

# Linguagem C

## vetores, matrizes e strings

André Tavares da Silva

[andre.silva@udesc.br](mailto:andre.silva@udesc.br)

# Matrizes

- Pode ser definido como sendo um conjunto de variáveis do mesmo tipo. São também conhecidos por vetores ou *array*;
- Sua declaração segue o mesmo formato de variáveis, porém acrescenta-se o número de elementos desejados entre colchetes ([ e ]).

`<tipo> <identificador>[<quantidade>]`

```
int posicao_mouse[2];
```

```
float vet3d[3];
```

```
char nome[60];
```

# Matrizes

- Os elementos são enumerados de 0 (zero) até o número especificado como quantidade menos 1 (n-1);
- Para matrizes multidimensionais, são fornecidos tantos índices quanto o número de dimensões;
- Para referenciar uma das variáveis, devemos fornecer o índice do elemento entre colchetes.
- A linguagem C **não verifica** se o índice é inválido, permitindo a leitura de "lixo" e alteração de outras posições de memória.

```
float valor, matriz[3][3];
```

```
matriz[2][1] = 3.26;
```

```
valor = matriz[2][1];
```

# Inicialização na declaração

- A inicialização pode ser feita apenas na declaração.
- Se o num de elementos é menor que o tamanho do vetor, os elementos restante são inicializados com zero.
- Podemos omitir o tamanho do vetor quando ele é inicializado (mas **somente** quando inicializado!). No caso de matrizes de mais de uma dimensão, somente a primeira dimensão pode ser omitida.

```
int vet[] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};  
float Mat3x3[3][3] = {{ 1, 0, 0 },  
                        { 0, 1, 0 },  
                        { 0, 0, 1 }};  
char nome[60] = "Curso de C";
```

# Exercícios

- Fazer um programa para ler um vetor (matriz unidimensional) de 8 números inteiros. Imprimir na tela os conteúdos do vetor lido.
- Altere o primeiro exercício para também mostrar a soma de seus elementos e apresentar quantos deles são positivos.
- Encontre, a seguir, o maior elemento deste vetor e o escreva.

# Exercícios

- Escrever um algoritmo que leia dois vetores de 5 posições e calcule o produto escalar entre os dois vetores escrevendo a resposta.
- Escreva um programa em C que, dada uma matriz 3x3, armazena em cada posição da matriz, a soma dos valores da linha e coluna que definem a posição. Por exemplo, na posição [1][2] você deverá armazenar o valor  $1+2=3$  e assim por diante. Exiba esta matriz.

# *String* em C

- Chamamos de *string* um vetor de caracteres (também chamado de cadeia de caracteres).
- Como são muito usados, existe um conjunto de funções para manipulação de *strings* em C:
  - `gets(<variável>)` : lê uma “frase” do teclado (a função *scanf* para no primeiro espaço);
  - `strlen(<variável>)` : informa o tamanho de caracteres;
  - `strcmp(<variável 1>, <variável 2>)` : compara o conteúdo de duas variáveis retornando 0 (zero) se o conteúdo é idêntico, +1 se a primeira for maior ou -1 se for menor;
  - `strcpy(<variável 1>, <variável 2>)` : copia o conteúdo da variável 2 para a variável 1.

# Exercício

- Escrever um programa em C que leia uma frase e exiba ela ao contrário (de trás para frente);
- Escreva um programa para informar se uma frase lida pelo usuário é um palíndromo;
- Faça com que o exercício anterior ignore espaços, pontuações (considerar somente letras e números) e maiúscula/minúscula.