

---

## ALGORITMOS II

### 1ª LISTA DE EXERCÍCIOS

- 1 Escreva um programa para converter uma cadeia de caracteres de letras maiúsculas em letras minúsculas. Não é possível utilizar a função *tolower()*. **Dica:** some 32 dos caracteres cujo código ASCII está entre 65 e 90.
- 2 Leia uma cadeia de caracteres e converta todos os caracteres para maiúscula. Não é possível utilizar a função *toupper()*. **Dica:** subtraia 32 dos caracteres cujo código ASCII está entre 97 e 122.
- 3 Faça uma função que compare duas *strings* (não utilize a função *strcmp()*). Escreva um programa que utilize a função desenvolvida.
- 4 Faça um programa que conte o número de 1's que aparecem em uma *string*.  
Exemplo: "0011001" -> 3
- 5 Escreva um programa que substitua as ocorrências de um caractere '0' em uma *string* por outro caractere '1'.
- 6 Faça um programa que receba uma palavra e a imprima de trás-para-frente.
- 7 Faça um programa que receba do usuário uma *string*. O programa imprime a *string* sem suas vogais.
- 8 Faça um programa que receba uma *string* e calcule quantas vogais (a, e, i, o, u) possui essa *string*.
- 9 Faça um programa que leia uma *string* e um caractere (vogal ou consoante) e, substitua todas as vogais da *string* dada por esse caractere.
- 10 Faça um programa em que troque todas as ocorrências de uma letra  $L_1$  pela letra  $L_2$  em uma *string*. A *string* e as letras  $L_1$  e  $L_2$  devem ser fornecidas pelo usuário.
- 11 Faça um programa que receba duas frases distintas e imprima de maneira invertida, trocando as vogais por \*.
- 12 Dados uma frase e uma palavra, escreva uma função que verifique o número de vezes que a palavra ocorre na frase. Faça um programa, com reprocessamento, que utilize esta função.

**Exemplo:**

Na frase ANA E MARIANA GOSTAM DE BANANA, a palavra ANA ocorre 4 vezes.

- 13 Escreva uma função que retorne o número de palavras de uma frase. Faça um programa, com reprocessamento, que utilize esta função.
- 14 Escreva uma função que mostre as palavras de uma frase. Faça um programa, com reprocessamento, que utilize esta função.
- 15 Escreva uma função que retorne a maior palavra de uma frase. Faça um programa, com reprocessamento, que utilize esta função.

- 
- 16 Faça um programa que leia uma palavra (máximo de 50 letras) e some 1 no valor ASCII de cada caractere da palavra. Imprima a *string* resultante.
- 17 Escreva uma função que classifique/ordene um vetor alfanumérico de no máximo de 45 elementos em ordem crescente. Faça um programa, com reprocessamento, que utilize essa função.
- 18 O código de César é uma das mais simples e conhecidas técnicas de criptografia. É um tipo de substituição na qual cada letra do texto é substituída por outra, que se apresenta no alfabeto abaixo dela um número fixo de vezes. Por exemplo, com uma troca de três posições, 'A' seria substituído por 'D', 'B' se tornaria 'E', e assim por diante. Implemente um programa que faça uso desse Código de César (3 posições), entre com uma *string* e retorne a *string* codificada.

**Exemplo:**

String: a ligeira raposa marrom saltou sobre o cachorro cansado

Nova string: D OLJHLUD UDSRVD PDUURP VDOWRX VREUH R FDFKRUUR FDQVDGR

- 19 Faça um programa que, dada uma string, diga se ela é um palíndromo ou não. Lembrando que um palíndromo é uma palavra que tenha a propriedade de poder ser lida tanto da direita para a esquerda como da esquerda para a direita.

**Exemplo:**

ovo

arara

Socorram-me, subi no onibus em Marrocos

Anotaram a data da maratona

- 20 Faça uma função que receba por parâmetro uma *string* e retorne a mesma sem os espaços em branco. O programa principal deve mostrar a *string* resultante.