



ESTRUTURA DE DADOS HOMOGÊNEAS

(VETOR - MATRIZ)



Portugol Studio

VAMOS AUMENTAR O NÍVEL DE NOSSA IDE

Criar desde algoritmos simples até jogos completos, totalmente em português. A linguagem é baseada em C e PHP, e simplifica muitas funcionalidades gráficas e matemáticas.

```
programa
{
    funcao inicio()
    {
        inteiro valor1, valor2, soma, sub, mult, resto
        real div

        escreva("Digite dois valores inteiros: ")
        leia(valor1, valor2)

        soma = valor1 + valor2
        sub = valor1 - valor2
        mult = valor1 * valor2
        div = valor1 / valor2
        resto = valor1 % valor2

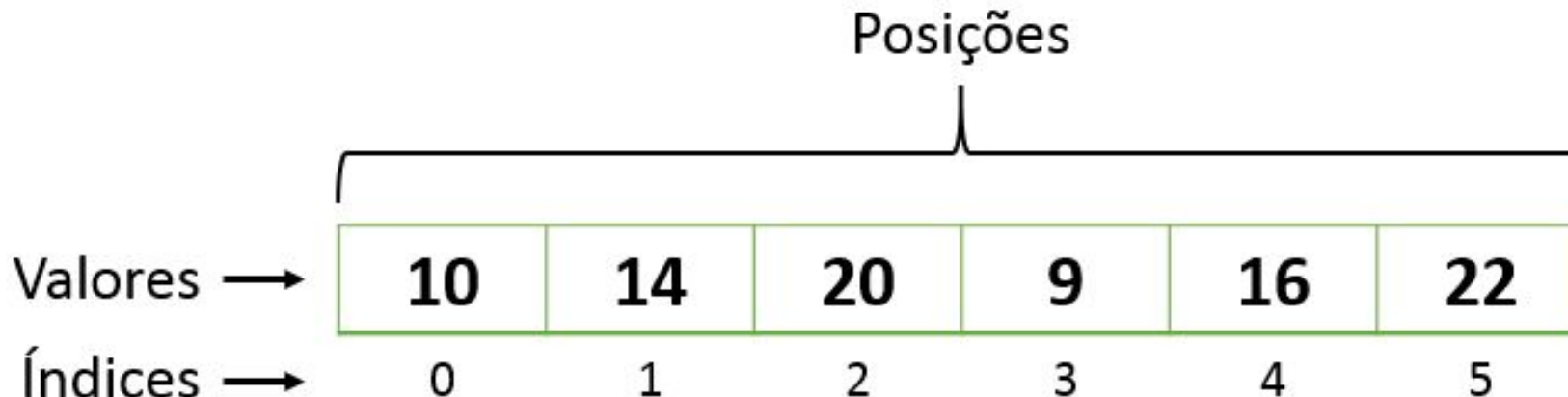
        escreva("\nSoma de ", valor1, " + ", valor2, " = ", soma)
    }
}
```

programa

```
{  
  
    funcao inicio()  
    {  
        inteiro valor1, valor2, soma, sub, mult, resto  
        real div  
  
        escreva("Digite dois valores inteiros: ")  
        leia(valor1, valor2)  
  
        soma = valor1 + valor2  
        sub = valor1 - valor2  
        mult = valor1 * valor2  
        div = valor1 / valor2  
        resto = valor1 % valor2  
  
        escreva("\nSoma de ", valor1, " + ", valor2, " = ", soma)  
    }  
}
```

VETOR (ARRAY UNIDIMENSIONAL)

- Aprender a trabalhar com conjunto de dados
 - ex: calcular a média de nota de uma turma com 35 alunos, podemos criar uma variável que contenha 35 slots de memória.
- São chamadas de vetor ou array unidimensional (Possuem apenas uma dimensão, representadas por uma linha ou coluna)



CRIANDO UM VETOR

- Declaração:
 - Vetor é um conjunto de elementos do mesmo tipo.
 - Além de seu tipo e do identificador precisamos informar sua capacidade.

inteiro vetor[35]

inteiro vetor[3] = {10, 54, 12}

Como percorrer um vetor?

