



Universidade Federal do ABC

Bacharelado em Ciência e Tecnologia  
Processamento da Informação



Universidade Federal do ABC

Vetores Unidimensionais

# Vetores Unidimensionais

## Objetivos

- Entender a importância e a necessidade do uso de Vetores
- Definição de Vetores Unidimensionais
- Manipulação de Vetores
  - Inserir elementos em um vetor (usando laços ou não)
  - Acessar elementos de um vetor (usando laços ou não)



Universidade Federal do ABC

# Bacharelado em Ciência e Tecnologia Processamento da Informação

## Vetores Unidimensionais

# PROBLEMA

## Problema 1

- **Calcular a média de uma classe a partir da nota de seus 10 alunos.**

■ **SAÍDA:**

## Problema 1

- **Calcular a média** de uma classe a partir da nota de seus **10** alunos.

■ **SAÍDA:** a média

## Vetores Unidimensionais

### Problema 1

- **Calcular a média de uma classe a partir da nota de seus 10 alunos.**

■ **SAÍDA:** a média

■ **ENTRADA:**

Vetores Unidimensionais

## Problema 1

- Calcular a média de uma classe a partir da nota de seus 10 alunos.

▪ **SAÍDA:** a média

▪ **ENTRADA:** nota

10 vezes

## Vetores Unidimensionais

### Problema 1

- **Calcular a média de uma classe a partir da nota de seus 10 alunos.**

▪ **SAÍDA:** a média

▪ **ENTRADA:** nota

**10 vezes**

▪ **PROCESSAMENTO:**

## Vetores Unidimensionais

### Problema 1

- **Calcular a média de uma classe a partir da nota de seus 10 alunos.**

▪ **SAÍDA:** a média

▪ **ENTRADA:** nota

10 vezes

▪ **PROCESSAMENTO:**

- Acumular a nota (somar com as anteriores)
- Dividir a soma por 10

10 vezes

Vetores Unidimensionais

## Problema 1

- **Calcular a média de uma classe a partir da nota de seus 10 alunos.**

- **LÓGICA (SOLUÇÃO)**

- Para cada um dos 10 alunos:
  - Ler a nota **N**
  - Acumular a nota (somar com as anteriores)
- $\text{Media} = \text{soma} / 10$

<b>N</b>									
?	?	?	?	?	?	?	?	?	?

<b>SOMA</b>
0

<b>MEDIA</b>
0

Vetores Unidimensionais

## Problema 1

- **Calcular a média de uma classe a partir da nota de seus 10 alunos.**

- **LÓGICA (SOLUÇÃO)**

- Para cada um dos 10 alunos:
  - Ler a nota N
  - Acumular a nota (somar com as anteriores)
- $Media = soma / 10$

**ATENÇÃO:** a cada nota digitada vai acumulando com a anterior.  
Pode-se usar a mesma variável “N” para ler a nota do aluno seguinte

N	N	N	N	N	N	N	N	N	N
5.5	?	?	?	?	?	?	?	?	?

SOMA
5.5

MEDIA
0

Vetores Unidimensionais

## Problema 1

- Calcular a média de uma classe a partir da nota de seus 10 alunos.

- LÓGICA (SOLUÇÃO)

- Para cada um dos 10 alunos:
  - Ler a nota N
  - Acumular a nota (somar com as anteriores)
- $Media = soma / 10$

**ATENÇÃO:** a cada nota digitada vai acumulando com a anterior.  
Pode-se usar a mesma variável “N” para ler a nota do aluno seguinte

N	N	N	N	N	N	N	N	N	N
5.5	6.5	?	?	?	?	?	?	?	?

SOMA
11.0

MEDIA
0

**E assim por diante ....**

Vetores Unidimensionais

## Problema 1

- **Calcular a média de uma classe a partir da nota de seus 10 alunos.**

- **LÓGICA (SOLUÇÃO)**

- Para cada um dos 10 alunos:
  - Ler a nota N
  - Acumular a nota (somar com as anteriores)
- $Media = soma / 10$

**ATENÇÃO:** a cada nota digitada vai acumulando com a anterior.  
**Pode-se usar a mesma variável “N” para ler a nota do aluno seguinte**

