



Lista 2 - Revisão Geral

Exercícios

1) Gerar um vetor que não contenha números repetidos, com 10 elementos aleatórios entre 0 e 50. Utilizar uma função para verificar se cada elemento do vetor é ou não um número primo. Essa função retorna 0 para indicar que o número é primo ou 1 para indicar que o número não é primo. Criar uma matriz 10x3 onde a primeira coluna vai armazenar o índice do vetor, a segunda coluna vai armazenar o elemento do vetor e a terceira coluna vai armazenar 0 se o elemento do vetor for um número primo ou 1 caso o elemento do vetor não seja um número primo. Fazer a média dos números primos encontrados no vetor. A saída do programa deve ser apresentada como no exemplo a seguir:

Exemplo:

```
==== VETOR ====
24    1    21   37   42    8   33   23   28   17

INDICE  NUMERO  PRIMO
0       24     1
1        1     0
2       21     1
3       37     0
4       42     1
5        8     1
6       33     1
7       23     0
8       28     1
9       17     0

Media dos numeros primos: 19.5

Deseja repetir o programa (S/s)?
```

2) Ler uma string e copiar o valor decimal do código ASCII de cada elemento da string para um vetor. Mostrar o vetor. Ordenar o vetor. Mostrar o vetor ordenado. Percorrer o vetor ordenado contando, sem repetição, os números pares e os números ímpares e mostrar a soma total desses valores. Criar um vetor chamado vPares e um vetor chamado vImpares e copiar, sem repetição, os elementos pares e ímpares para os respectivos vetores. Mostrar os dois vetores. Criar uma matriz chamada mPares e uma matriz chamada mImpares, sendo que mPares deverá conter na primeira coluna todos os elementos pares do vetor e na segunda coluna a

quantidade de ocorrências de cada elemento par no vetor e na matriz mImpares fazer o mesmo com os elementos ímpares. Mostrar as duas matrizes.

Exemplo:

```
Informe uma string: Eu gosto de estudar

== VETOR ==
69 117 32 103 111 115 116 111 32 100 101 32 101 115 116 117 100 97 114

== VETOR ORDENADO ==
32 32 32 69 97 100 100 101 101 103 111 111 114 115 115 116 116 117 117

Numero de pares (sem repeticao): 4
Numero de impares (sem repeticao): 7

== VETOR PAR ==
32      100      114      116

== VETOR IMPAR ==
69      97      101      103      111      115      117

== MATRIZ DE PARES ==
PARES  OCORRENCIAS
32      3
100     2
114     1
116     2

== MATRIZ DE IMPARES ==
IMPARES OCORRENCIAS
69      1
97      1
101     2
103     1
111     2
115     2
117     2
```

3) Ler uma string e armazenar em um vetor o valor decimal ASCII apenas dos caracteres que fazem parte do alfabeto. Mostrar o vetor. Ordenar o vetor. Mostrar o vetor ordenado. Em uma segunda string, armazenar os caracteres ordenados do vetor e mostrar a string.

```
Informe uma string: Eu gosto de estudar

String 1: Eu gosto de estudar

== VETOR LETRAS ==
69 117 103 111 115 116 111 100 101 101 115 116 117 100 97 114

== VETOR LETRAS ORDENADO ==
69 97 100 100 101 101 103 111 111 114 115 115 116 116 117 117

String 2: Eaddeegoorssttuu
```

4) Ler uma string e contar os espaços em branco (sucessivos ou não) e armazenar a contagem em um vetor. Mostrar o vetor. Percorrer o vetor copiando apenas os números que são primos para um novo vetor. Mostrar o novo vetor (de números primos).

Exemplo: -Eu--gosto---de-programar----em-Linguagem-----de---Programacao-C-----

```
Informe uma string: Eu gosto de programar em Linguagem de Programacao C

==== STRING ====
Eu gosto de programar em Linguagem de Programacao C

==== VETOR ====
1 2 3 1 4 1 5 3 1 6

==== PRIMOS ====
2 3 5 3

Deseja repetir o programa (S/s para Sim)?
```

5) Fazer um programa que mostra os caracteres repetidos de uma string.

Exemplo:

```
Informe uma string: linguagem de programacao

==== CARACTERES REPETIDOS ====
aegmor
```

6) Fazer um programa que retira todos os caracteres repetidos de uma string.

7) Desafio: Strings que gaguejam

Criar uma função que duplique cada vogal da string.