



# Programação de Computadores I

## Pseudocódigo

PROFESSORA CINTIA CAETANO

# Pseudocódigo

---

- ▶ **Pseudocódigo** é uma forma genérica de escrever um algoritmo, utilizando uma linguagem simples (nativa a quem o escreve, de forma a ser entendida por qualquer pessoa) sem necessidade de conhecer a sintaxe de nenhuma linguagem de programação.
- ▶ É, como o nome indica, um **pseudo-código** e, portanto, não pode ser executado num sistema real.
- ▶ Representado por PALAVRAS RESERVADAS, seguidas numa certa seqüência lógica.



# Palavras-chaves

---

- ▶ Uma palavra chave é uma palavra ou identificador que tem um significado especial para a linguagem de programação.
- ▶ O significado das palavras chave varia largamente de uma linguagem de programação para outra.



# Pseudocódigo

---

- ▶ **Início** – Indica o início de um bloco;
- ▶ **Fim** – Indica o final de um bloco;
- ▶ **Programa** – Define o nome do programa;
- ▶ **Var** – Lista de variáveis usadas;
- ▶ **Leia** – Lê um determinado dado do teclado;
- ▶ **Escreva** – Imprime um valor na tela.
- ▶ **//** - comentário



# Exemplo

---

Programa TESTE

Var

    a,b,c: inteiro;

Início

    Leia (a, b, c);

    Escreva (a,b,c);

Fim

---



# Exemplo

---

```
Programa Soma // nome programa Soma
Var a,b,c: inteiro //declara 3 variáveis inteiras
Inicio // Inicio do programa

    Escreva("Digite valor de A") //Imprime mensagem
    Leia(a) // Digitei 1
    Escreva("Digite valor de B") //Imprime mensagem
    Leia(b) // Digitei 2

    c <- a + b; // c recebe a + b

    Escreva("A soma é ", c) // A soma é 3
Fim // Fim do programa
```

---



# Indentação

---

- ▶ Indentação é um termo aplicado ao código fonte de um programa para indicar que os elementos hierarquicamente dispostos têm o mesmo avanço relativamente à posição  $(x,0)$ ;
- ▶ Na maioria das linguagens a indentação tem um papel meramente estético, tornando a leitura do código fonte mais fácil.



# Exercícios

---

- ▶ Faça um pseudocódigo de um programa que receba pelo teclado dois números inteiros, calcule e mostre o dobro do primeiro e o triplo do segundo.
- ▶ Faça um pseudocódigo que receba pelo teclado um número inteiro e calcule e mostre o quadrado deste número.
- ▶ Faça um pseudocódigo que receba pelo teclado as notas de duas provas, dois trabalhos e uma lista de exercícios. Calcule e mostre a média, sabendo que as provas têm peso 6, os trabalhos, peso 3 e as listas, peso 1.



# Exercícios

---

- ▶ Escreva um pseudocódigo para ler o valor de uma temperatura em Farenheit e convertê-la para graus Celsius e mostre o resultado no vídeo. A fórmula é:  
$$C = 5/9 (F - 32).$$
- ▶ Elabore um pseudo-código e um programa para ler dois valores inteiros, efetuar e mostrar o resultado das operações de adição, subtração, multiplicação e divisão de um número pelo outro. Não é necessário verificar se o divisor é zero.



# Exercícios

---

- ▶ Faça um pseudocódigo que receba o valor do raio de uma esfera o seu volume, basendo-se nas fórmulas dadas. Mostre o valor da área e o valor do volume da esfera. Utilize para  $\pi$  o valor de 3,14. (AREA =  $4 \pi R^2$ , VOLUME =  $\frac{4}{3} \pi R^3$ )
- ▶ Uma loja de animais precisa de um programa para calcular os custos da criação de coelhos. Deve-se ler o número de coelhos, calcular e apresentar o resultado. O custo é dado por:  $CUSTO = (N\_coelhos * 0.70) / 18 + 10$ .