N	N	N	N	N	N	N	N	N	N
5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0

SOMA
61.0

MEDIA
6.1

## Solução Em Portugol Studio

```
funcao inicio()  
{  
    real nota, soma = 0, media  
    inteiro cont = 0  
  
    enquanto (cont < 10) {  
        escreva("\n Digite a nota: ")  
        leia (nota)  
        soma = soma + nota  
        cont++  
    }  
  
    media = soma/10  
    escreva("\n Media da sala = ", media)  
}
```

### Vetores Unidimensionais

```
import java.util.*;

public class Media00 {
    public static void main(String[] args) {
        double media, soma = 0, nota;
        int cont = 0;
        Scanner teclado = new Scanner(System.in);

        while (cont < 10) {
            System.out.println("Digite a nota : ");
            nota = teclado.nextDouble();
            soma = soma + nota;
            cont++;
        }
        media = soma / 10;
        System.out.println("Media da Sala = " + media);
    } }
```

## Problema 2

- Calcular a média de uma classe a partir da nota de seus 10 alunos **e verificar quantos conseguiram nota acima da média da classe:**

■ **SAÍDA:**

## Problema 2

- Calcular a média de uma classe a partir da nota de seus 10 alunos **e verificar quantos conseguiram nota acima da média da classe:**

■ **SAÍDA:** média, quantidade acima media (cont)

■ **ENTRADA:**

## Problema 2

- Calcular a média de uma classe a partir da nota de seus 10 alunos **e verificar quantos conseguiram nota acima da média da classe:**

■ **SAÍDA:** média, quantidade acima media (cont)

■ **ENTRADA:** nota

10 vezes

■ **PROCESSAMENTO:**

## Problema 2

- Calcular a média de uma classe a partir da nota de seus 10 alunos **e verificar quantos conseguiram nota acima da média da classe:**

■ **SAÍDA:** média, quantidade acima media (cont)

■ **ENTRADA:** nota

10 vezes

■ **PROCESSAMENTO:**

- Acumular a nota (somar com as anteriores)
- Dividir a soma por 10

10 vezes

■ **E O QUE MAIS ?**

## Problema 2

- Calcular a média de uma classe a partir da nota de seus 10 alunos **e verificar quantos conseguiram nota acima da média da classe:**

■ **SAÍDA:** média, quantidade acima media (cont)

■ **ENTRADA:** nota

10 vezes

■ **PROCESSAMENTO:**

- Acumular a nota (somar com as anteriores)
- Dividir a soma por 10
- **Para cada uma das 10 notas:**

10 vezes

SE nota > media  
ENTÃO contar

## Problema 2

- Calcular a média de uma classe a partir da nota de seus 10 alunos **e verificar quantos conseguiram nota acima da média da classe:**

■ **SAÍDA:** média, quantidade acima media (cont)

■ **ENTRADA:** nota

10 vezes

■ **PROCESSAMENTO:**

- Acumular a nota (somar com as anteriores)
- $MEDIA = soma / 10$
- **Para cada uma das 10 notas:**

10 vezes

SE nota > **MEDIA**  
ENTÃO contar

## Problema 2

- Calcular a média de uma classe a partir da nota de seus 10 alunos **e verificar quantos conseguiram nota acima da média da classe:**

■ **SAÍDA:** média, quantidade acima media (cont)

■ **ENTRADA:** nota } 10 vezes

■ **PROCESSAMENTO:**

- Acumular a nota (somar com as anteriores)
- $MEDIA = soma / 10$
- **Para cada uma das 10 notas:**

SE nota > MEDIA  
ENTÃO contar

10 vezes

Como fazer isso se a MÉDIA só existirá quando a décima nota for lida?

## Problema 2

Para cada um dos 10 alunos:

Ler a nota N

soma = soma + N

//Acumular a nota (somar com as anteriores)

▪ MEDIA = soma / 10

Para cada uma das 10 notas:

SE nota > media  
ENTÃO contar

**ATENÇÃO:** Note que é preciso **armazenar os 10 valores**, para que depois de calculada a média se possa verificar se cada uma das 10 notas estão acima da média

N1	N2	N3	N4	N5	N6	N7	N8	N9	N10
5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0

SOMA
61.0

MEDIA
6.1

## Problema 2

Calcular a média de uma classe a partir da nota de seus 10 alunos e **verificar quantos conseguiram nota acima da média da classe:**

N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	SOMA	MEDIA
5,5	6,5	8,0	3,0	7,5	2,5	7,5	6,0	4,5	10,0	61,0	6,1

