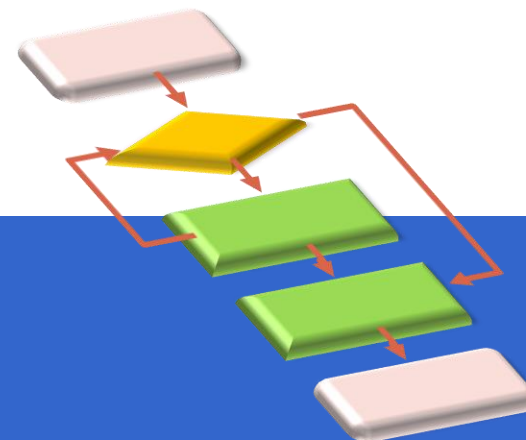




INSTITUTO DE COMPUTAÇÃO



Algoritmos II

MODULARIZAÇÃO Parametrização.

Prof.^a Vanessa de Oliveira Campos

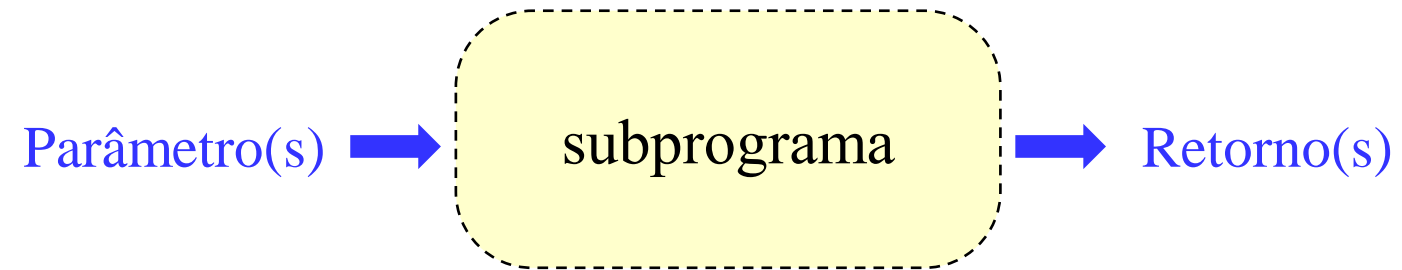
Parâmetros

- Parâmetros formais e reais;
- Parâmetros por valor ou por referência.



Parâmetros

- Parâmetros são espaços de armazenamento que permitem a comunicação entre subprogramas e o “mundo externo”.



Parâmetros

Exemplo:

```
funcao soma (x,y: inteiro): inteiro  
inicio  
    retorne x + y  
fimfuncao
```



Parâmetros

Exemplo:

```
Funcao substring (str: literal; p: inteiro): literal
Var
    i: inteiro
    resultado: literal
Inicio
    resultado <- ""
    para i de 1 ate p faca
        resultado <- resultado + copia(str,i,1)
    fimpara
    retorne resultado
Fimfuncao
```



Parâmetros

- Todos os elementos utilizados em um módulo devem ser declarados.
- Declaração de parâmetros, além de dar nomes, definem seu tipo.
- Na declaração se define a forma de uso dos parâmetros, bem como a ordem em que devem ser passados.
- Durante a chamada, seus valores são substituídos por variáveis ou valores do mesmo tipo.



Algoritmo "MetodoComParametros"

Var

 a: inteiro

Inicio

 leia(a)

 metodoComParametro(a)

 metodoComParametro(3)

Fimalgoritmo

Procedimento metodoComParametro(param: inteiro)

Var

 i: inteiro

Inicio

 para i de 1 ate param faca

 escreva(i)

 fimpara

Fimprocedimento

Algoritmo "DiferancasDeParametros"

Var

 k: inteiro

Inicio

 leia(k)

 doisParametros (k, 7)

Fimalgoritmo

procedimento doisParametros (i: inteiro; j: real)

Inicio

 se i < 10 entao

 j <- j + 2

 senao

 j <- 0

 fimse

 escreva(j)

Fimprocedimento

Parâmetros Formais e Reais

9

Algoritmo "DiferancasDeParametros"

