

REVISÃO - PROGRAMAÇÃO PARA COMPUTADORES

1. Tipos de Dados

Define o tipo de valor que uma variável pode armazenar.

Palavra	Significado	Exemplo
<code>inteiro</code>	Números inteiros (positivos ou negativos).	<code>inteiro idade = 20</code>
<code>real</code>	Números decimais (ponto flutuante).	<code>real altura = 1.75</code>
<code>caracter</code>	Um único caractere (entre aspas simples).	<code>caracter letra = 'A'</code>
<code>cadeia</code>	Texto (entre aspas duplas).	<code>cadeia nome = "João"</code>
<code>logico</code>	Valores booleanos (<code>verdadeiro</code> ou <code>falso</code>).	<code>logico ativo = verdadeiro</code>

2. Valores Lógicos

Usados em expressões condicionais.

Palavra	Significado	Exemplo
<code>verdadeiro</code>	Valor lógico verdadeiro.	<code>se (10 > 5) então ...</code>
<code>falso</code>	Valor lógico falso.	<code>logico erro = falso</code>

REVISÃO - PROGRAMAÇÃO PARA COMPUTADORES

3. Estruturas de Controle

Comandos para controle de fluxo do programa.

Palavra	Significado	Exemplo
<code>se</code>	Inicia uma condição (if).	<code>se (idade >= 18) então ...</code>
<code>senao</code>	Executa se a condição do <code>se</code> for falsa (else).	<code>senao escreva("Menor de idade")</code>
<code>escolha</code>	Estrutura de múltiplas condições (switch-case).	<code>escolha (opcao) { caso 1: ... }</code>
<code>caso</code>	Define um caso dentro do <code>escolha</code> (case).	<code>caso 1: escreva("Opção 1")</code>
<code>pare</code>	Sai de um <code>caso</code> ou loop (break).	<code>caso 2: escreva("Sair"); pare</code>
<code>para</code>	Loop com contador (for).	<code>para (inteiro i = 0; i < 10; i++)</code>
<code>enquanto</code>	Loop condicional (while).	<code>enquanto (x < 10) faca ...</code>
<code>faca</code>	Usado com <code>enquanto</code> (do-while).	<code>faca { ... } enquanto (condicao)</code>

4. Funções e Procedimentos

Usados para modularizar o código.

Palavra	Significado	Exemplo
<code>funcao</code>	Define uma função que retorna um valor.	<code>funcao inteiro soma(inteiro a, b)</code>
<code>retorne</code>	Retorna um valor de uma função (return).	<code>retorne a + b</code>
<code>vazio</code>	Indica que uma função não retorna nada (void).	<code>funcao vazio imprimir()</code>

REVISÃO - PROGRAMAÇÃO PARA COMPUTADORES

5. Entrada e Saída

Comandos para interação com o usuário.

Palavra	Significado	Exemplo
<code>leia</code>	Lê um valor do usuário (input).	<code>leia(idade)</code>
<code>escreva</code>	Exibe um valor na tela (print).	<code>escreva("Olá, mundo!")</code>

6. Diretivas e Bibliotecas

Usados para incluir bibliotecas e constantes.

Palavra	Significado	Exemplo
<code>inclua</code>	Inclui uma biblioteca (include).	<code>inclua biblioteca Matematica</code>
<code>biblioteca</code>	Define o uso de uma biblioteca externa.	<code>Matematica.raiz(16)</code>
<code>constante</code>	Define um valor fixo (const).	<code>constante PI = 3.14</code>

REVISÃO - PROGRAMAÇÃO PARA COMPUTADORES

```
inclua biblioteca Matematica

funcao real calcularArea(real raio) {
    constante PI = 3.1415
    retorne PI * Matematica.potencia(raio, 2)
}

funcao vaziao main() {
    escreva("Digite o raio do círculo: ")
    real raio
    leia(raio)

    se (raio > 0) então {
        real area = calcularArea(raio)
        escreva("Área: ", area)
    } senao {
        escreva("Raio inválido!")
    }
}
```

ESTRUTURA DO PORTUGOL STUDIOã

A estrutura básica de um programa em Portugol segue uma classificação definida por programas e métodos, que organizam o conteúdo do Sistema. Cada parte do programa tem uma função específica.

Passos e espaços do corpo Portugol Studio:

1. Declaração do início do programa e da função principal (início):

```
programa {
    funcao inicio () {
```

//aqui dentro da função inicio será escrito o nosso código

REVISÃO - PROGRAMAÇÃO PARA COMPUTADORES

```
}  
  
}
```

2. Declaração de variável(éis)

```
programa {
```

```
funcao inicio () {
```

```
    //declaração de variaveis
```

```
    inteiro soma = 0
```

-> *variavel do tipo inteiro (ex. 4)*

```
    real valor = 0
```

-> *variavel do tipo real (ex. 4.5)*

```
    cadeia nomeDeAlgo
```

-> *variavel do tipo cadeia (ex. nome2)*

```
    }  
  
}
```

3. Funções:

```
programa {
```

```
funcao inicio () {
```

```
    //declaração de variaveis
```

```
    inteiro soma = 0
```

-> *variavel do tipo inteiro (ex. 4)*

```
    real valor = 0
```

-> *variavel do tipo real (ex. 4.5)*

```
    cadeia nomeDeAlgo  
nome2)
```

-> *variavel do tipo cadeia (ex.*

REVISÃO - PROGRAMAÇÃO PARA COMPUTADORES

// Informe um número inteiro e exiba ele para o usuário

escreva("Informe um número")

leia(soma)

escreva("O número informado é: " + soma)

}

}