- **SE** N1 > media ENTÃO cont = cont + 1
- **SE** N2 > media ENTÃO cont = cont + 1
- **SE** N3 > media ENTÃO cont = cont + 1
- ...
- **SE** N9 > media ENTÃO cont = cont + 1
- **SE** N10 > media ENTÃO cont = cont + 1

**Acima da Média**  
N2, N3, N5, N7, N10

**5 notas acima da média**

## Solução com variáveis simples

```
import java.io.*;

public class Media01 {
    public static void main(String[ ] args) {
        float media, soma = 0, n1, n2, n3, n4, n5, n5, n6, n7,
n8, n9, n10;
        int cont = 0;
        Scanner teclado = new Scanner(System.in);

        System.out.println("Digite a nota  n1: ");
        n1 = teclado.nextFloat();          ...

        System.out.println("Digite a nota n2: ");
        n2 = teclado.nextFloat();          ...

        System.out.println("Digite a nota  n3: ");
        n3 = teclado.nextFloat();          ...

        ...
    }
}
```

## Solução com variáveis simples

```
...  
System.out.println("Digite a nota n10");  
n10 = teclado.nextFloat(); ...
```

```
soma = n1 + n2 + n3 + n4 + n5 + n6 + n7 + n8 + n9 + n10;  
media = soma / 10;
```

```
if (n1 > media) cont = cont + 1;  
if (n2 > media) cont = cont + 1;  
if (n3 > media) cont = cont + 1;  
...  
...
```

```
if (n9 > media) cont = cont + 1;  
if (n10 > media) cont = cont + 1;
```

```
System.out.println("Existem " + cont + " notas cima da media " + media);
```

```
}
```

```
}
```



## Problema

**E se fosse uma turma de 500 alunos ?**  
**500 variáveis ?**  
**500 “if<sub>s</sub>” ?**

**A maioria das linguagens  
implementam variáveis do tipo  
CONJUNTO, chamadas de VETOR  
ou ARRAY**



Universidade Federal do ABC

# Bacharelado em Ciência e Tecnologia Processamento da Informação

## Vetores Unidimensionais

# **VETOR** **(ARRAY)**

## VETOR

- Permite a declaração de variáveis do tipo **CONJUNTO** (com várias posições)
- Uma variável do tipo CONJUNTO pode armazenar mais de um valor (um em cada posição).
- No ato da declaração da variável deve-se informar o seu tamanho.

**nota**

conteúdo	5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0
índice	0	1	2	3	4	5	6	7	8	9

## VETOR

- Para o problema de armazenar **10 notas**, pode-se definir uma variável de **tamanho 10** do tipo real.

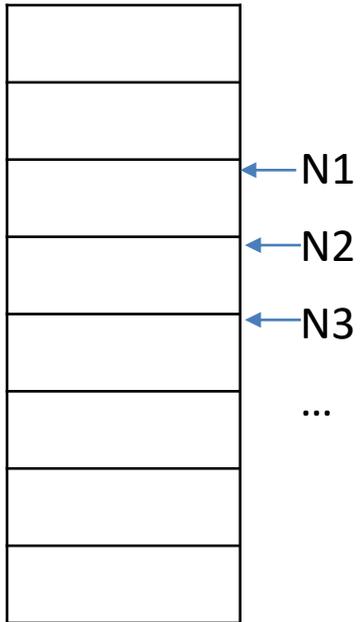
	nota									
conteúdo	5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0
índice	0	1	2	3	4	5	6	7	8	9

Note que isso é diferente de declarar 10 variáveis do tipo real

N1	N2	N3	N4	N5	N6	N7	N8	N9	N10
5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0

### Vetores Unidimensionais

Memória

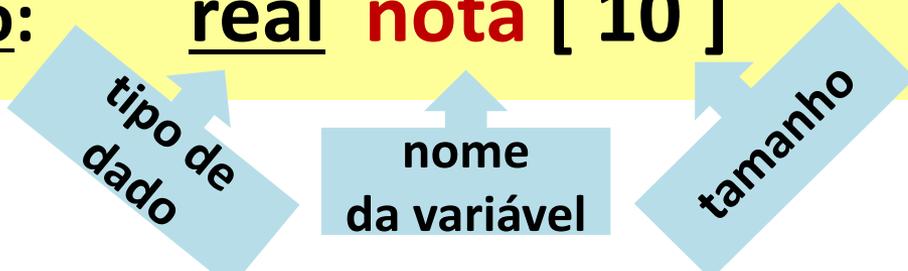


Posição de memória  
Variável **escalar**

## VETOR – No Portugol Studio

Para o problema de armazenar **10 notas**, pode-se definir uma variável de **tamanho 10** do tipo real.