Var

k: inteiro

Inicio

leia(k)

doisParametros (k, 7)

Fimalgoritmo

parâmetros reais

procedimento doisParametros (i: inteiro; j: real)

Inicio

se i < 10 entao

j <- j + 2

senao

j <- 0

fimse

escreva(j)

Fimprocedimento

parâmetros formais

Parâmetros Formais e Reais

- Os parâmetros reais podem ser:
 - Constantes
 - Variáveis
 - Expressões
 - Chamada de funções



Parâmetros Formais e Reais

- Os parâmetros reais podem ser:

- **Constantes**

- Variáveis

- Expressões

- Chamada de funções

```
Algoritmo "ExemploDeParametros"
```

```
Inicio
```

```
    metodo (5)
```

```
Fimalgoritmo
```



Parâmetros Formais e Reais

- Os parâmetros reais podem ser:
 - Constantes
 - **Variáveis**
 - Expressões
 - Chamada de funções

```
Algoritmo "ExemploDeParametros"  
Var  
    a: inteiro  
Inicio  
    leia(a)  
    metodo(a)  
Fimalgoritmo
```



Parâmetros Formais e Reais

- Os parâmetros reais podem ser:
 - Constantes
 - Variáveis
 - **Expressões**
 - Chamada de funções

```
Algoritmo "ExemploDeParametros"  
Var  
    a, b: inteiro  
Inicio  
    leia(a,b)  
    metodo((a + b) * 4)  
Fimalgoritmo
```



Parâmetros Formais e Reais

- Os parâmetros reais podem ser:
 - Constantes
 - Variáveis
 - Expressões
 - **Chamada de funções**

```
Algoritmo "ExemploDeParametros"  
Var  
    a: inteiro  
Inicio  
    leia(a)  
    metodo( metodo(a) )  
Fimalgoritmo
```



Algoritmo "MetodoComRetorno"

Var

 a, b: inteiro

Inicio

 leia(a)

 b ← RotinaComRetorno(RotinaComRetorno(a))

 escreva(b)

Fimalgoritmo

Funcao RotinaComRetorno(aux: inteiro): inteiro

Var

 resposta: inteiro

Inicio

 resposta ← aux * 5

 retorne resposta

Fimfuncao

Passagem de Parâmetros

▪ Por valor

- Também chamado de “por cópia”;
- As modificações do parâmetro formal não afeta o parâmetro real.

▪ Por referência

- As modificações efetuadas nos parâmetros formais também afetarão os parâmetros reais.



Passagem de Parâmetros

17

```
Algoritmo "AlgoritmoValor"
Var
  a: inteiro

  Procedimento metodo( b: inteiro )
    Inicio
      b <- b + 1;
      escreval( "Durante: ", b)
    Fimprocedimento

  Inicio
    a <- 1
    escreval( "Antes: ", a )
    metodo( a )
    escreval( "Depois: ", a )
  Fimalgoritmo
```

Qual a saída?

Antes: 1

Durante: 2

Depois: 1

Algoritmo "AlgoritmoReferencia"

Var

 a: inteiro

Procedimento metodo(**var** b: inteiro)

Inicio

 b <- b + 1;

 escreval("Durante: ", b)

Fimprocedimento

Inicio

 a <- 1

 escreval("Antes: ", a)

 metodo(a)

 escreval("Depois: ", a)

Fimalgoritmo

Qual a saída?

Antes: 1

Durante: 2

Depois: 2

Passagem de Parâmetros Por Referência

- Na passagem por referência, o **endereço** de uma variável deve ser passado na chamada de subprogramas;
- Dessa forma, o subprograma pode **modificar a variável diretamente**, o que em geral não é recomendável, mas há situações onde esse recurso é necessário;
 - Por exemplo: subprogramas que devolvem mais de um valor.



Recomendações

- **Identificação de parâmetros.**
- **Não usar variáveis globais.**
- **Testar passagem de parâmetro incorreta.**
- **Verificar a correção dos parâmetros.**
- **Um só ponto de saída do subprograma.**