# ESTRUTURA DE DECISÃO

---

- ▶ Com as instruções de **desvio** pode-se fazer com que o algoritmo proceda de uma ou outra maneira, de acordo com as decisões lógicas tomadas em função dos dados ou resultados anteriores.
- ▶ As principais estruturas de decisão são:
  - ▶ SE...ENTÃO
  - ▶ SE...ENTÃO...SENÃO
  - ▶ CASO... ENTÃO

# Se...Então...Senão

---

- ▶ Sintaxe

**Se** <condição> **então** <instruções 1>  
**senão** <instruções 2>

## **FimSe**

- ▶ Usado para decidir entre duas alternativas de ações. Um dos dois conjuntos de ações será executado e o outro não.
- ▶ Significado: Se a <condição> resultar em verdadeiro, então executar <instruções 1>. Senão, ignorar <instruções 1> e executar <instruções 2>.

# Se...Então...Senão

---

- ▶ Exemplo: “Mostrar a diferença entre 2 números quaisquer.”

Ler N1

Ler N2

**Se** (N1 > N2)      **então** DIF = N1 - N2

**senão** DIF = N2 - N1

## Se...Então...Senão (Encadeado)

---

- ▶ Se <condição> então <instruções >  
    senão se <condição> então <instruções 2>  
        senão <instruções 3>
- ▶ Exemplo

**Se** (valor<0) **então** res = 0  
**senão se** (valor>10) **então** res = 2  
**senão** res = 3

# Se...Então...Senão (Encadeado)

---

- ▶ **Exemplo: Verificar se um número é maior, menor ou igual a zero.**

```
Leia num;
```

```
Se (num = 0) então
```

```
    Imprime("igual a zero");
```

```
Senão Se (num > 0) então
```

```
        Imprime("maior que zero");
```

```
    Senão
```

```
        Imprime("menor que zero");
```

# Caso... Então...

---

- ▶ Sintaxe:

**Caso** <valor>:

<valor 1> **então** <ações 1>

<valor 2> **então** <ações 2>

<valor 3> **então** <ações 3>

<etc., quantos precisar>

## **FimCaso**

- ▶ Usada para escolher apenas um conjunto de ações dentre vários alternativos.
- ▶ Significado: Se <valor> for igual a <valor 1>, então executar <ações 1>. Se for igual a <valor 2>, ignorar <ações 1> e executar <ações 2>, e assim por diante.

# Caso... Então...

---

- ▶ Exemplo: “Escolher uma das 4 operações básicas para aplicar em dois números.”

Ler N1

Ler N2

Ler OPERACAO (que será do tipo caractere)

**Caso** OPERACAO:

“+” **então** RESULT = N1 + N2

“-” **então** RESULT = N1 - N2

“\*” **então** RESULT = N1 \* N2

“/” **então** RESULT = N1 / N2

**caso contrario** exibir “Operação não válida!”

**FimCaso**

# ESTRUTURAS DE REPETIÇÃO

---

- ▶ Com as instruções de **repetição** pode-se fazer com que o algoritmo repita um conjunto de instruções, baseado numa decisão lógica.
- ▶ Repetir...Enquanto
- ▶ Enquanto...Repetir
- ▶ Para...ate...repetir

# Enquanto...Repetir

---

- ▶ Sintaxe

**Enquanto** <condição> **repetir**

<instruções>

**FimEnquanto**

- ▶ Usado para repetir um conjunto de ações baseado numa condição.
- ▶ A repetição só ocorre se a condição for verdadeira.
- ▶ A condição é testada no início da estrutura de repetição.
- ▶ Significado: Enquanto a <condição> for verdadeira, repetir as <instruções>.

# Enquanto...Repetir

---

- ▶ **Ler 50 números fornecidos pelo usuário e calcular e exibir a média.**

```
Real: soma, num, media;  
Inteiro: cont;  
Soma = 0;  
Cont = 0;  
Enquanto (cont < 50) repetir  
    ler num;  
    soma = soma + num;  
    cont = cont + 1;  
Fim Enquanto  
Media = soma/cont;  
Imprimir (media);
```

# Repetir...Enquanto

---

## **Repetir**

<instruções>

**Enquanto** (condição)

**FimRepetir**

- ▶ Usado para repetir um conjunto de ações baseado numa condição.
- ▶ A repetição só ocorre se a condição for verdadeira.
- ▶ A condição é testada no fim da estrutura de repetição.
- ▶ Significado: Repete as <instruções>, enquanto a <condição> for verdadeira.