Declaração:      real **nota** [ 10 ]



A declaração acima cria em memória uma variável do tipo REAL, chamada NOTA e tamanho 10. As 10 posições são numeradas de 0 a 9 (índice)

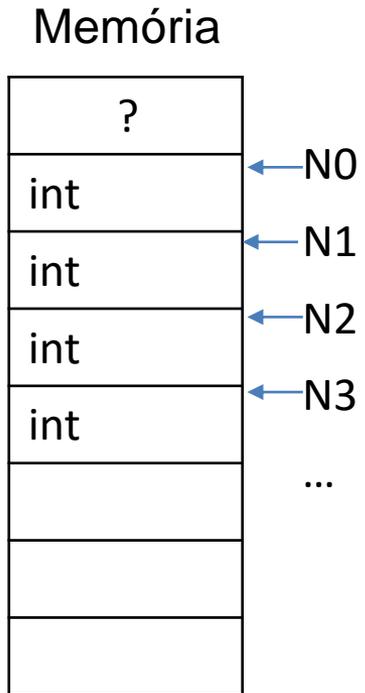
**nota**

conteúdo

5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0
0	1	2	3	4	5	6	7	8	9

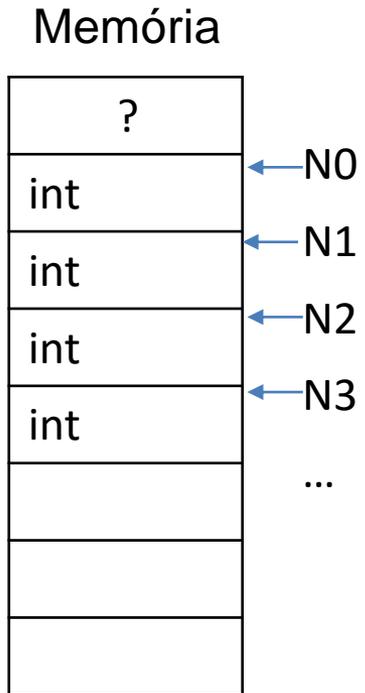
índice

### Vetores Unidimensionais

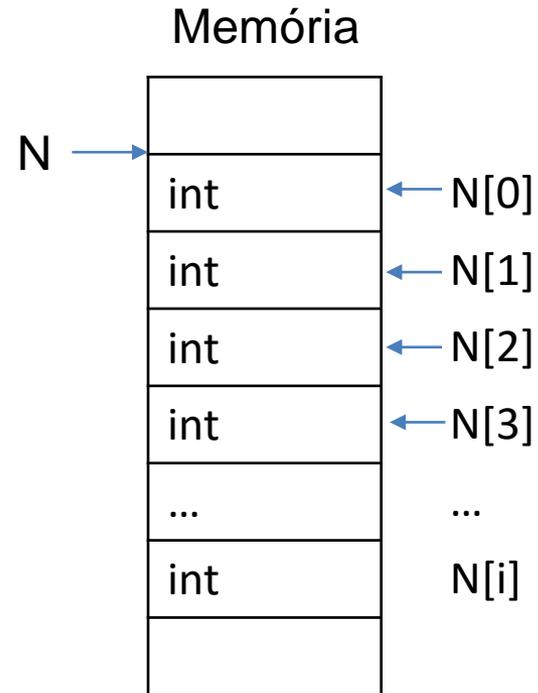


Posição de memória  
Variável **escalar**

### Vetores Unidimensionais



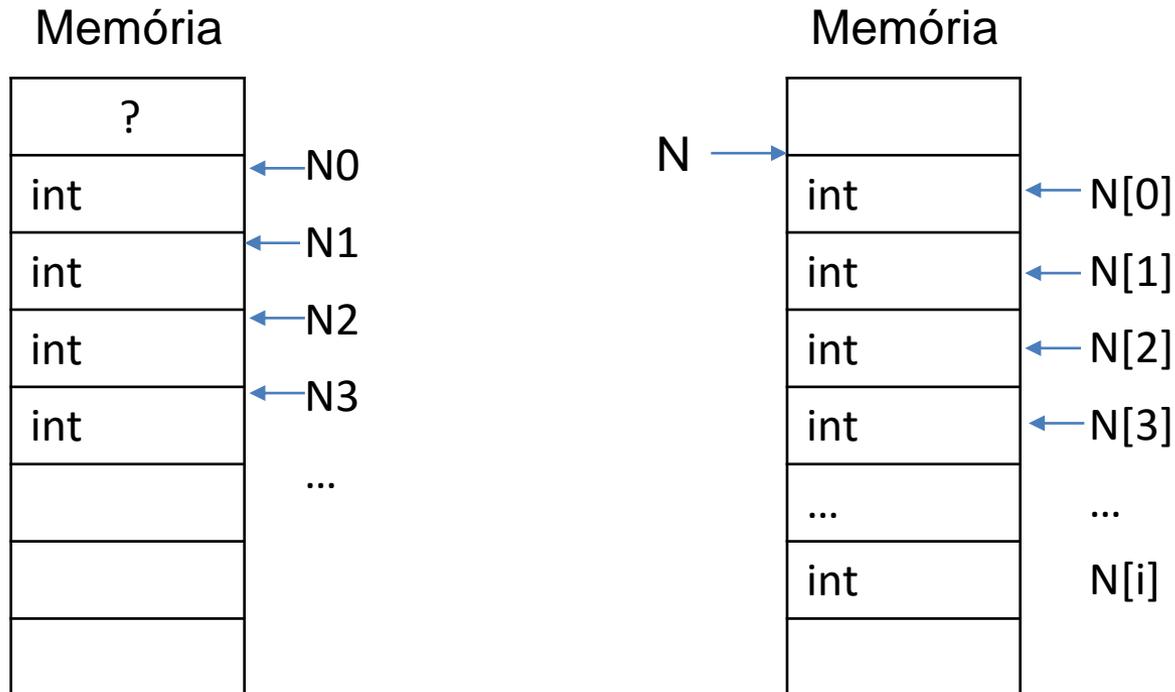
Posição de memória  
Variável **escalar**



Posição de memória  
Vetor de variáveis **escalares**

$$\text{Posição de memória} = \text{pos}(N) + \text{bytes}(\text{tipo}) * i$$

### Vetores Unidimensionais



Posição de memória =  $\text{pos}(N) + \text{bytes}(\text{tipo}) * i$

**EXEMPLO:**  $\text{bytes}(\text{int}) = 4$  (32 bits)

$\text{Posição\_dado}(N[4]) = \#22 + 4 * 4 = \#32$

## VETOR – No Portugol Studio

Declaração:      **real** **nota** [ 10 ]

tipo de  
dado

nome  
da variável

tamanho

As 10 posições são numeradas de 0 a 9 (índice)

**nota**

conteúdo

5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0
0	1	2	3	4	5	6	7	8	9

índice

## VETOR – No Portugol Studio

Declaração:      real **nota** [ 10 ]

tipo de  
dado

nome  
da variável

tamanho

nota

conteúdo	5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0
índice	0	1	2	3	4	5	6	7	8	9

### Acesso a variável:

Para acessar cada posição deve-se usar o nome da variável e a sua posição (**ou índice**) entre colchetes. A instrução abaixo imprime a terceira nota (índice = 2) da variável.

