

# INTRODUÇÃO À PROGRAMAÇÃO

Prof.<sup>a</sup> Priscilla Abreu

[priscilla.braz@rj.senac.br](mailto:priscilla.braz@rj.senac.br)



# Introdução à Programação



## Roteiro de Aula

- Objetivo da aula
- Vetores unidimensionais

# Introdução à Programação



## Objetivo da aula

Compreender o funcionamento e manipulação de vetores.

# Introdução à Programação



## INTRODUÇÃO

Tipos de dados compõem uma área essencial no contexto de algoritmos e também de estrutura de dados.

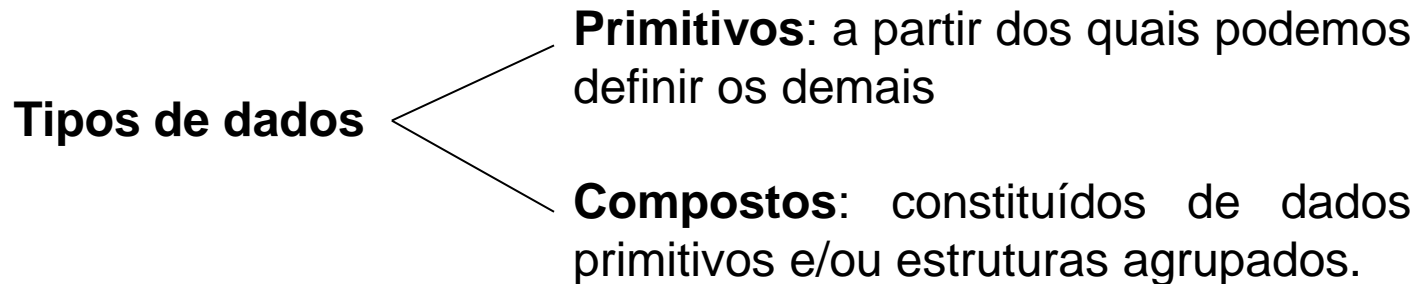
**QUAIS TIPOS DE DADOS VOCÊS JÁ  
CONHECEM?**


**COMO ESSES TIPOS DE DADOS PODEM SER  
DEFINIDOS E UTILIZADOS?**

# Introdução à Programação



## INTRODUÇÃO



- Tipos primitivos 
  - inteiro, real, lógico (boolean), caracter
- Tipos compostos
  - Conjunto de informações agrupadas de uma forma coerente (com alguma relação entre elas)
  - Ex.: lista de chamada da turma.

Para cada variável de tipo primitivo, armazenamos um valor por vez!!!

# Introdução à Programação



## INTRODUÇÃO

- Exemplos:
  - Tipos primitivos:
    - `int idade;`
    - `float altura;`
  - Tipos compostos:
    - `float notas[50];`
    - `aluno alunos[50];`



ALUNO
Nome
Matricula
Nota1
Nota2
Endereço

# Introdução à Programação



## PARA PENSAR...

Considere a seguinte situação:

Suponha que você precise fazer um programa para cadastrar as notas de 20 alunos de uma determinada turma, calcular a média da turma e contabilizar quantos alunos tiveram nota acima da média da turma. Como você resolveria essa situação?”

# Introdução à Programação



**PENSANDO...**

Quantidade de notas a serem lidas: 20

**Preciso de 20 variáveis?**

**float nota1, nota2, ..., nota20 ????**

**Uma só variável para leitura de notas?**

**float nota**

**Outra forma?**



PENSANDO...

## Uso de vetores!

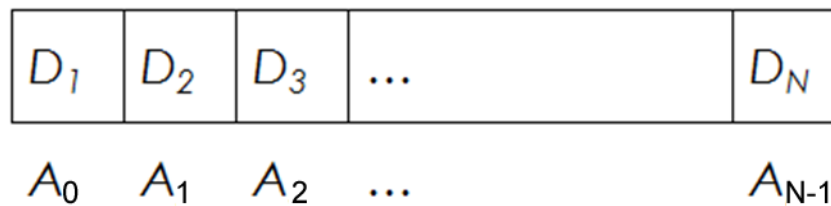
## O que são vetores???

# Introdução à Programação



## VETOR

- É uma coleção de variáveis do mesmo tipo, referenciada por um nome comum;
- Um elemento específico é acessado através de um índice;
- São também denominados de tipos estruturados homogêneos unidimensionais.



# Introdução à Programação



## VETOR

- Declaração de um vetor em C:

```
int nome_vetor[tamanho];
```

Tipo dos dados  
do conjunto

Identificador ou  
nome do conjunto

Dimensão, ou tamanho  
máximo do conjunto

# Introdução à Programação



## VETOR

Quantidade de posições de memória para essa variável

- `float vetNotas[50];`
- Na declaração de vetor, o que está entre colchetes deve ser um número constante .

