

Lógica

Introdução a Lógica de Programação



Agenda

Estruturas Condicionais

- Laço de Repetição



O que é um Laço de Repetição?

Um laço de repetição permite executar um bloco de código várias vezes, sem precisar escrevê-lo repetidamente.

Isso torna o código mais eficiente e organizado.

Estrutura do Laço para



para <variável> de <valor_inicial> ate <valor_final> passo <incremento> faca
 <bloco de código>
fimpara



variável: Variável que controla o laço.

valor_inicial: Valor inicial da variável.

valor_final: Valor final da variável.

incremento: Valor que será somado à variável a cada iteração.

bloco de código: Código que será repetido.



Estruturas Condicionais

Tente praticar

```
algoritmo "contador1a10"  
var  
    C: inteiro  
inicio  
    Para C <- 1 ate 10 faca  
        Escreval (C)  
    FimPara  
fimalgoritmo
```

Tente praticar escolhendo a contagem regressiva e outro nº para passo

```
algoritmo "contador1a10"  
var  
    C: inteiro  
inicio  
    Para C <- 10 ate 1 passo 3 faca  
        Escreval (C)  
    FimPara  
fimalgoritmo
```



Estruturas Condicionais

Tente praticar

```
algoritmo "somador"
var
  N, C, S: inteiro
inicio
  S <- 0
  Para C <- 1 ate 5 faca
    Escreval ("Digite um valor: ")
    Leia (N)
    S <- S + N
  FimPara
  Escreval ("A sooma dos valores é ",S)
finalgoritmo
```



Estruturas Condicionais

Quando não podemos utilizar a estrutura Para

A estrutura de repetição Para não é utilizada quando não sabemos quantas vezes queremos repetir. Nesse caso devemos utilizar a estrutura Enquanto.

Mesmo que existam alguns métodos para se fazer esta ação não é aconselhado que se utilize a estrutura Para nestes casos.

Exemplo 1 - Contagem Simples

```
algoritmo "ContagemSimples"  
var  
    i: inteiro  
inicio  
    para i de 1 ate 10 faca  
        escreval(i)  
    fimpara  
fimalgoritmo
```



O laço irá executar 10 vezes.
A cada iteração, o valor de i será exibido na tela.




Exemplo 2 - Soma de Números

```
algoritmo "SomaNumeros"
var
    i, soma: inteiro
inicio
    soma <- 0
    para i de 1 ate 5 faca
        soma <- soma + i
    fimpara
    escreval("A soma dos números
    é: ", soma)
finalgoritmo
```

O laço irá executar 5 vezes.

A cada iteração, o valor de i será somado à variável soma.

No final, o valor total da soma será exibido na tela.



Exemplo 3 - Tabuada

```
algoritmo "Tabuada"
var
    i, numero: inteiro
inicio
    escreva("Digite um número: ")
    leia(numero)
    para i de 1 ate 10 faca
        escreval(numero, " x ", i, " = ", numero * i)
    fimpara
fimalgoritmo
```

O programa solicita ao usuário um número.
O laço executa 10 vezes, calculando e exibindo a tabuada do número informado.



Estruturas Condicionais

Tente fazer um algoritmo para mostrar quantos valores digitados estão entre 0 e 10, após 6 digitações.

Estruturas Condicionais

Tente fazer um algoritmo para mostrar quantos valores digitados estão entre 0 e 10, após 6 digitações.

```
algoritmo "quantosEntre0e10"
var
    C, V, Tot010: inteiro
inicio
    Para C <- 1 ate 6 faca
        Escreva ("Digite um valor: ")
        Leia (V)
        Se (V >= 0) e (V<=10) entao
            Tot010 <- Tot010 + 1
        FimSe
    FimPara
    Escreval ("Ao todo foram ",Tot010," valores entre 0 e 10")
finalgoritmo
```



Estruturas Condicionais

Tente fazer um algoritmo para mostrar quantos valores digitados estão entre 0 e 10, após 6 digitações. E também mostre a soma de todos ímpares entre 0 e 10 digitados.

Estruturas Condicionais

Tente fazer um algoritmo para mostrar quantos valores digitados estão entre 0 e 10, após 6 digitações. E também mostre a soma de todos ímpares entre 0 e 10 digitados.

```
algoritmo "quantosEntre0e10"
var
  C, V, Tot010, SImp: inteiro
inicio
  Para C <- 1 ate 6 faca
    Escreva ("Digite um valor: ")
    Leia (V)
    Se (V >= 0) e (V <= 10) entao
      Tot010 <- Tot010 + 1
      Se (V % 2 = 1) entao
        SImp <- SImp + V
      FimSe
    FimSe
  FimPara
  Escreval ("Ao todo foram ", Tot010, " valores entre 0 e 10")
  Escreval ("Nesse intervalo, a soma de ímpares foi ", SImp)
fimalgoritmo
```



Conclusão

Pontos importantes sobre o laço para:

- É uma estrutura poderosa para repetir tarefas.
- Torna o código mais eficiente e organizado.
- Permite automatizar tarefas repetitivas.