escreva(“ A terceira nota = ” , **nota** [ 2 ] )

O índice pode ser uma  
variável: **Ex.** **nota** [ x ]



Universidade Federal do ABC

Bacharelado em Ciência e Tecnologia  
Processamento da Informação

Vetores Unidimensionais

# SOLUÇÃO EM PORTUGOL

**COM VETOR**

### Vetores Unidimensionais

```
funcao inicio()  
{  
    real nota[10], soma = 0, media  
    inteiro cont = 0  
    inteiro indice = 0  
  
    enquanto (indice < 10) {  
        escreva("\n Digite a nota: ")  
        leia (nota [indice] )  
  
        soma = soma + nota [indice]  
        indice++  
    }  
  
    media = soma/10  
    escreva("\n Media da sala = ", media)
```

**Lê as notas via teclado e guarda no vetor**



?	?	?	?	?	?	?	?	?	?
0	1	2	3	4	5	6	7	8	9

## Vetor no Portugol Studio

```
// verificar quantas notas que estão acima da media
```

```
indice = 0
```

```
enquanto (indice < 10) {
```

```
  se (nota[indice] > media) {  
    cont = cont + 1
```

```
  }
```

```
  indice++
```

```
}
```

```
escreva("\n A quantidade de notas acima da média são = ", cont)
```

Verifica se cada nota armazenada está acima da média

	nota									
conteúdo	5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0
índice	0	1	2	3	4	5	6	7	8	9

media

**6.1**



Universidade Federal do ABC

Bacharelado em Ciência e Tecnologia  
Processamento da Informação

Vetores Unidimensionais

# **VETOR**

## **EM JAVA**

## VETOR – No Portugol Studio

Para o problema de armazenar **10 notas**...

