



ALGORITMO

Elidiane Martins

elidiane@fgf.edu.br



C – INTRODUÇÃO

Para Não ESQUECER:

- Toda instrução termina com **ponto e virgula** “.”
- Sequencia de instruções deve ser colocada entre chaves “{ }”, conhecida também como BLOCO, pode conter além de instruções, definições de variáveis e tipos;



C – INTRODUÇÃO

Para Não ESQUECER:

- O scanf tem a seguinte estrutura:

`scanf (“%formato”, &EndVariavel)`

- O printf tem a seguinte estrutura:

`printf(“Texto qualquer %formato”, variavel)`



C – INTRODUÇÃO

Controle de fluxo:

- Tomadas de decisão: **if-else**;
- Laços com teste de encerramento no início: **while, for**;
- Laços com teste de encerramento no fim: **do-while**;
- Seleção de um dentre um conjunto de possíveis casos: **switch**.
- Parada: **break, continue**.

C – INTRODUÇÃO

Estruturas homogêneas:

- Estruturas que permitem o armazenamento de valores que são agregados de um tamanho definido;
- Essas estruturas armazenam um só tipo de dado.



C – INTRODUÇÃO

Matrizes unidimensionais

- Também conhecidas como *array* ou vetores. Em sua forma mais simples, é declarada da seguinte forma:
tipo nome-do-array [tamanho];
- Os elementos do vetor são acessados por meio de índice, que indicam a posição do elemento. O elemento inicial possui índice 0.



C – INTRODUÇÃO

Matrizes unidimensionais

- Iniciação do vetor:

```
int vet[5];
```

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

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

```
int vet[10] = {[3]= 4, [7] = 3, [9] = 0, [0]=1};
```

Vale ressaltar que passamos para o scanf o endereço de cada elemento do vetor
&vet[i]

```
1  #include <stdio.h>
2
3  int main(){
4      int vet[3], soma;
5
6      vet[0]=10;
7      vet[1]=15;
8      vet[2]=5;
9
10     soma=v[0]+v[1]+v[2];
11
12     printf("O total da soma dos elementos é: %d\n", soma);
13
14     return 0;
15 }
```

```
[MacBook-Air-de-Elidiane:vetoires elidianemartins$ gcc -c vetorsimples.c
[MacBook-Air-de-Elidiane:vetoires elidianemartins$ gcc -o prog vetorsimples.c
[MacBook-Air-de-Elidiane:vetoires elidianemartins$ ./prog
O total da soma dos elementos é: 30
MacBook-Air-de-Elidiane:vetoires elidianemartins$
```




```
1  #include <stdio.h>
2
3  int main(){
4      int soma;
5
6      int vet[] = {10, 15, 5};
7
8      soma=vet[0]+vet[1]+vet[2];
9
10     printf("O total da soma dos elementos é: %d\n", soma);
11
12     return 0;
13 }
```



```
1  #include <stdio.h>
2
3  int main(){
4      ...
5      int i, vet[5];
6      ...
7      for (i=0; i<5; i++){
8          vet[i]=i+2;
9      }
10     printf("Escolha uma posição do vetor\n");
11     scanf("%d", &i);
12     printf("o %d número armazenado foi %d\n", i, vet[i]);
13
14     return 0;
15 }
```

```
[MacBook-Air-de-Elidiane:vetores elidianemartins$ gcc -c vetor.c
[MacBook-Air-de-Elidiane:vetores elidianemartins$ gcc -o prog vetor.c
[MacBook-Air-de-Elidiane:vetores elidianemartins$ ./prog
Escolha uma posição do vetor
4
o 4 número armazenado foi 6
[MacBook-Air-de-Elidiane:vetores elidianemartins$ ./prog
Escolha uma posição do vetor
5
o 5 número armazenado foi 32767
MacBook-Air-de-Elidiane:vetores elidianemartins$
```



Introdução

EXERCÍCIO DE FIXAÇÃO.



Referências

SILVA, O. Q. Estrutura de Dados e Algoritmos usando C – Fundamentos e Aplicações. Rio de Janeiro: Editora Ciência Moderna Ltda., 2007

OLIVEIRA, U. Programando em C Fundamentos. Volume 1. Rio de Janeiro: Editora Ciência Moderna. 2008.

TENENBAUM, A. M. Estrutura de Dados usando C. Makron books. 1991.