

# MC102

## Algoritmos e Programação de Computadores

Aula de Laboratório 07

Instituto de Computação  
Primeiro Semestre de 2012

16 de abril de 2012



## Matrizes

- Uma Matriz é um vetor que possui duas ou mais dimensões;

### Declaração

**<tipo>** **<identificador>** [**<linhas>**][**<colunas>**]

**<tipo>** **<identificador>** [**<dim1>**][**<dim2>**] ... [**<dimN>**]

Ex.: **int a** [4][4];

**int b** [4][5][3];

Ex.: **int matriz** [4][4];

|   |   |   |   |   |
|---|---|---|---|---|
|   |   |   |   |   |
|   | 0 | 1 | 2 | 3 |
| 0 |   |   |   |   |
| 1 |   |   |   |   |
| 2 |   |   |   |   |
| 3 |   |   |   |   |

MC102 - Algoritmos e Programação de Computadores



3 / 15

## Conteúdo

1 Matrizes

2 Funções

## Matrizes

### Exemplo

Ex.: **int a** [4][4];

|   |         |         |         |         |
|---|---------|---------|---------|---------|
|   |         |         |         |         |
|   | 0       | 1       | 2       | 3       |
| 0 | a[0][0] | a[0][1] | a[0][2] | a[0][3] |
| 1 | a[1][0] | a[1][1] | a[1][2] | a[1][3] |
| 2 | a[2][0] | a[2][1] | a[2][2] | a[2][3] |
| 3 | a[3][0] | a[3][1] | a[3][2] | a[3][3] |



## Inicializando Matrizes

### Inicializando uma Matriz

```
int mat[2][3] = {{10,20,30},
                {60,70,80}};
```

### Inicializando uma Matriz

```
int mat[4][4] = {{10,20,30,40},
                 {60,70,80,90},
                 {2,4,6,8},
                 {7,5,13,14}};
```

- Faça um programa que imprima a primeira linha da Matriz acima.



## Inicializando Matrizes

### Inicializando uma Matriz

```
int mat[4][4] = {{10,20,30,40},
                 {60,70,80,90},
                 {2,4,6,8},
                 {7,5,13,14}};
```



## Inicializando Matrizes

### Inicializando uma Matriz

```
int mat[4][4] = {{10,20,30,40},
                 {60,70,80,90},
                 {2,4,6,8},
                 {7,5,13,14}};
```

- Faça um programa que imprima a primeira linha da Matriz acima.

### Solução

```
1 #include <stdio.h>
2 int main(){
3     int i, matriz[4][4] = { {10,20,30,40},
4                             {60,70,80,90},
5                             {2,4,6,8},
6                             {7,5,13,14}};
7     for(i = 0; i < 4; i++){
8         printf("%d ", matriz[0][i]);
9     }
10    printf("\n");
11 }
```



## Matrizes

### Problema

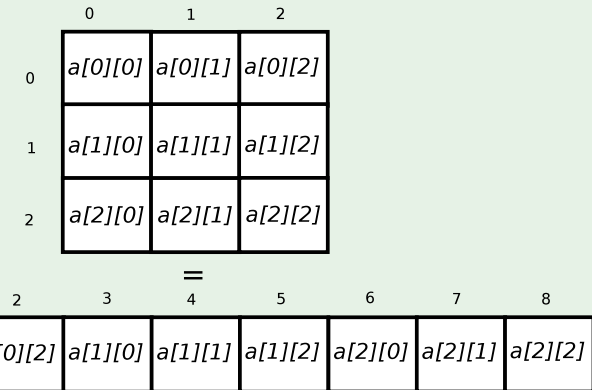
- Multiplique os valores de uma Matriz por um valor do teclado.



## Linearização

- Podemos representar uma Matriz como um Vetor;
- Para isso basta utilizar um padrão para acessar esse vetor como se fosse uma Matriz.

### Representação



## Matrizes

### Problema

- Multiplique os valores de uma Matriz por um valor do teclado.

