é **organizado**
  - Quais são os **tipos de dados** disponíveis
  - Como declarar **variáveis**
  - Como fazer a **entrada e saída** de dados
  - Como **atribuir valores** às variáveis

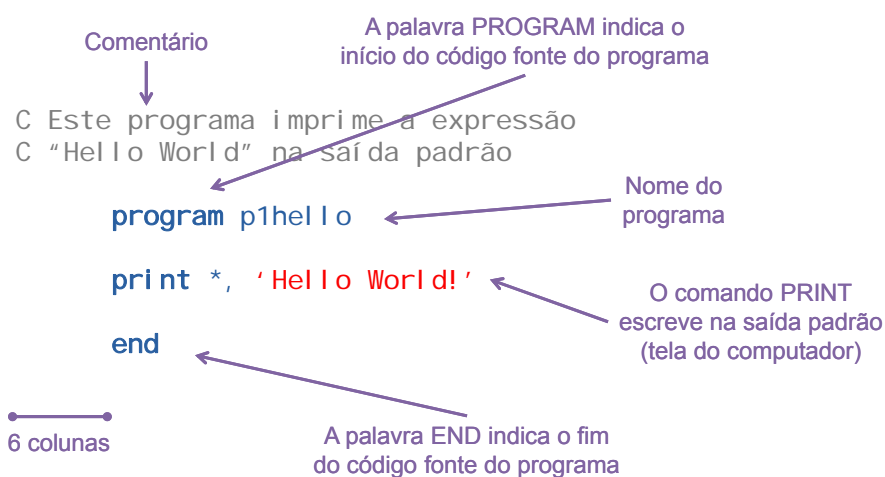
## ***Organização de Programas***

- O programa deve ser escrito utilizando um editor de código, como o **Force** (<http://force.lepsch.com>)
- O FORTRAN faz exigências quanto o uso das colunas no código fonte
  - As **colunas 1 a 6** são **reservadas** para controle
  - As **colunas 7 a 72** são utilizadas **para o programa** em si
  - As **colunas 73 em diante** são ignoradas
- Você **DEVE** **documentar o código fonte**

## Documentação do Código Fonte

- Sintaxe
  - Iniciar a coluna 1 da linha de comentário com C ou \*, com isso o restante da linha será ignorada
- Boas práticas
  - Utilize linguagem natural (português)
  - Descreva a finalidade do programa nas primeiras linhas do código fonte
  - Descreva o que cada “pedaço” de código faz

## Exemplo Simples de Programa FORTRAN



## ***Tipos de Dados***

- Toda variável declarada **deve ter um tipo de dado**
- Lembre-se
  - O tipo define quais são os **valores possíveis**
  - Diz ao compilador **quanto de memória** deve ser reservado para a variável
  - Os tipos dividem-se em
    - Numéricos: inteiro, real e real com dupla precisão
    - Não numéricos: caractere e lógico

## ***Tipos de Dados no FORTRAN (32bits)***

- Tipo inteiro: **integer**
  - Valores **numéricos inteiros** de  $-2^{31}$  a  $2^{31}$
  - Exemplos: 6, 123, -45
- Tipo real com precisão simples: **real**
  - Valores **numéricos reais** com precisão de até **sete casas decimais**
  - Magnitude de  $10^{-38}$  a  $10^{38}$
  - O ponto (".") separa a parte inteira da parte fracionária
  - Exemplo:  $0.6023E24 = 0.6023 \times 10^{24}$

## ***Tipos de Dados no FORTRAN (32bits)***

- Tipo real com precisão dupla: **double precision**
  - Valores **numéricos reais** com precisão de até **quinze casas decimais**
  - Magnitude de  $10^{-308}$  a  $10^{308}$
  - Exemplos:  $0.54336D-94 = 0.54336 \times 10^{-94}$
- Tipo lógico: **logical**
  - Permite representar **valores lógicos**, ou seja, verdadeiro (.true.) ou falso (.false.)
  - Note que é preciso utilizar um ponto antes e outro depois dos valores

## ***Tipos de Dados no FORTRAN***

- Tipo caractere: **character**
  - Variáveis deste tipo permitem a **representação de texto** demarcado por aspas simples (')
  - Exemplo: 'Isso é um texto'
  - Para utilizar o caractere ' no meio do texto, coloque duas aspas simples consecutivas
  - Exemplo: 'Quero usar "aspas" aqui'

## Declaração de Variáveis

- FORTRAN aceita declaração **implícita e explícita** de variáveis

### Declaração Implícita

```
program p2i m p l i c  
  
print *, 'Informe A: '  
read *, a  
  
x = 2 + a  
print *, 'Total: ', x  
  
end
```

### Declaração Explícita

```
program p3expl i c  
  
real a, x  
  
print *, 'Informe A: '  
read *, a  
  
x = 2 + a  
print *, 'Total: ', x  
  
end
```

## Declaração Implícita

- **Declaração implícita** se baseia na primeira letra do identificador (nome) da variável
  - Variáveis que começam com a-h ou o-z são reais
  - Variáveis que começam com i-n são inteiras
- Contudo, isso pode **levar a erros!**
- É recomendado utilizar declaração explícita

## ***Declaração Explícita***

- A **declaração explícita** define qual é o tipo de cada variável que será utilizada no programa
- A declaração **deve vir antes** de todos os demais comandos
- Regras para identificadores (nomes) de variáveis
  - Ter somente letras de A a Z, números de 0 a 9, e \_
  - Começar com uma letra
  - Maiúsculas e minúsculas são equivalentes
- Use o comando especial **implicit none** no início do programa para forçar o uso de declaração explícita

## ***Exemplo de uso do “implicit none”***

Veja o arquivo p4forced.f

## ***Declaração Explícita***

- Mais de uma variável pode ser declarada numa mesma lista

**integer** contador, n

**real** X, Y, Z

**character**\*20 nome                      nome é uma variável caractere de tamanho 20

**character** nome\*20, endereco\*30 nome tem tamanho 20 e endereco tamanho 30

**logical** teste, aberto

**double precision** raio, diametro

## ***Entrada e Saída de Dados***

- Entrada
  - Leitura de valores informados pelo usuário ou vindos de arquivo
  - Comando **read**
- Saída
  - Impressão na tela ou em arquivo
  - Comandos **write** e **print**
- Por simplicidade, utilizaremos apenas read e print



## ***Sintaxe dos Comandos “read” e “print”***

- Ambos os comandos esperam o formato do dado e a lista de parâmetros
  - `read (unidade, formato) lista_de_parâmetros`
  - `print (unidade, formato) lista_de_parâmetros`
- *Por enquanto utilizaremos formato livre*
  - `read *, altura, base`
  - `print *, 'A altura é ', altura, ' e a base é ', base`

O asterisco seguido de vírgula indica formato livre

## ***Exemplos de Declaração, Entrada e Saída***

Veja o arquivo p5misc.f

## ***Atribuição de Valores***

- No FORTRAN, o símbolo de igual (=) representa a atribuição de valores a variáveis
  - Exemplo:  $x = 5 + b$  (a variável  $x$  recebe o valor de  $5 + b$ )
- Na escrita de algoritmos, o símbolo  $\leftarrow$  representa atribuição e o símbolo  $=$  é o operador lógico de comparação de igualdade
  - Não confunda os símbolos  $=$  da escrita de um algoritmo com o  $=$  em FORTRAN!

## ***Exemplo de Atribuição de Valores***

Veja o arquivo p6atribu.f