- Cada elemento possui um índice de acesso:
 - No português e assim como na maioria das linguagens de programação o primeiro índice **SEMPRE** está na posição 0.
 - exemplo:

Inteiro vetor[3] = {10, 54, 12}

0 1 2

Concluimos que: todo vetor de tamanho N terá com índices válidos **N-1**


```
1. programa{
2.
3.     funcao inicio(){
4.         inteiro indice, vetIdade[5] = {16,25,32,45,87}
5.         real notas[5] = {4.5,5.8,8.3,9.2,10.0}
6.         caracter sexo[4] = {'f','F','m','M'}
7.
8.         escreva("Idades: ")
9.         // imprime o vetor vetIdades
10.        para(indice = 0; indice < 5; indice++)
11.            escreva(vetIdade[indice], ", ")
12.
13.        escreva("\n\nNotas: ")
14.        // imprime o vetor notas
15.        para(indice = 0; indice < 5; indice++)
16.            escreva(notas[indice], ", ")
17.
18.        escreva("\n\nVetor sexo: ")
19.        // imprime o vetor sexo
20.        para(indice = 0; indice < 4; indice++)
21.            escreva(sexo[indice], ", ")
22.
23.    }
24. }
```

Como ler um vetor?

```
1. inteiro valor
2.
3. escreva("Digite um número: ")
4. leia(valor)
```

```
1. inteiro i, meuVetor[10]
2.
3. para(i = 0; i < 10; i++){
4.     escreva("Digite um número: ")
5.     leia(meuVetor[i])
6. }
```

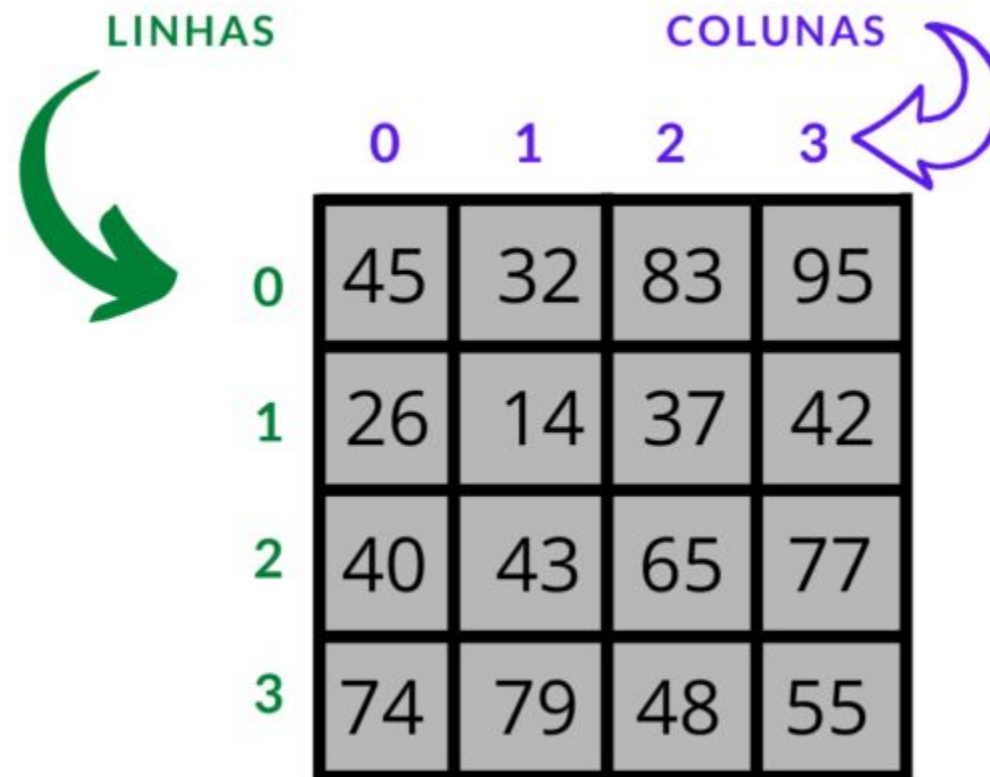


```
1. programa{
2.
3.     funcao inicio(){
4.         real notas[10]
5.         inteiro i
6.
7.         // Lê o vetor digitado pelo usuário
8.         para(i = 0; i < 10; i++){
9.             escreva("Digite a nota da posição ", i, ": ")
10.            leia(notas[i])
11.        }
12.
13.        // imprime o vetor lido
14.        para(i = 0; i < 10; i++)
15.            escreva(notas[i], ", ")
16.    }
17. }
```

Desafio: Crie dois Arrays um que colete a idade de todos em sala de aula e outro que colete o nome. Imprima as idades e os nomes de cada um.

