

Algoritmos e Programação I

Estruturas de Controle e Repetição

Estruturas de Controle

- ❖ Uma vez que a expressão condicional foi avaliada, isto é, reduzida para um valor V ou F, uma estrutura de controle é necessária para governar as ações que se sucederão. Daqui em diante, veremos dois tipos de estruturas de controle: **estrutura condicional** e **estrutura de repetição**.

Estruturas de Controle

- ❖ A estrutura condicional mais simples é a se - então - fimse.
- ❖ Forma Geral:
se <expressão condicional> então
comando₁
comando₂
...
comando_n
fimse

Estruturas de Controle

- ❖ Como exemplo, considere um algoritmo para ler dois números, a e b , e escrevê-los em ordem não decrescente.

Estruturas de Controle

// algoritmo para escrever dois números em ordem não decrescente
algoritmo "ordena dois números"

var

// declaração de variáveis
a, b, temp : **inteiro**

inicio

// lê os números
leia a, b

// ordena os números

se a > b **então**

temp ← a

a ← b

b ← temp

fimse

// escreve os números ordenados

escreva a, b

finalgoritmo

Estruturas de Controle

- ❖ A operação de troca:
 - ❖ Esta operação é realizada com a ajuda de uma outra variável, denominada *temp*, e através de três atribuições.
 - ❖ Para atribuir o valor de *a* a *b* e o valor de *b* a *a*, temos de utilizar uma outra variável, pois ao executarmos $a \leftarrow b$, atribuímos o valor de *b* a *a* e, conseqüentemente, perdemos o valor que, anteriormente, estava retido em *a*.

Estruturas de Controle

❖ Uma variação da estrutura de controle se - então - fimse é a estrutura se - então - senão - fimse.

❖ A forma geral deste comando é dada a seguir.

se <expressão condicional> então

comando₁

comando₂

...

comando_n

senão

Comando₁

Comando₂

...

comando_n

fimse

Estruturas de Controle

- ❖ Como exemplo considere um algoritmo para escrever uma mensagem com o resultado de um exame, dada a nota obtida pelo candidato no exame.

Estruturas de Controle

```

// algoritmo para escrever resultado de um exame
// baseado na nota obtida pelo candidato
algoritmo "resultado exame"
    // declaração variáveis
    var
    NOTAMINIMA, nota : real

    inicio
        //define constante
        NOTAMINIMA ← 6.0

        // lê a nota do candidato
        leia nota

        // compara a nota lida e escreve resultado
        se nota ≥ NOTAMINIMA então
            escreva "candidato aprovado"
        senão
            escreva "candidato reprovado"
        fimse
    fimalgoritmo

```

Estruturas de Controle

- ❖ As estruturas condicionais podem estar “**aninhadas**”, isto é, elas podem ocorrer umas dentro de outras.
- ❖ Como exemplo disto, considere o seguinte algoritmo que lê três números e escreve-os em ordem não decrescente:

Estruturas de Controle

```
// algoritmo para ordenar três números
algoritmo "ordena três números"
  // declaração de variáveis
  var
    a, b, c, temp : inteiro

  inicio
    // lê os três números
    leia a, b, c

    // encontra o menor dos três números e guarda em a
    se (a > b) OU (a > c) então
      se (b ≤ c) então
        temp ← a
        a ← b
        b ← temp
      senão
        temp ← a
        a ← c
        c ← temp
      fimse
    fimse
  fimse
```

Estruturas de Controle

```
// encontra o valor intermediário e guarda em b  
se (b > c) então  
    temp ← b  
    b ← c  
    c ← temp
```

fimse

```
// escreve os números em ordem não decrescente  
escreva a, b, c
```

fimalgoritmo

Estruturas de Repetição

- ❖ A estrutura de repetição, ou simplesmente laço, permite que um grupo de comandos seja executado repetidamente um número determinado de vezes ou até que uma determinada condição se torne verdadeira ou falsa.

- ❖ Nesta Subseção, estudaremos três estruturas de repetição:
 - a estrutura para - faça
 - a estrutura enquanto - faça
 - a estrutura repita - até

Estruturas de Repetição

- ❖ A estrutura de repetição para - faça possui a seguinte forma geral:

para <variável> de <valor inicial> até <valor final> faça
comando₁
comando₂
...
comando_n
fimpara

Estruturas de Repetição

- ❖ Como exemplo, considere o seguinte algoritmo para calcular e exibir a soma de todos os números pares desde 100 até 200, inclusive.

