

# Etapa 6 — Vetores e Matrizes (Arrays and Matrices)

## Trilha: Lógica de Programação

### Introdução

Em muitos algoritmos, precisamos **armazenar vários valores relacionados**, como notas de alunos, preços de produtos ou números em uma sequência.

Fazer isso criando várias variáveis diferentes é ineficiente e confuso.

Para resolver esse problema, usamos **estruturas de dados (data structures)** chamadas **Vetores (Arrays) e Matrizes (Matrices)**.

Elas permitem **guardar e acessar vários valores** dentro de uma única variável organizada.

### O que é um Vetor (Array)?

Um **vetor** é uma **coleção linear (one-dimensional)** de elementos do mesmo tipo.

Cada valor é identificado por um **índice (index)**, que indica sua posição dentro do vetor.

**Exemplo conceitual:**

Índice	1	2	3	4	5
Valor	10	8	9	7	6

**Observação:** Em algumas linguagens, os índices começam em **0**; em outras, começam em **1**.

### Declaração e Uso de Vetores

**Sintaxe genérica:**

```
declare notas[5] como real
```

### **Atribuição e acesso:**

```
notas[1] ← 8.5
```

```
escreva notas[1]
```

### **Saída:**

```
8.5
```

Cada posição do vetor pode ser **lida, alterada ou usada em cálculos.**

## **Preenchendo e Lendo um Vetor**

É comum utilizar **estruturas de repetição (loops)** para preencher e percorrer vetores.

### **Exemplo:**

```
para i de 1 ate 5 faca
```

```
    escreva "Digite a nota ", i, ":"
```

```
    leia notas[i]
```

```
fimpara
```

### **Exibir todas as notas:**

```
para i de 1 ate 5 faca
```

```
    escreva "Nota ", i, " = ", notas[i]
```

```
fimpara
```

**Dica:** Sempre utilize loops para lidar com vetores de tamanho variável ou desconhecido.

## **Operações Comuns com Vetores**

- Calcular a soma dos elementos.
- Calcular a média dos valores.
- Identificar o maior ou menor elemento.

### **Exemplo:**

```
soma ← 0  
para i de 1 ate 5 faca  
    soma ← soma + notas[i]  
fimpara  
media ← soma / 5  
escreva "Média = ", media
```

### **Saída (exemplo):**

Média = 8.2

## **O que é uma Matriz (Matrix)?**

Uma **matriz** é uma estrutura **bidimensional (two-dimensional)** — uma **tabela** de valores organizados em **linhas (rows)** e **colunas (columns)**.

### **Exemplo visual:**

	Coluna 1	Coluna 2	Coluna 3
Linha 1	5	8	9
Linha 2	2	4	6

Cada elemento é identificado por **dois índices**: um para a linha e outro para a coluna.

## **Declaração e Acesso de Matrizes**

### **Sintaxe genérica:**

```
declare matriz[2][3] como inteiro
```

**Atribuição e leitura:**

```
matriz[1][2] ← 8
```

```
escreva matriz[1][2]
```

**Saída:**

```
8
```

**Importante:**

O **primeiro índice** sempre representa a **linha**, e o **segundo** representa a **coluna**.

## Preenchendo e Exibindo uma Matriz

**Exemplo:**

```
para i de 1 ate 2 faca
```

```
    para j de 1 ate 3 faca
```

```
        escreva "Digite o valor [", i, ",", j, "]:"
```

```
        leia matriz[i][j]
```

```
    fimpara
```

```
fimpara
```

**Exibir matriz:**

```
para i de 1 ate 2 faca
```

```
    para j de 1 ate 3 faca
```

```
        escreva matriz[i][j], " "
```

```
    fimpara
```

```
    escreva ""
```

```
fimpara
```

**Uso típico:** notas de alunos, tabelas, grades de jogos, mapas e imagens digitais.

## Operações Comuns com Matrizes

1. Somar ou subtrair duas matrizes.
2. Calcular médias de linhas e colunas.
3. Procurar um valor específico na tabela.

### Exemplo de soma de matrizes:

declare A[2][2], B[2][2], C[2][2] como inteiro

para i de 1 ate 2 faca

    para j de 1 ate 2 faca

        C[i][j] ← A[i][j] + B[i][j]

    fimpara

fimpara

### Resultado:

C contém a soma elemento a elemento de A e B.

## Diferença Entre Vetor e Matriz

Estrutura	Dimensões	Exemplo de acesso	Analogia
Vetor	1D	notas[3]	Linha
Matriz	2D	matriz[2][3]	Tabela

### Resumo:

Vetor = linha única de dados.

Matriz = tabela com linhas e colunas.

## Erros Comuns

- Acessar índices fora do limite definido.
- Confundir a ordem dos índices (linha e coluna).
- Não inicializar os elementos antes de usar.

Sempre **defina o tamanho corretamente** e teste seus loops para evitar falhas.

## Exemplo Completo

algoritmo "notas\_alunos"

inicio

declare notas[5] como real

para i de 1 ate 5 faca

escreva "Digite a nota ", i, ":"

leia notas[i]

fimpara

soma ← 0

para i de 1 ate 5 faca

soma ← soma + notas[i]

fimpara

media ← soma / 5

escreva "Média = ", media

fim

**Saída (exemplo):**

Média = 8.2

# Resumo da Etapa

- **Vetores** → armazenam uma sequência linear de dados.
- **Matrizes** → armazenam dados bidimensionais.
- **Loops (for/while)** → usados para preencher e percorrer estruturas.
- **Índices** → controlam o acesso aos elementos.
- **Evite erros** testando os limites e inicializando variáveis.

## Conclusão:

“Vetores e matrizes tornam o algoritmo mais organizado, eficiente e escalável.”

## Perguntas de Múltipla Escolha

### 1. O que é um vetor?

- a) Um conjunto de variáveis independentes
- b) Uma estrutura que armazena um único valor
- c) Uma sequência de elementos do mesmo tipo
- d) Uma matriz com várias colunas

### 2. Como se acessa o terceiro elemento de um vetor?

- a) vetor(3)
- b) vetor{3}
- c) vetor[3]
- d) vetor<3>

### 3. Qual a principal diferença entre vetor e matriz?

- a) Vetor é tridimensional
- b) Matriz armazena dados em duas dimensões
- c) Vetor é mais lento
- d) Matriz só armazena números inteiros

### 4. Qual comando é mais usado para percorrer vetores e matrizes?

- a) se...entao
- b) enquanto
- c) para
- d) escolha

**5. O que acontece ao acessar notas[6] em um vetor declarado como notas[5]?**

- a) O programa repete a execução
- b) O algoritmo entra em loop infinito
- c) Erro: índice fora dos limites
- d) O valor é lido como zero

**Gabarito**

- 1. c
- 2. c
- 3. b
- 4. c
- 5. c