# Repetir...Enquanto

---

- ▶ **Ler 50 números fornecidos pelo usuário e calcular e exibir a média.**

```
Real: soma, num, media;
```

```
Inteiro: cont;
```

```
Soma = 0;
```

```
Cont = 0;
```

```
Repetir
```

```
    ler num;
```

```
    soma = soma + num;
```

```
    cont = cont + 1;
```

```
Enquanto (cont < 50)
```

```
FimRepetir
```

```
Media = soma/cont;
```

```
Imprimir (media);
```

# Para...até...repetir

---

- ▶ **Formato:**

**Para** <variável> = <valor inicial> **até** <valor final> **repetir**  
<ações>

- ▶ **Significado:** A <variável> é inicializada com <valor inicial>. Após cada execução das <ações>, soma-se 1 à <variável> e repete-se as <ações>, continuando assim até que a <variável> atinja o <valor final>.
- ▶ Cria um contador automático, que nós não precisamos mandar incrementar. Ao usar esta estrutura já está subentendido que a <variável> inicia com <valor inicial> e é incrementada a cada ciclo (podendo-se inclusive aproveitar seu valor dentro do ciclo), e que as <ações> serão repetidas até que a <variável> tenha o <valor final>.

# Para...até...repetir

---

- ▶ **Ler 50 números fornecidos pelo usuário e calcular e exibir a média.**

Real: soma, num, media

Inteiro: cont

Soma = 0

Para cont =1 até 50 repetir

    ler num

    soma = soma + num

FimPara

Media = soma / cont

Mostrar media

# Para...até...repetir

---

- ▶ Mostrar os dobros dos inteiros de 3 a 11.

**Para CONT = 3 até 11 repetir**

Imprime (CONT \* 2)



Isto será repetido 9 vezes.

- ▶ Usamos esta estrutura quando sabemos quantas vezes temos de repetir certas ações, mesmo que o número de vezes só seja conhecido *durante a execução*. Por exemplo:  
“Perguntar ao usuário de quantos valores ele quer tirar uma média. Ler os números e calcular a média.”

# Exercícios

---

- ▶ Faça um pseudocódigo para calcular a média de notas de alunos em uma turma de 35 alunos.
- ▶ Utilizando a estrutura **PARA...ATÉ... REPETIR...**, construa um pseudocódigo que exiba os números ímpares entre 100 e 200. Dica: a função  $\text{mod}(X, Y)$  retorna o resto da divisão de  $X$  por  $Y$ .
- ▶ Construa um pseudocódigo que calcule a média de  $N$  números reais informados pelo usuário, onde o valor de  $N$  também é informado no início da execução. Antes do término da execução do algoritmo, mostre a média calculada.

# Exercícios

---

- ▶ Construa um algoritmo (pseudocódigo) que leia 500 valores inteiros e positivos e encontre e mostre o maior valor, o menor valor. Ao final calcule e mostre a média dos números lidos.
- ▶ Faça um algoritmo que mostre a tabuada do número 5.
- ▶ Faça um algoritmo que mostre a tabuada de qualquer número escolhido pelo usuário (considerar tabuada do número 1 ao 10)

# Exercícios

---

- ▶ Construa um algoritmo (pseudocódigo e fluxograma) que determine e mostre o maior número entre vários fornecidos pelo usuário. A condição de parada é a entrada de um valor 0 (zero), ou seja, o algoritmo deve ficar escolhendo o maior até que a entrada seja igual a 0 (zero). Utilize a estrutura ENQUANTO ... REPETIR ... no algoritmo. Assuma que valores negativos nunca serão informados pelo usuário.