Declaração:      double   **nota**[ ] = **new** double [ 10 ]

tipo de  
dado

nome  
da variável

tamanho

A declaração acima cria em memória uma variável do tipo **double**,  
chamada **NOTA** e tamanho **10**. As 10 posições são numeradas de 0 a 9 (índice)

**nota**

conteúdo

5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0
0	1	2	3	4	5	6	7	8	9

índice

## Sintaxe: regra geral

```
tipo nome[] = new tipo[quantidade];
```

Indicador de múltiplos valores

Vetor

Quantos valores do mesmo tipo

```
nome[índice] = valor;
```

Posição no vetor

## ■ Índice

- Pode ser um número inteiro ou uma variável inteira
- Começa a contagem sempre no 0
- Última posição = quantidade-1
- Permite o uso de laços para variar o índice  
Indica endereço de memória dentro do vetor
  - Ex: Tipo int tem 32 bits ou 4 bytes  
Dado: **int A[] = new int[5];**  
Posição de **A** na memória = **&A;**  
**&A[3] = &A + 3 \* [tamanho de int]**
- No vetor, a manipulação é feita unicamente pelas variáveis internas do vetor, no caso A[posição].

### Vetores Unidimensionais

#### ■ Exemplos:

```
int Z[] = new int[10]; // Aloca 10 valores inteiros para 'Z'
Z[0] = -10; // certo → posição 0 da memória de Z recebe -10
Z[7] = 21; // certo → posição 21 da memória de Z recebe 21
Z[10] = 0; // errado! → não há posição 10
Z=0; // errado! → Z não é um número, é um vetor
int i = 0; // Variável índice
Z[i++] = 11; // certo → posição 0 da memória de Z recebe 11
Z[i++] = 12; // certo → posição 1 da memória de Z recebe 12
Z[i] = 15; // certo → posição 3 da memória de Z recebe 15
Z[i] = 16; // certo → posição 3 da memória de Z recebe 16
```

## Vetores

- Um vetor é representado por:
  - **Identificador** - nome do vetor
  - **Tamanho (dimensão)** - quantidade de elementos (dados)
  - **Tipo** - tipo de dado armazenado (integer, real, string)
- Sintaxe:

```
tipo identificador [ ] = new tipo[tamanho];
```

```
double nota [ ] = new double [10];
```



## Solução com VETOR

```
import java.io.*;  
public class Media02 {  
    public static void main(String[ ] args) {  
        double nota [ ] = new double [10];  
        double media, soma = 0;  
        int indice = 0, cont = 0;
```

```
        Scanner teclado = new Scanner(System.in);
```

```
        while (indice < 10) {
```

```
            System.out.println("Digite a nota :");
```

```
            nota [ indice ] = teclado.nextDouble();
```

```
            soma = soma + nota [ indice ];
```

```
            indice++;
```

```
        }
```

```
        media = soma / 10;
```



	NOTA									
CONTEÚDO	5,5	6,5	8,0	3,0	7,5	2,5	7,5	6,0	4,5	10,0
ÍNDICE	0	1	2	3	4	5	6	7	8	9

ind  
0

SOMA  
61,0

CONTINUA →

### Vetores Unidimensionais



## Solução com VETOR

	NOTA									
CONTEÚDO	5,5	6,5	8,0	3,0	7,5	2,5	7,5	6,0	4,5	10,0
ÍNDICE	0	1	2	3	4	5	6	7	8	9

*/\* ----- verificar quem está acima da média ----- \*/*

`indice = 0;`      */\* recomeça a percorrer a partir da posição ZERO !!! \*/*

```
while ( indice < 10) {
    if ( nota [ indice ] > media ) {
        cont++;
    }
    indice++;
}
```

