

INSTITUTO FEDERAL

Bahia

Campus Santo Antônio de Jesus

<http://www.portal.ifba.edu.br/santoantonio>

LINGUAGEM DE PROGRAMAÇÃO

Prof. George Pacheco Pinto

AGENDA

- ❑ Linguagem C
 - ❑ Vetores e Matrizes

VETORES

- ❑ Um vetor é um conjunto de dados HOMOGÊNEO;
- ❑ Permite trabalhar com o agrupamento de várias informações dentro de uma mesma variável;
- ❑ A quantidade de informações disponíveis para armazenamento é indicada na declaração do vetor.

0	1	2	...	N
elemento1	elemento2	elemento3		elementoN

DECLARANDO VETORES

- ❑ `<tipo> identificador [<numero de posições>];`
- ❑ Tipo
 - ❑ Qualquer um dos tipos primitivos estudados. Ex: int, float, char.
- ❑ Identificador
 - ❑ Nome da variável que será referenciada no programa.
- ❑ Número de posições
 - ❑ Número máximo de posições para armazenamento.

DECLARANDO VETOR

❏ Exemplos:

```
int numeros[1000]; /* conjunto de 1000 números inteiros */
```

```
float notas[65]; /* conjunto de 65 números reais */
```

```
char nome[40]; /* conjunto de 40 caracteres */
```

ATRIBUIÇÃO DE DADOS

❏ Exemplos

```
numeros[0] = 34; /* insere o valor 34 na primeira posição do  
vetor de números. */
```

```
notas[1] = 10.0; /* insere a nota 10.0 na segunda posição do  
vetor de notas*/
```

```
notas[99] = 3.5; /* insere a nota 3.5 na posição 99 do  
vetor*/
```

PERCORRENDO UM VETOR

- ❑ Utilizada estrutura de repetição para percorrer todos os registros de um vetor de forma sequencial.
- ❑ O valor da variável de teste é utilizado como índice de acesso ao vetor.

```
for(i = 0; i < 100; i++){  
    vetor[i];  
}
```

EXEMPLO

```
int main () {  
    float notas[10];  
    int i = 0;  
    for(i = 0; i < 10; i++) {  
        printf("Digite a nota do aluno %d: ", i+1);  
        scanf("%f", &notas[i]);  
    }  
    for(i = 0; i < 10; i++){  
        printf("Nota do aluno %d: %f", i+1, notas[i]);  
    }  
}
```


MATRIZES

❑ Como armazenar as notas de 3 disciplinas de 100 alunos?

```
float portugues[100], matematica[100], geografia[100];
int i;
for(i = 0; i < 100; i++)
{
    printf("Nota de Português do aluno %d: ", i+1);
    scanf("%f",&portugues[i]);
    printf("Nota de Matemática do aluno %d: ", i+1);
    scanf("%f",&matematica[i]);
    printf("Nota de Geografia do aluno %d: ", i+1);
    scanf("%f",&geografia[i]);
}
```

MATRIZES

Como armazenar as notas de 20 disciplinas de 100 alunos?

```
float disci1[100],disci2[100] ....., disci20[100];
for(i = 0; i < 100; i++)
{
    printf("Nota da disciplina1 do aluno %d: ", i+1);
    scanf("%f",&disci1[i]);
    printf("Nota de disciplina2 do aluno %d: ", i+1);
    scanf("%f",& disci2[i]);
    /* .... */
    printf("Nota de disciplina20 do aluno %d: ", i+1);
    scanf("%f",& disci20[i]);
}
```

MATRIZES

Como armazenar as notas de 20 disciplinas de 100 alunos?

```
float disci1[100],disci2[100] ....., disci20[100];
for(i = 0; i < 100; i++)
{
    for(j = 0; j < 20; j++){
        printf("Nota da disciplina%d do aluno %d: ", j + 1,i+1);
        scanf("%f",&disci???[i]);
    }
}
```

*Não é possível selecionar o vetor correto em tempo de execução dentro da estrutura do segundo laço de repetição for.

**Precisamos de um tipo de vetor capaz de armazenar todas as notas de todas as disciplinas.

MATRIZES

- ❑ Como armazenar as notas de 20 disciplinas de 100 alunos?