MATRIZ (ARRAY BIDIMENSIONAL)

- Possui duas dimensões: Linhas x Colunas



The diagram illustrates a 4x4 matrix with row and column indices. A green arrow labeled 'LINHAS' points to the row indices (0, 1, 2, 3) on the left. A purple arrow labeled 'COLUNAS' points to the column indices (0, 1, 2, 3) on the top. The matrix cells contain the following values:

	0	1	2	3
0	45	32	83	95
1	26	14	37	42
2	40	43	65	77
3	74	79	48	55

MATRIZ (ARRAY BIDIMENSIONAL)

- Criando uma matriz:

```
1. // Criando uma matriz de inteiros chamada mat com 7 linhas e 3 colunas
2. inteiro mat[7][3]
```

- Como preencher uma matriz:

```
3. Primeira linha: {34,56}
4. Segunda linha: {87,90}
5. Terceira linha: {25,58}
6.
7. Primeira coluna: 34, 87, 25
8. Segunda coluna: 56, 90, 58
9. */
10. inteiro mat[3][2] = {{34,56},{87,90},{25,58}}
```

MATRIZ (ARRAY BIDIMENSIONAL)

- Como percorrer as posições de uma matriz:

```
1. inteiro l, c, mat[3][2] = {{34, 56}, {87, 90}, {25, 58}}
2.
3. para(l = 0; l < 3; l++){ // índice das linhas
4.     para(c = 0; c < 2; c++){ // imprime a linha saltando as colunas
5.         escreva(mat[l][c], " ") // imprime os elementos da matriz na posição l c
6.     }
7.     escreva("\n") // para saltar uma linha
8. }
```

MATRIZ (ARRAY BIDIMENSIONAL)

- Como ler os valores de uma matriz:

diferença que agora teremos **dois índices**, um para linha e outro para coluna, assim:

```
1. inteiro l, c, mat[3][2]
2.
3. para(l = 0; l < 3; l++){
4.     para(c = 0; c < 2; c++){
5.         escreva("Digite um valor: ")
6.         leia(mat[l][c])
7.     }
8. }
```



```
1. programa{
2.
3.     funcao inicio(){
4.
5.         inteiro l, c, mat[3][3] = {{45,86,95},{71,42,36},{49,16,35}}
6.
7.         // Lê uma matriz 3 por 3 do teclado
8.         para(l = 0; l < 3; l++){
9.             para(c = 0; c < 3; c++){
10.                 escreva("Digite o valor da posição ", l, " ", c, ": ")
11.                 leia(mat[l][c])
12.             }
13.         }
14.
15.         // Imprime uma matriz 3 por 3
16.         para(l = 0; l < 3; l++){
17.             para(c = 0; c < 3; c++){
18.                 escreva(mat[l][c], " ")
19.             }
20.             escreva("\n")
21.         }
22.     }
23. }
```

ATIVIDADE AVALIATIVA

Faça um programa em Portugol para SOMAR as matrizes A e B e salvar na matriz C

Importância da lógica de programação

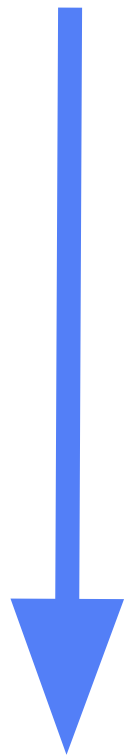
A lógica de programação é essencial para a construção de algoritmos e solução de problemas. Ela ajuda a organizar ideias e a estruturar o pensamento de forma lógica, facilitando o desenvolvimento de programas.

Conceitos fundamentais da lógica de programação

Os conceitos fundamentais incluem sequência, seleção e repetição. Compreender esses conceitos é crucial para a criação de algoritmos eficientes e compreensão dos processos de programação.

Algoritmos

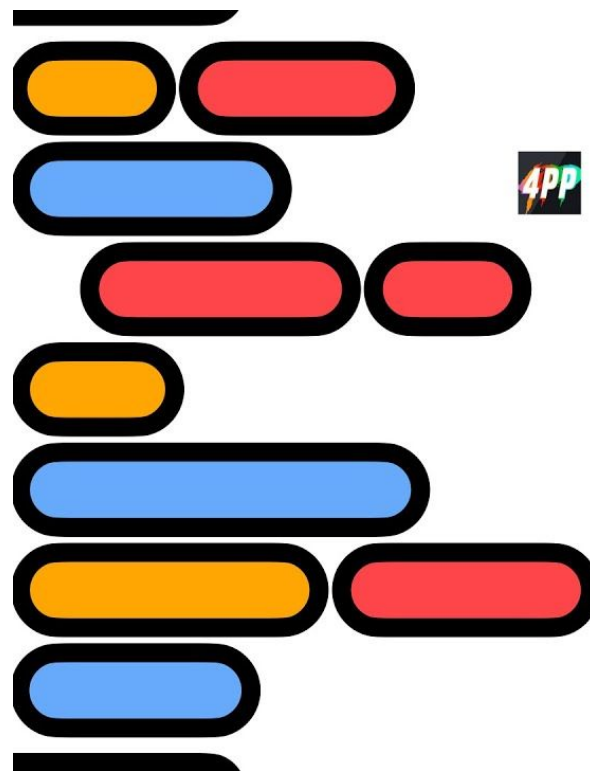
Task: Estudar na faculdade de manhã



Sequência	Ação
Primeiro	Acordar
Segundo	Tomar Banho
Terceiro	Vestir-Se
Quarto	Tomar Café
Quinto	Escovar Os Dentes
Sexto	Ir Para A Faculdade