**ind**  
**0**

**SOMA**  
**61,0**

**MEDIA**  
**6,1**

**contA**  
**5**

```
System.out.println("Existem " + cont + " notas cima da media " + media);
```

```
}
}
```



# Respondendo à pergunta:

**E se fosse uma  
turma de 500  
alunos ?**

## Solução com VETOR

```
import java.io.*;
public class Media02 {
    public static void main(String[ ] args) {

        double nota [ ] = new double [500];
        double media, soma = 0;
        int indice = 0, cont = 0;

        Scanner teclado = new Scanner(System.in);

        while (indice < 500) {

            System.out.println("Digite a nota :");

            nota [ indice ] = teclado.nextDouble ();

            soma = soma + nota [ indice ];
            indice++;

        }

        media = soma / 500 ;
    }
}
```



## Solução com VETOR

```
/* ----- verificar quem está acima da média -----*/  
  
indice = 0; // começa a partir da posição ZERO  
while ( indice < 500) {  
    if ( nota [ indice ] > media ) {  
        cont++;  
    }  
    ind++;  
}  
  
System.out.println("Existem " +  
                    cont + " notas cima da media " + media);  
  
}
```



Universidade Federal do ABC

Bacharelado em Ciência e Tecnologia  
Processamento da Informação

Vetores Unidimensionais

# Conceituando Vetores

## Vetores

- Definição de vetores:
  - “Coleção de variáveis de mesmo tipo”.
  - “Estrutura de dados **homogênea**”.
- Porque **homogênea**?
  - Agrupam várias informações
    - Dentro de uma **mesma** variável e
    - Com o **mesmo** tipo de dado

## Vetores

- Ex: armazenar as médias de 5 alunos

posição	Medias
0	4.5
1	6.5
2	8.0
3	3.5
4	6.0

**Mesmo tipo  
de dado  
tipo REAL**

- ✓ E onde ficam armazenados os dados?

R: De forma agrupada na **Memória Principal** !

## Vetores

- Um vetor é representado por:
  - **Identificador** - nome do vetor
  - **Tamanho (dimensão)** - quantidade de elementos (dados)
  - **Tipo** - tipo de dado armazenado (inteiro, real, cadeia)
- Sintaxe:

**tipo identificador [tamanho]**

**inteiro idade[50]**

## Manipulação de Vetores

- **Cada elemento** de um vetor é tratado como se fosse uma variável simples.
- Para referência a um elemento do vetor utiliza-se o nome do vetor e o seu **índice** entre colchetes.
- O **índice** indica a **posição do dado (conteúdo)** no vetor

**nota**

conteúdo	5.5	6.5	8.0	3.0	7.5	2.5	7.5	6.0	4.5	10.0
índice	0	1	2	3	4	5	6	7	8	9

## Manipulação de Vetores

**Vetor media**

4.5	6.5	8.0	3.5	6.0
0	1	2	3	4

*dados*

*índice*

media [0] = 4.5

media [1] = 6.5

media[2] = 8.0

media[3] = 3.5

media [4] = 6.0

*Atribuição de valores em um vetor*

*índice* *dados*

## OUTROS TIPOS DE DADOS

### ■ Vetor de Character

```
character conceitos [50]
```

```
// declara um vetor para armazenar o conceito de 50 alunos
```

### ■ Vetor de Cadeia (string)

```
cadeia nomes [100]
```

```
// declara um vetor para armazenar o nome de 100 pessoas
```



Universidade Federal do ABC

# Bacharelado em Ciência e Tecnologia Processamento da Informação

## Vetores Unidimensionais

# Alerta !!!

### Vetores Unidimensionais

```
int vet A = new int[5];  
int vet B = new int[5];
```

- **Jamais** fazer:  $A = B$ .  
A atribuição de dois vetores tem que ser feito elemento por elemento.

**Ex:  $A[1] = B[1]$**

- **Jamais** fazer:  $A = \text{teclado.nextInt}()$ ;  
Tem que ler elemento por elemento.  
**Ex:  $A[0] = \text{teclado.nextInt}()$ ;**

- **Jamais** fazer:  $\text{System.out.println}(A)$ ;  
Tem que escrever elemento por elemento.  
**Ex:  $\text{System.out.println}(A[1])$**

### Vetores Unidimensionais

```
int[] A = {1, 2, 3, 4, 5};
```

#### – Errado:

```
System.out.println(A);
```

Não exibirá os valores, mas o endereço de memória base

```
[Ljava.lang.String  
;@28d93b30
```

