

AULA 8

VETORES

Faça um algoritmo que  
leia 5 números e  
mostre os 5 na tela

**PRÁTICA**

```
1 algoritmo "exemplo_vetor_inicializacao"
2 var
3 // Declara um vetor chamado 'numeros' com 5 posições
4 inicio
5     // Inicializando posições específicas do vetor
6     numeros[1] <- 10
7     numeros[2] <- 20
8     numeros[3] <- 30
9     numeros[4] <- 40
10    numeros[5] <- 50
11
12    // Mostrando o valor de algumas posições
13    escreval("O valor na posição 1 é: ", numeros[1])
14    escreval("O valor na posição 3 é: ", numeros[3])
15    escreval("O valor na posição 5 é: ", numeros[5])
16
17 fimalgoritmo
```

```
1 algoritmo "exemplo_vetor_loop"
2 var
3     // Vetor para armazenar 4 notas de alunos
4     notas: vetor [1..4] de real
5     // Variável para usar como índice no loop
6     i: inteiro
7
8 inicio
9     // Lendo as notas para o vetor usando um loop
10    para i de 1 ate 4 faca
11        escreva("Digite a nota ", i, ": ")
12        leia(notas[i])
13    fimpara
14
15    escreval // Pula uma linha para melhor formatação
16
17    // Mostrando as notas armazenadas no vetor
18    escreval("As notas informadas são:")
19    para i de 1 ate 4 faca
20        escreval("Nota ", i, ": ", notas[i])
21    fimpara
22
23 fimalgoritmo
```

Crie um **vetor** lista de  
compras com 4  
posições, insira 4 itens  
no vetor e mostre a lista  
formatada

```
1 algoritmo "lista_compras_predefinida"
2 var
3   // Declara um vetor para 4 itens c(aracteres)
4   listaproductos: vetor [1..4] de caractere
5   i: inteiro // Variável para usar no loop de exibição
6 inicio
7   // --- Inserindo os 4 itens no vetor ---
8   listaproductos[1] <- "Pão"
9   listaproductos[2] <- "Leite"
10  listaproductos[3] <- "Ovos"
11  listaproductos[4] <- "Café"
12
13  // --- Exibindo a lista de compras formatada ---
14  escreval("--- Minha Lista de Compras ---")
15
16  para i de 1 ate 4 faca
17    // Mostra cada item da lista usando o índice
18    escreval("- ", listaproductos[i])
19  fimpara
20
21 fimalgoritmo
```

Altere o algoritmo para  
receber os dados do  
usuário



```

1 algoritmo "lista_de_compras_usuario" // Nome alterado
2 var
3     listaprodutos: vetor [1..4] de caractere // Declara
4     i: inteiro // Variável para controlar os loops
5 inicio
6     // --- Entrada de dados: Preenchendo a lista de compras
7     escreval("--- Montando sua Lista de Compras (4 itens):")
8
9     para i de 1 ate 4 faca
10         escreva("Digite o nome do item ", i, ": ")
11         leia(listaprodutos[i]) // Lê o nome do item digitado
12     fimpara
13
14     // --- Saída de dados: Exibindo a lista de compras
15     escreval // Pula uma linha para organizar a saída
16     escreval("--- Sua Lista de Compras ---")
17
18     para i de 1 ate 4 faca
19         escreval("- ", listaprodutos[i]) // Exibe cada item
20     fimpara
21
22 fimalgoritmo

```

# Exercícios

1. **Exercício 1 (Fácil):** Crie um algoritmo que declare um vetor de 3 posições para armazenar números inteiros. Atribua os valores 10, 20 e 30 a cada posição e depois exiba apenas o valor da segunda posição.

2. **Exercício 2 (Fácil):** Desenvolva um algoritmo que declare um vetor de 5 posições para nomes (caractere). Peça ao usuário para digitar 5 nomes e armazene-os no vetor. Ao final, mostre a lista completa de nomes digitados.

3. **Exercício 3 (Fácil):** Faça um algoritmo que crie um vetor de 4 posições para números reais. Leia 4 números do usuário e armazene-os no vetor. Calcule e mostre a soma de todos os números no vetor.

4. **Exercício 4 (Médio):** Crie um algoritmo com um vetor de 6 posições para números inteiros. Peça ao usuário para preencher o vetor. Encontre e exiba o maior valor armazenado no vetor.

5. **Exercício 5 (Médio):** Desenvolva um algoritmo que declare um vetor de 7 posições para idades (inteiro). Leia as 7 idades. Conte e mostre quantas pessoas têm idade superior a 18 anos.

6. **Exercício 6 (Médio/Difícil):** Faça um algoritmo que declare dois vetores de 5 posições para números inteiros (VetorA e VetorB). Preencha ambos os vetores com valores lidos do usuário. Crie um terceiro vetor (VetorC) onde cada posição seja a soma da posição correspondente em VetorA e VetorB ( $C[i] = A[i] + B[i]$ ). Mostre o VetorC.



