

## Lista de Exercícios 5 – Vetores/Strings

**Exercício 1** - Escreva um programa que leia e mostre um vetor de 20 elementos inteiros. A seguir, conte quantos valores pares e ímpares existem no vetor.

**Exercício 2** – Faça um programa que leia um vetor de 20 inteiros positivos (permita apenas que valores corretos sejam digitados, pedindo para o usuário repetir cada número negativo até que ele digite um positivo) e imprima todos os valores abaixo da média desses valores.

**Exercício 3** - Escreva um programa que leia dois vetores de 10 posições e faça a multiplicação dos elementos de mesmo índice, colocando o resultado em um terceiro vetor. Mostre o vetor resultante.

**Exercício 4** - Faça um algoritmo que leia um vetor de 20 posições e que em seguida percorra o vetor e encontre o menor e o maior valor. Mostre-os juntamente com seus respectivos índices no vetor.

**Exercício 5** - Escreva um programa que leia um vetor de 20 posições e mostre-o na tela. Em seguida, troque o primeiro elemento com o último, o segundo com o penúltimo, o terceiro com o antepenúltimo, e assim sucessivamente. Mostre o novo vetor depois da troca.

**Exercício 6** – Escreva um programa que teste a passagem de parâmetro de vetores. Dentro da função main crie um vetor de de inteiros de tamanho qualquer. Solicite ao usuário que insira os dados para cada uma das posições e exiba os números na tela. Em seguida chame a função ‘void altera\_vetor (int vet[])’ que receberá como parâmetro o vetor preenchido no main. A função deverá alterar alguns dos valores do vetor manualmente (ex: vet[0]=-1, vet[1]=10) e não deverá retornar nada. Após a chamada da função altera\_vetor, no método main novamente, exiba novamente os valores em cada uma da posição do vetor e verifique se os valores estão diferentes.

**Exercício 7** - Codifique um programa que leia e armazene duas notas de 10 alunos e calcule e exiba as respectivas médias. O programa deve utilizar:

1 - Um procedimento (função) que receba dois vetores de notas decimais e o tamanho dos vetores e peça ao usuário para digitar as notas, de forma a preencher os vetores.

2 - Um segundo procedimento deve receber os dois vetores de notas decimais e um vetor de médias decimais, bem como o tamanho dos vetores, e calcular a média das notas, preenchendo o vetor de médias.

3 - Um terceiro procedimento deve receber os vetores de notas e médias, além do tamanho dos mesmos, e imprimir as informações de cada aluno (utilizando 2 casas decimais onde for preciso) no seguinte formato:

Aluno 1

nota 1 : 7 . 5 0

nota 2 : 8 . 5 0

media : 8 . 0 0

**Exercício 8** – Codifique um programa que leia e armazene a matrícula (um número inteiro) e o salário de 5 funcionários utilizando vetores. Os funcionários cuja matrícula for um número par devem receber um aumento de 15%; já os funcionários cuja matrícula for um número ímpar, devem receber um aumento de 20%. Utilize procedimentos para ler os salários, aplicar o reajuste salarial e imprimir as informações, no seguinte formato:

```
Funcionario 1
matricula : 2784
salario base : R$ 7 8 6 0 . 5 0
percentual de aumento : 15
salario corrigido : R$ 9 0 3 9 . 5 7
```

**Exercício 9** – Crie uma função com cabeçalho `int strichr (char str[],char c)` que determina a primeira ocorrência do caractere representado pela variável `c` na string representada por `str`. Deve ser retornada a posição de ocorrência ou -1 caso o caractere não ocorra na string.

**Exercício 10** – Um palíndromo é uma palavra ou frase, que é igual quando lida da esquerda para a direita ou da direita para a esquerda. Escreva um programa que leia uma string de até 50 caracteres, e imprima “Palíndromo”, caso a string seja um palíndromo, e “Não Palíndromo”, caso contrário. Assuma que só são usados caracteres minúsculos e sem acentos. Exemplos de palíndromo: ana, mussum, rodador.

**Exercício 11** – Faça uma nova função palindromo que também reconheça como palíndromo frases que podem ser lidas igualmente da esquerda para a direita ou da direita para a esquerda, desconsiderando-se os espaços em branco. Por exemplo, as frases “a sacada da casa” e “socorram me subi no onibus em marrocos” também devem ser reconhecidas como palíndromos.

**Exercício 12** - Implementar uma função que crie uma cópia de um string sem utilizar o método `strcpy`.

**Exercício 13** - Criar uma função `elimina_espacos` que elimine os espaços em branco de uma cadeia de caracteres. A função deverá receber dois arrays de caracteres, sendo que o primeiro será o array de destino e o segundo o array de origem. Para o texto de entrada “Joaquim Silva Pereira”, deve-se salvar o seguinte texto no array de destino: “JoaquimSilvaPereira”.

**Exercício 14** - Faça um programa que solicite ao usuário que digite uma frase, calcule e mostre a quantidade de vogais da frase digitada. O programa deverá reconhecer vogais maiúsculas e minúsculas.

**Exercício 15** - Crie uma função com cabeçalho `void strins (char str[], char c, int pos)` que insira o caractere representado pela variável `c` na string representada por `str` na posição `pos`. Atenção, não será feita a substituição, e sim a inserção. Por exemplo, dada a string de entrada “Joanesburgo” e o caractere ‘c’ e o inteiro 4, a string deverá ser modificada para “Joancesburgo”.

**Exercício 16** - Linguagem do P (Fácil). Crie um programa que leia uma frase inteira, insira o caractere P no lugar de cada vogal e imprima a frase resultante ao final.

**Exercício 17** - Linguagem do P (Desafio). Crie um programa que leia uma frase inteira, insira o caractere P ao início de cada vogal e imprima a frase resultante ao final. Não confunda inserção com substituição. Por exemplo, para a string de entrada “Antonio” a saída será “pAntponpipo”. Dica: utilize o método `strins` desenvolvido no exercício 16.