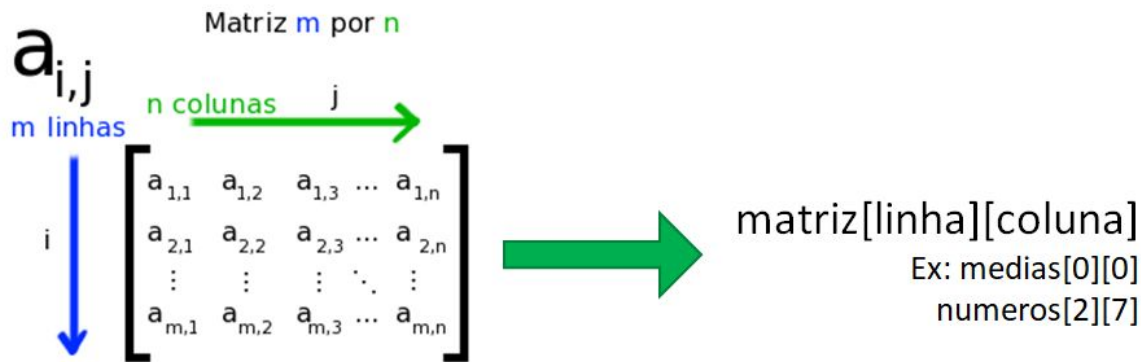
	1	2	3	100
disci1	9.0	7.5	3.5	..	8.5
disci2	3.5	6.0	4.0	9.5	7.0
disci3	2.0	10.0	4.2	7.0	8.0
disci4	4.5	7.0	8.0	10.0	9.0
.....					
disci20	0.0	5.0	6.0	9.0	10.0

MATRIZES

- ❑ Uma matriz é um vetor (ou seja, um conjunto de variáveis de mesmo tipo) que possui duas ou mais dimensões.
- ❑
- ❑ Um vetor de vetores

MATRIZES

- ❑ Em um vetor, para acessar um valor qualquer, é necessário usar um índice para determinar sua posição.
 - ❑ Ex: `numeros[2]`, `idades[15]`.
- ❑ Na matriz são necessários dois índices. O primeiro determina em que linha da matriz se encontra o elemento, enquanto o segundo determina a coluna.



MATRIZES

- Exemplo: matriz de dados 5 X 5
m[5][5]

Colunas					
Linhas	m[0][0]	m[0][1]	m[0][2]	m[0][3]	m[0][4]
	m[1][0]	m[1][1]	m[1][2]	m[1][3]	m[1][4]
	m[2][0]	m[2][1]	m[2][2]	m[2][3]	m[2][4]
	m[3][0]	m[3][1]	m[3][2]	m[3][3]	m[3][4]
	m[4][0]	m[4][1]	m[4][2]	m[4][3]	m[4][4]

DECLARANDO MATRIZES

- ❑ `<tipo> identificador [<numero de linhas>][<numero de colunas>];`
- ❑ Tipo: Qualquer um dos tipos primitivos estudados.
- ❑ Identificador: Nome da variável que será referenciada no programa.
- ❑ Número de linhas: Número máximo de linhas.
- ❑ Número de colunas: Número máximo de colunas.

DECLARANDO MATRIZES

```
int numeros[10][1000];
```

```
float notas[20][100];
```

```
char nomes [10][30];
```

MANIPULANDO MATRIZES

```
int main() {  
    float notas[20][100];  
    //Percorre todas as linhas (0 até 19)  
    for(i = 0; i < 20; i++)  
    {  
        //Percorre todas as colunas (0 até 99)  
        for(j = 0; j < 100; j++){  
            printf("Nota da disciplina %d do aluno %d: ", j+1,i+1);  
            scanf("%f",&notas[i][j]);  
        }  
    }  
}
```

EXERCÍCIOS

1. Construa um programa em C para armazenar as notas de uma turma com 15 alunos utilizando vetores. Imprima a menor nota do vetor.
2. Construa um programa em C que solicite a idade de dez pessoas armazenando os dados em um vetor. Imprimir a idade de todas as pessoas. Imprimir a menor idade digitada.
3. Fazer um programa em C para ler um vetor de inteiros positivos de 50 posições. Imprimir a quantidade de números pares e de múltiplos de 5.

EXERCÍCIOS

4. Construa um programa para armazenar as notas de 5 turmas com 15 alunos cada.

Imprima todas as notas no formato matriz.

```
| X X X X X X |  
| X X X X X X |
```

Imprima a média das notas de cada turma.

Imprima a menor nota de cada turma.

EXERCÍCIOS

5. Criar um programa em C que leia os elementos de uma matriz inteira de 4×4 e imprimir os elementos da diagonal principal.
6. Criar um programa em C que leia os elementos de uma matriz inteira de 3×3 e imprimir outra matriz multiplicando cada elemento da primeira matriz por 2.

REFERÊNCIAS

Consultar ementário.