Assim, não é correto fazer algo como:

- `int n = 20;`
- `float x[n]; /* não é permitido declarar colocando uma variável */`

# Introdução à Programação



## VOLTANDO À SITUAÇÃO...

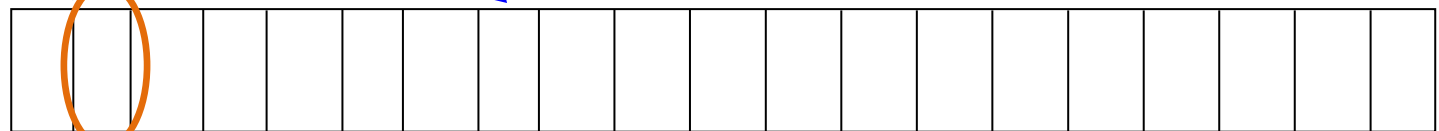
Solução para o problema das múltiplas variáveis de mesmo tipo: um vetor.

Nome: comum a todas as variáveis

Notas

(...)

Nota na  
posição 1:  
estaremos  
acessando a  
2ª nota.



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

Através da posição acessamos cada elemento do vetor.

# Introdução à Programação



## VOLTANDO À SITUAÇÃO...

Solução para o problema das múltiplas variáveis de mesmo tipo: um vetor.



```
float notas[20];
```

# Introdução à Programação



## VOLTANDO À SITUAÇÃO...

Que outras variáveis serão necessárias?

- Variável para percorrer o vetor e acessar cada posição;
- Quantidade de notas acima da média;
- Média.

**int i, qtde;**  
**float media;**

# Introdução à Programação



## SOLUÇÃO PARCIAL...

```
#include <stdio.h>
int main(){
    int i, qtde;
    float notas[20],media;
    media = 0;
    qtde = 0;
    ...
}
```

Teremos uma variável chamada notas, que terá 20 posições na memória. Isto é, 20 notas poderão ser armazenadas.



# **Precisamos fazer a leitura das notas...**

## **Como armazenar em um vetor???**

# Introdução à Programação



## ACESSANDO UM ELEMENTO...

Coloca-se o nome da variável e entre [] coloca-se o índice, que indica a posição do elemento.

O índice é uma constante inteira, uma variável inteira ou um cálculo que resulte em valor inteiro.

Exemplos:

- `notas[1] = 10;`
- `vetor[5] = 30.4;`
- `i = 0;`
- `notas[i] = 7.5;`

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

Cada vetor tem um único nome de variável, o que modifica é apenas a posição de cada elemento no vetor.

## USO DE ESTRUTURA DE REPETIÇÃO!!!

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

9.5	10	8	9.4	3.5	2.9	7	8	6.8	10
notas[0]	notas[1]	notas[2]	notas[3]	notas[4]	notas[5]	notas[6]	notas[7]	notas[8]	notas[9]

**IDENTIFICAÇÃO  
DA VARIÁVEL**

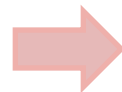


notas



mesmo identificador  
para todos os elementos

**VARIAÇÃO**



índice



As posições seguem a  
sequência de 0...19

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

9.5	10	8	9.4	3.5	2.9	7	8	6.8	10
notas[0]	notas[1]	notas[2]	notas[3]	notas[4]	notas[5]	notas[6]	notas[7]	notas[8]	notas[9]

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...



### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i );  
    scanf("%f", &notas[i]);  
}
```

# Introdução à Programação



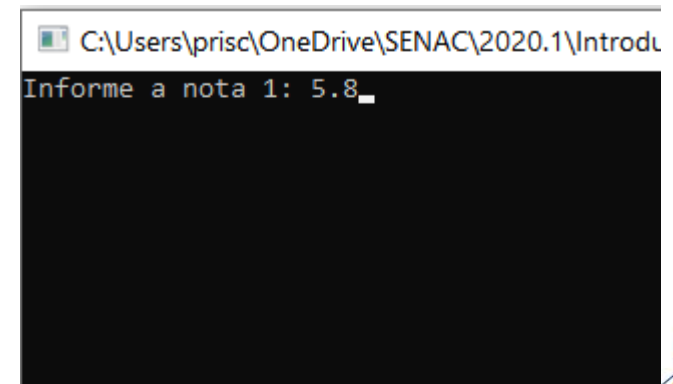
## ACESSANDO VÁRIOS ELEMENTOS...

**i = 0**

5.8									
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```



# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 1**

5.8	9.2								
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1'  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2
```



# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 2**

5.8	9.2	10							
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1\Int  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2  
Informe a nota 3: 10
```

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 3**

5.8	9.2	10	8.5						
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1\Introdução  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2  
Informe a nota 3: 10  
Informe a nota 4: 8.5
```

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 4**

5.8	9.2	10	8.5	7.0					
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1\Introc  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2  
Informe a nota 3: 10  
Informe a nota 4: 8.5  
Informe a nota 5: 7.0
```

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 5**

5.8	9.2	10	8.5	7.0	10				
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1\Introduçã  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2  
Informe a nota 3: 10  
Informe a nota 4: 8.5  
Informe a nota 5: 7.0  
Informe a nota 6: 10_
```