Estruturas de Repetição

```

// algoritmo para somar os pares de 100 a 200
algoritmo "soma pares 100 a 200"
    // declaração de variáveis
    var
        soma, i : inteiro

    inicio
        // inicializa com 0 a variável que guardará a soma
        soma ← 0
        // calcula a soma
        para i de 100 até 200 faça
            se i MOD 2 = 0 então
                soma ← soma + i
            fimse
        fimpara

        // escreve o valor da soma
        escreva "A soma dos pares de 100 a 200 é: ", soma

fimalgoritmo
    
```

Estruturas de Repetição

❖ Uma variação da estrutura de repetição para - faça é aquela que nos possibilita controlar o valor do incremento da variável contadora do laço.

```
para <variável> de <valor inicial> até <valor final> passo <incremento> faça  
  comando1  
  comando2  
  ...  
  comandon  
fimpara
```

Estruturas de Repetição

- ❖ Esta variação da estrutura de repetição para - faça nos permite, por exemplo, resolver o problema de calcular a soma de todos os pares de 100 a 200 de uma forma mais elegante do que aquela vista anteriormente, como podemos constatar a seguir.

Estruturas de Repetição

```
// algoritmo para somar os pares de 100 a 200
algoritmo "soma pares 100 a 200"
    var
        // declaração de variáveis
        soma, i : inteiro

    inicio
        // inicializa com 0 a variável que guardará a soma
        soma ← 0
        // calcula a soma
        para i de 100 até 200 passo 2 faça
            soma ← soma + i
        fimpara

        // escreve o valor da soma
        escreva "A soma dos pares de 100 a 200 é: ", soma
fimalgoritmo
```

Estruturas de Repetição

- ❖ A estrutura de repetição para - faça deve ser utilizada apenas quando queremos repetir a execução de um ou mais comandos um número conhecido de vezes, como no exemplo anterior.
- ❖ Entretanto, há problemas em que não é possível determinar, previamente, o número de repetições. Neste caso, devemos utilizar a estrutura enquanto - faça ou repita - até.

Estruturas de Repetição

❖ A estrutura enquanto - faça possui a seguinte forma geral:

enquanto <expressão condicional> faça
comando₁
comando₂
...
comando_n
fimenquanto

Estruturas de Repetição

- ❖ Como exemplo, considere o seguinte algoritmo para ler um número inteiro n , que não contém dígito 0, e escrever um número inteiro m que corresponde a n invertido.

Estruturas de Repetição

```
// algoritmo para inverter um número inteiro sem dígito 0
algoritmo "inverte número"

    // declaração de variáveis
    var
        n, r, m : inteiro

    inicio
        // lê o valor de um inteiro
        leia n

        // inicializa a variável que conterà o inteiro invertido
        m ← 0

        // encontra o número invertido
        enquanto n > 0 faça
            r ← n MOD 10
            m ← m * 10 + r
            n ← n DIV 10
        fimenquanto

        // exibe o número invertido
        escreva m
```

fimalgoritmo

Estruturas de Repetição

- ❖ A estrutura de repetição repita - até é semelhante à estrutura enquanto - faça, pois ambas são utilizadas quando não conhecemos, antecipadamente, o número de repetições.
- ❖ A diferença entre elas reside no fato que a sequência de instruções da estrutura repita - até é executada pelo menos uma vez, independentemente da expressão condicional ser ou não verdadeira.

Estruturas de Repetição

- ❖ A estrutura repita - até tem a seguinte forma geral:

repita

comando₁

comando₂

...

comando_n

até <expressão condicional>

Estruturas de Repetição

❖ Como exemplo, considere o seguinte trecho algorítmico:

...

repita

 escreva "entre com um número positivo: "

leia n

até $n > 0$

...

Referências

- ❖ **SIQUEIRA, Marcelo F. Algoritmos e Estrutura de Dados. Mato Grosso do Sul: CCET/CPCX - UFMS, 2007.**