

DISCIPLINA: ESTRUTURAS DE DADOS

PROGRAMA DE ARTICULAÇÃO DA FORMAÇÃO PROFISSIONAL MÉDIA E SUPERIOR (AMS)

- Prof. Alexandre Ponce de Oliveira
alexandreponceo@gmail.com
Aulas 01 e 02
Lins/SP

Ementa

- **Revisão**
 - Vetores, Matrizes, Funções e Structs
- Ponteiros
- Manipulação de Arquivos
- Recursão
- Alocação dinâmica de memória
- Listas, Pilhas e Filas
- Algoritmos de Ordenação
- Árvores Binárias
- **Grafos**

Referências

REFERÊNCIAS BÁSICAS

- PEREIRA, S. L. Estruturas de dados fundamentais – Conceitos e Aplicações. São Paulo: Érica, 2009.
- EDELWEISS, N; GALANTE, R. Estruturas de dados. V 18. Porto Alegre: Bookman, 2009.
- ASCENCIO, A. F. G. Estruturas de dados. São Paulo: Pearson Brasil, 2011.

REFERÊNCIAS COMPLEMENTARES

- ORMEN, T. H. et al. Algoritmos. Rio de Janeiro: Campus, 2012.
- EDELWEISS, N.; GALANTE, R. Estruturas de Dados. Série: Livros Didáticos Informática UFRGS. Volume 18. Porto Alegre: Bookman. 2008.
- GOODRICH, M. T.; TAMASSIA, R. Estruturas de Dados & Algoritmos em Java. 5 ed. Porto Alegre: Bookman. 2013.

Critérios de avaliação

1º Bimestre: P1 + Exercícios com Algoritmos

2º Bimestre: P2 + Exercícios com Algoritmos

3º Bimestre: P3 + Exercícios com Algoritmos

4º Bimestre: P4 + Exercícios com Algoritmos

$$\text{Média Final} = (1^\circ \text{ Bim} + 2^\circ \text{ Bim} + 3^\circ \text{ Bim} + 4^\circ \text{ Bim}) / 4$$

Revisão - Vetores

Vetores

- São matrizes unidimensionais.
- Vetores é uma estrutura de dados muito utilizada.
- É importante notar que vetores, matrizes bidimensionais e matrizes de qualquer dimensão são caracterizados por terem todos os elementos pertencentes ao mesmo tipo de dado.
- Para se declarar um vetor pode utilizar a seguinte forma geral:

```
Tipo_variável  nome_variável [tamanho];
```

Exemplo:

```
int num[5];
```

Vetores – Exemplo01

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
main()
{
    int num[5]; /*Declara vetor int de 5 posicoes*/
    printf ("\nEntre com 5 (cinco) numeros: ");
    scanf ("%d",&num[0]);
    scanf ("%d",&num[1]);
    scanf ("%d",&num[2]);
    scanf ("%d",&num[3]);
    scanf ("%d",&num[4]);
    printf ("\n\n O segundo numero do vetor e: ");
    printf (" %d",num[1]);
    printf ("\n\n O quarto numero do vetor e: ");
    printf (" %d",num[3]);
    system("pause");
}
```

Vetores – Exemplo02

```
#include <stdio.h>
#include <conio.h>
main()
{
    int i;
    int vet[5];
    printf("Digite 5 numeros para o Vetor:\n\n");
    scanf("%d", &vet[0]);
    scanf("%d", &vet[1]);
    scanf("%d", &vet[2]);
    scanf("%d", &vet[3]);
    scanf("%d", &vet[4]);
    for(i=0; i < 5; i++)
    {
        printf("Valor de vet[%d] = %d\n", i, vet[i]);
    }
    system("pause");
}
```


Vetores – Exemplo03

```
#include <stdio.h>
#include <conio.h>
main()
{
    int i;
    int vet[5];
    printf("Digite 5 numeros para o Vetor:\n\n");
    for(i=0; i < 5; i++)
    {
        scanf("%d", &vet[i]);
    }
    for(i=0; i < 5; i++)
    {
        printf("Valor de vet[%d] = %d\n", i, vet[i]);
    }
    system("pause");
}
```

Vetores – Exercícios

01 – De acordo com os exemplos anteriores, altere o tamanho do vetor, de acordo com um valor que será definido pelo usuário. Depois preencha todo o vetor e imprima seus valores.

02 – Depois complemente o programa com uma opção de abortar a digitação dos valores do vetor caso o usuário digite (-1) mesmo o vetor não estando cheio.