Pseudocódigo

- O intermediário entre a linguagem falada e a linguagem de programação (Ex: javascript, java, python, c++).
- Praticamente um programa escrito em português que, depois, podemos passar para o computador.
- Fáceis de se interpretar e codificar.



**O QUE É
PSEUDO
CÓDIGO?**

Vamos conhecer alguns comando básicos?

- **algoritmo** - palavra usada para indicar o início do programa.
- **inicio** - palavra usada para iniciar o programa principal.
- **var** - palavra usada para declarar variáveis.
- **escreva(" ")** - comando usado para imprimir uma mensagem na tela.
- **leia ()** - comando usado para ler valores digitados no teclado.
- **<-** - comando de atribuição.
- **fimalgortimo** - palavra usada para finalizar o algoritmo.

Programa como um todo

1. **algoritmo** "SomarDoisNumeros"
2. **var**
3. n1, n2, soma: real
- 4.
5. **inicio**
6. **escreva**("Digite um número")
7. **leia** (n1)
8. **escreva**("Digite um número")
9. **leia** (n1)
10. soma **<-** n1 + n2
11. **escreva**("Soma =",soma)
- 12.
13. **fimalgortimo**

Ambiente de declaração de variável

programa principal

Comando de Controle de Programa

- **Usado em nossos programas para direcionar o fluxo de sua execução.**
- **Desvio Condicional: IF**
 - desvia o programa pela avaliação de uma condição.
 - utilizado sempre quando precisamos analisar uma situação e decidir o que vamos fazer em seguida.
- **Exemplo clássico: queremos desenvolver um programa que:**
 - Solicite as notas das duas provas bimestrais dos estudantes.
 - Calcule a média dessas notas.
 - Verifique a situação do estudante:
 - Aprovado se a média for maior ou igual a 7.
 - Recuperação se a média for menor que 7 e maior que 4.
 - Reprovado se a média for menor que 4.
 - Apresente na tela os resultados.

Comando IF Simples

```
algoritmo "se_simples"
var
    n1, n2, media : real
inicio
    escreva(" Digite a nota da primeira prova: ")
    leia(n1)
    escreva(" Digite a nota da segunda prova: ")
    leia(n2)
    media <- (n1+n2)/2
    se media >= 7 entao
        escreval(" A nota da primeira prova do estudante é: ", n1)
        escreval(" A nota da segunda prova do estudante é: ", n2)
        escreval(" A média do estudante é: ", media)
        escreval(" O estudante está APROVADO ")
    fimse
    escreval(" FIM DO PROGRAMA ")
finalgoritmo
```

```
algoritmo "se_aninhado"
```

```
var
n1, n2, media : real
```

```
inicio
escreva(" Digite a nota da primeira prova: ")
leia(n1)
escreva(" Digite a nota da segunda prova: ")
leia(n2)
media <- (n1+n2)/2
```

```
    se (media >= 7) entao
        escreval(" A nota da primeira prova do estudante é: ", n1)
        escreval(" A nota da segunda prova do estudante é: ", n2)
        escreval(" A média do estudante é: ", media)
        escreval(" O estudante está APROVADO ")
```

```
    senao
        se (media >= 4) entao
            escreval(" A nota da primeira prova do estudante é: ", n1)
            escreval(" A nota da segunda prova do estudante é: ", n2)
            escreval(" A média do estudante é: ", media)
            escreval(" O estudante está de RECUPERAÇÃO ")
```

```
    senao
        escreval(" A nota da primeira prova do estudante é: ", n1)
        escreval(" A nota da segunda prova do estudante é: ", n2)
        escreval(" A média do estudante é: ", media)
        escreval(" O estudante está REPROVADO ")
```

```
    fimse
```

```
    fimse
```

```
escreval(" FIM DO PROGRAMA ")
finalgoritmo
```

Sintaxe:

Se condição então

comandos

Senão

Se condição então

comandos

Senão

comandos

fimSe //fim do senão

fimSe //fim do se principal

Desafio

Abra seu visualg ou use uma folha de caderno para desenvolver as questões:

1 - Crie uma sequência lógica detalhada para trocar um pneu de carro furado (Folha)