#### – Correto:

```
for (int i=0; i<5; i++)
```

```
    System.out.println("A[" + i + "] = " + A[i]);
```

```
A[0] = 1  
A[1] = 2  
A[2] = 3  
A[3] = 4  
A[4] = 5
```

### Vetores Unidimensionais

- Cada elemento de um vetor é uma variável simples.
- Referência a um elemento:  
**nome\_do\_vetor [índice]**
- Exemplo 1:
  - No vetor de 80 notas para atribuir a **nota 10** na **posição 80** a instrução é:
  - **notas[ 79] = 10**

0	1	2	3	4	5	6	7	8	...	79
5	7	10	7	8	6	10	3	4		<b>10</b>

## Exemplo 2

- Faça um programa que crie um vetor de string com 5 elementos e preencha cada elemento com uma cor de acordo como desenho abaixo. O programa deve pedir um número de 0 a 4 para o usuário, e imprimir a cor correspondente a esse número.

0	1	2	3	4
Azul	Rosa	Verde	Roxo	Preto

### Vetores Unidimensionais

#### Exemplo 2

0	1	2	3	4
Azul	Rosa	Verde	Roxo	Preto

```
Scanner teclado = new Scanner(System.in);
String Cor[] = new String[5];
Cor[0] = "Azul";
Cor[1] = "Rosa";
Cor[2] = "Verde";
Cor[3] = "Roxo";
Cor[4] = "Preto";
System.out.print("Digite um número de 0 a 4:");

int num = teclado.nextInt();
System.out.println("A cor correspondente é "+ Cor[num]);
```

## Vetores Unidimensionais

Alternativamente

0	1	2	3	4
Azul	Rosa	Verde	Roxo	Preto

```
Scanner teclado = new Scanner(System.in);  
String Cor[]={"Azul", "Rosa", "Verde", "Roxo", "Preto"};  
System.out.print("Digite um número de 0 a 4: ");  
System.out.println("A cor correspondente é "+  
Cor[teclado.nextInt()]);
```



Universidade Federal do ABC

# Bacharelado em Ciência e Tecnologia Processamento da Informação

## Vetores Unidimensionais

# Exercícios

## Exercícios

1. Escreva um algoritmo que Leia 10 valores inteiros e armazene-os em um Vetor. Após a digitação do último valor, imprima os elementos pares.
2. Escreva um algoritmo que Leia 15 nomes de pessoas e armazene-os em um Vetor. Após a digitação do último nome, imprima-os na sequência contrária.
3. Escreva um algoritmo que leia 8 elementos em um vetor inteiro A. Construa um outro vetor B, de mesma dimensão de A, com seus elementos sendo a multiplicação do elemento correspondente de A por 3. Mostre os elementos de B.

No Java → `Tipo Nome_Vetor [ ] = new tipo [ 50 ];`

## Exercícios

- 4 . Escreva um algoritmo que Leia um vetor de N elementos inteiros  $>0$ . Após a digitação do último valor (0), encontre e mostre o maior elemento.
5. Leia 20 elementos em um vetor A e construa o vetor B com elementos de A, porém invertidos, ou seja, o primeiro elemento de A passa a ser o último elemento de B, e assim por diante. Mostre os 2 vetores.
6. Leia um vetor de 9 elementos. Imprima os elementos do vetor identificando como pares ou ímpares.

## Problema

Escreva um programa para preencher um vetor de inteiro com 20 termos de uma PA cujo termo inicial e a razão deverá ser informada pelo usuário. Somente após o preenchimento completo do vetor, o mesmo deverá ser exibido na tela

- **SAÍDA:** Quais os resultados que o algoritmo dever fornecer ?
- **ENTRADA:** Quais os insumos necessários para se obter a saída?
- **PROCESSAMENTO:** Como transformar os insumos na saída?

## Atividade

- Construir um algoritmo em Java que leia em um vetor até 50 modelos de carros (ex: Golf 1.6, Vectra CD 2.0, etc.) e em outro vetor o consumo desses carros em km/litro (quantos quilômetros faz com 1 litro de combustível). Calcule e mostre:
  - Uma lista dos carros por ordem de consumo, começando pelo mais econômico.
  - Alinhar as colunas usando printf() de forma que exiba nas colunas 1. o nome do carro; 2. O consumo; 3. Quantos litros de combustível cada carro consome para percorrer uma distância de 1000 quilômetros.

**Dica: Utilizar um algoritmo de ordenação de vetores ou um vetor de índices.**