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 6**

5.8	9.2	10	8.5	7.0	10	7.7			
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1\Introdu  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2  
Informe a nota 3: 10  
Informe a nota 4: 8.5  
Informe a nota 5: 7.0  
Informe a nota 6: 10  
Informe a nota 7: 7.7_
```

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 7**

5.8	9.2	10	8.5	7.0	10	7.7	8.0		
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1\Introd  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2  
Informe a nota 3: 10  
Informe a nota 4: 8.5  
Informe a nota 5: 7.0  
Informe a nota 6: 10  
Informe a nota 7: 7.7  
Informe a nota 8: 8.0_
```

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 8**

5.8	9.2	10	8.5	7.0	10	7.7	8.0	4.6	
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1\Introd  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2  
Informe a nota 3: 10  
Informe a nota 4: 8.5  
Informe a nota 5: 7.0  
Informe a nota 6: 10  
Informe a nota 7: 7.7  
Informe a nota 8: 8.0  
Informe a nota 9: 4.6_
```

# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 9**

5.8	9.2	10	8.5	7.0	10	7.7	8.0	4.6	9.0
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1\I  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2  
Informe a nota 3: 10  
Informe a nota 4: 8.5  
Informe a nota 5: 7.0  
Informe a nota 6: 10  
Informe a nota 7: 7.7  
Informe a nota 8: 8.0  
Informe a nota 9: 4.6  
Informe a nota 10: 9.0_
```



# Introdução à Programação



## ACESSANDO VÁRIOS ELEMENTOS...

**i = 10**

**Fim da  
repetição!!!**

5.8	9.2	10	8.5	7.0	10	7.7	8.0	4.6	9.0
0	1	2	3	4	5	6	7	8	9

### VETOR NOTAS

```
for (i=0; i<=9; i++){  
    printf("Informe a nota %d:", i+1 );  
    scanf("%f", &notas[i]);  
}
```

```
C:\Users\prisc\OneDrive\SENAC\2020.1\Introdução à Programação  
Informe a nota 1: 5.8  
Informe a nota 2: 9.2  
Informe a nota 3: 10  
Informe a nota 4: 8.5  
Informe a nota 5: 7.0  
Informe a nota 6: 10  
Informe a nota 7: 7.7  
Informe a nota 8: 8.0  
Informe a nota 9: 4.6  
Informe a nota 10: 9.0  
  
-----  
Process exited after 1284 seconds with return value 0  
Pressione qualquer tecla para continuar. . .
```

# Introdução à Programação



## SOLUÇÃO...

```
#include <stdio.h>
int main(){
    int i, qtde;
    float notas[20],media;
    media = 0;
    qtde = 0;
    ...
```

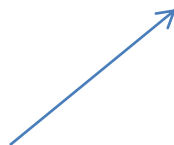
# Introdução à Programação




## SOLUÇÃO...

```
for(i=0;i<20;i++){  
    printf("Informe uma nota: ");  
    scanf("%f",&notas[i]);  
    media = media + notas[i];  
}  
media = media /20;
```

Faz a leitura de cada nota e soma na variável media.



Após a leitura de todas as notas, calcula a media.



# Introdução à Programação



## SOLUÇÃO...

```
for(i=0;i<20;i++){  
    if(notas[i]>media)  
        qtde++;  
}  
printf("Média: %.1f\n",media);  
printf("Notas acima da média: %d",qtde);  
}
```

Faz a contagem de quantas notas ficaram acima da média.

# Introdução à Programação



## EXEMPLO

Faça um programa que leia 20 números inteiros e após a leitura imprima os números informados.

# Introdução à Programação



## EXEMPLO

```
#include <stdio.h>
#define max 5
int main(){
    int valores[max];
    int i;
    for (i=0;i<max;i++){
        printf("Digite um número:");
        scanf("%d",&valores[i]);
    }
    for (i=0;i<max;i++){
        printf("Valor[%d]: %d \n",i,valores[i]);
    }
}
```

# Introdução à Programação



## EXEMPLO

Elabore um programa em C que leia 2 vetores com 5 valores inteiros cada e determine o vetor soma.

<b>v1</b>	2	5	3	10	15
-----------	---	---	---	----	----

<b>v2</b>	8	3	7	4	3
-----------	---	---	---	---	---

<b>vSoma</b>	10	8	10	14	18
--------------	----	---	----	----	----

# Introdução à Programação



## EXEMPLO

```
#include <stdio.h>
#define max 5
int main(){
    int v1[max], v2[max],vSoma[max];
    int i;
    printf("\nPrimeiro vetor\n");
    for (i=0;i<max;i++){
        printf("Digite um número:");
        scanf("%d",&v1[i]);
    }
```



# Introdução à Programação



## EXEMPLO

```
printf("\nSegundo vetor\n");
for (i=0;i<max;i++){
    printf("Digite um número:");
    scanf("%d",&v2[i]);
}
//Vetor Soma
for(i=0; i<max; i++){
    vSoma[i]= v1[i] + v2[i];
    printf("%d\n",vSoma[i]);
}
}
```

# Introdução à Programação



## VETORES

Vimos que podemos armazenar um conjunto de valores na memória do computador através do uso de vetores (arrays).

O vetor é uma forma simples de organizarmos dados na memória do computador.

Com vetores, os valores são armazenados na memória do computador em sequência, um após o outro, e podemos livremente acessar qualquer valor do conjunto.