

INSTITUTO FEDERAL  
BAIANO

# Algoritmos e Introdução à Programação

---

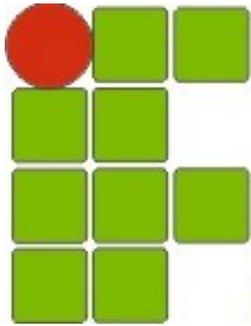
## Lógica e Linguagem de Programação

*Prof. José Honorato Ferreira Nunes*

**honoratonunes@softwarelivre.org**

**<http://softwarelivre.org/zenorato/honoratonunes>**





INSTITUTO FEDERAL  
BAIANO

# Estruturas Homogêneas

---

## Matrizes Unidimensionais (Vetores)

*Prof. José Honorato Ferreira Nunes*  
**honoratonunes@softwarelivre.or**



# Estruturas Homogêneas – Matrizes Unidimensionais (Vetores)

---

Podemos definir um Vetor como uma variável dividida em vários "pedaços", em várias "casinhas", onde cada pedaço desses é identificado através de um número, referente à posição de uma determinada informação no vetor em questão.

O número de cada posição do vetor é chamado de índice.

# Estruturas Homogêneas – Matrizes Unidimensionais (Vetores)

---

Conceito: Vetor é um conjunto de variáveis, onde cada uma pode armazenar uma informação diferente, mas todas compartilham o mesmo nome.

São associados índices a esse nome, que representam as posições do vetor, permitindo assim, individualizar os elementos do conjunto.

Podemos imaginar que na memória do computador o vetor seja mais ou menos da seguinte forma:



## Vetores - declaração

---

Todos os elementos de um vetor pertencem necessariamente ao mesmo tipo de dado. Esta é a essência das estruturas de dados homogêneas.

No C, um vetor é declarado com a seguinte estrutura:

```
int valores[10];
```

```
char locais[5];
```

## Vetores – atribuição e leitura

---

Para acessar (atribuir ou ler) um determinado elemento dentro do vetor, é necessário informar sua posição, também chamado de índice, por meio da seguinte sintaxe:

*{para atribuir}*

*nomeVetor[índice] = valor;*

*valores[0] = 10;*

*{para ler}*

*nomeVariavel=nomeVetor[índice];*

*nota = valores[0];*

# Vetores - exemplos

---

```
#include <stdio.h>
```

```
main () {
```

```
    int x, valores[10];
```

```
    for (x=0;x<=9;x++){
```

```
        printf("Informe um valor para posição %d: ", x);
```

```
        scanf("%d", &valores[x]);
```

```
    }
```

```
    for (x=0;x<=9;x++){
```

```
        printf("\n O elemento da posição %d vale: %d", x, valores[x]);
```

```
    }
```

```
}
```

# Atividades

---

- Crie um algoritmo que solicite ao usuário 10 valores inteiros, armazenando os dados em um vetor. Em seguida, o algoritmo deverá percorrer o vetor escrevendo na tela os valores armazenados nas posições ímpares.
- Escreva um algoritmo que solicite ao usuário 10 valores inteiros, armazenando os dados em um vetor. Em seguida escreva na tela os valores lidos em ordem inversa (da posição 9 até a 0).

# Atividades

---

- Faça um algoritmo que leia um vetor  $C[5]$ . Encontre a seguir o maior elemento de  $C$  e mostre-o.
- Escreva um algoritmo para ler a nota de 8 alunos, calcular a média geral da turma e escrever quantos alunos tiveram a nota acima da média calculada.
- Faça um algoritmo que leia um vetor  $V[6]$ . Conte a seguir, quantos valores de  $V$  são negativos e mostre essa informação.

# Bibliografia

- ❑ MANZANO, Wilson Y. Yamaturni-São Paulo-SP. **Lógica estruturada para programação de computadores**, Ed. Érica 1997 e 2001.
- ❑ MORAES, Celso Roberto. **Estruturas de Dados e Algoritmos**. Ed. Érica, São Paulo
- ❑ LOPES, Anita. **Introdução à programação**. Rio de Janeiro: Campus, 2002.
- ❑ BENEDUZZI, Humberto M. e METZ, João A. **Lógica e Linguagem de Programação – Introdução ao Desenvolvimento de Software (1ª edição)**. Editora do Livro Técnico, 2010.
- ❑ SEBESTA, Robert W. **Conceitos de linguagens de programação**. 9. ed. Porto Alegre: Bookman, 2003.
- ❑ CORMEN, Thomas H. **Algoritmos: teoria e prática**. Rio de Janeiro: Campus, 2002.
- ❑ ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Veneruchi de. **Fundamentos da programação de computadores: algoritmos, Pascal e C/C++ e Java**. 2. ed. São Paulo: Pearson Prentice Hall, 2008.
- ❑ ZIVIANI, Nivio. **Projeto de algoritmos com implementações em Pascal e C**. São Paulo: Pioneira Thomson Learning.