### Solução

```

1 #include <stdio.h>
2 int main(){
3     int num, i, j, matriz[3][3] = {{10,20,30},
4                                     {15,10,5},
5                                     {4,6,8}};
6     printf("Digite um valor: \n");
7     scanf("%d",&num);
8     for(i = 0; i < 3; i++){
9         for(j = 0; j < 3; j++){
10             matriz[i][j] = num * matriz[i][j];
11             printf("%d ",matriz[i][j]);
12         }
13         printf("\n");
14     }
15     printf("\n");
16 }
```



## Linearização

- `int matriz[12];` //invés de: `int matriz[3][4];`
- Podemos usar: `matriz[i*4+j]`, ou seja, `matriz[i*coluna+j]`;
- Note que `i` pula um **bloco de tamanho 3** (valor de **coluna**) e `j` indexa a **posição dentro do bloco**;

### Exemplo

```

1 #include <stdio.h>
2 int main(){
3     int mat[40]; //representando mat[5][8]
4     int i,j;
5     for(i = 0; i < 5; i++){
6         for(j = 0; j < 8; j++){
7             mat[i*8 + j] = i*j;
8         }
9     }
10    for(i = 0; i < 5; i++){
11        for(j = 0; j < 8; j++){
12            printf("%d, ", mat[i*8 + j]);
13        }
14    }
```



# Funções

- **Funções** são estruturas que agrupam um conjunto de comandos, que são executados quando a função é chamada.

Ex.: `printf(); scanf();`

- Funções podem receber e retornar valores;

Ex.: `x = sqrt(4);`

- Vantagens:

- 1 Evita que os blocos do programa fiquem grandes e difíceis de entender ;
- 2 Separa o programa em partes compreendidos de forma isolada;
- 3 Permite reaproveitamento de código já construído;

## Exemplo

```
1 #include <stdio.h>
2 void imprime(int num){
3     printf("\nNumero: %d\n",num);
4 }
5
6 int main(){
7     int res, a=4, b=-10;
8     res = a + b;
9     imprime(res);
10 }
```



# Funções

## Declarando

```
tipo nome (tipo param1, ..., tipo paramN){
    comandos;
    return valor de retorno;
}
```



# Funções

## Exemplo 2

```
1 #include <stdio.h>
2 int soma(int a, int b){
3     int c;
4     c = a + b;
5     return c;
6 }
7
8 int main(){
9     int res, a=4, b=-10;
10    res = soma(5,6);
11    printf("Primeira soma: %d\n",res);
12    res = soma(a,b);
13    printf("Segunda soma: %d\n",res);
14 }
```



# Funções



## Funções

### Exemplo 3

```

1 #include <stdio.h>
2
3 float soma (float op1, float op2);
4 float subtr (float op1, float op2);
5
6 int main () {
7     float a = 0, b = 5;
8     printf("%f\n %f\n", soma(a, b), subtr(a, b));
9     return 0;
10 }
11
12 float soma (float op1, float op2) {
13     return (op1 + op2);
14 }
15
16 float subtr (float op1, float op2) {
17     return (op1 - op2);

```



MC102 - Algoritmos e Programação de Computadores

13 / 15

## Questões?

Obrigado!

Para informação:

Página dos Laboratórios (Tarefas): <http://susy.ic.unicamp.br:9999/mc102ab>

Página do Curso: <http://www.lrc.ic.unicamp.br/~geraldoms/mc102>

E-mail:

[geraldoms\[at\]lrc.ic.unicamp.br](mailto:geraldoms[at]lrc.ic.unicamp.br)

[brhenrique.fischer\[at\]gmail.com](mailto:brhenrique.fischer[at]gmail.com)



MC102 - Algoritmos e Programação de Computadores

15 / 15

## Funções

### Problema

- Crie uma função que recebe um vetor com 10 notas de alunos e retorne a média;

### Solução

```

1 #include <stdio.h>
2 float media(float vetor[]){
3     float soma=0.0;
4     int i;
5     for(i=0; i<10; i++){
6         soma += vetor[i];
7     }
8     return (soma/10);
9 }
10
11 int main(){
12     float res, vet[10];
13     int i;
14     for (i=0; i<10; i++){
15         printf("\nDigite o %do numero: ",(i+1),&vet[i]);
16         scanf("%f",&vet[i]);
17     }
18     res = media(vet);
19     printf("Media: %.2f\n",res);
20 }

```



MC102 - Algoritmos e Programação de Computadores

14 / 15