



Lista 03 – String.

1. Faça um programa em C de um algoritmo que receba como entrada a partir do teclado uma *string* e o mostre na tela o número de caracteres digitados. Compare as funções `scanf()` e `gets()`.
2. Desenvolva um algoritmo de um programa para ler uma string do teclado e converter todos os caracteres para maiúsculo e mostre a string final na tela. Implemente o algoritmo em C.
3. Faça um programa que recebe uma string e um caractere, e retorne o número de vezes que esse caractere aparece na string.
4. Escreva um programa que recebe uma string e um caractere e apague todas as ocorrências desses caractere na string.
5. Faça um programa em C que receba uma string e mostre na tela quantas vezes aparece cada caractere.
Ex.: “Essa é a string digitada”
O caractere 'E' aparece 1 vez;
O caractere 's' aparece 3 vez;
O caractere 'a' aparece 4 vez;
O caractere 'é' aparece 1 vez;
O caractere 't' aparece 2 vez;
O caractere 'r' aparece 1 vez;
O caractere 'i' aparece 3 vez;
O caractere 'n' aparece 1 vez;
O caractere 'g' aparece 2 vez;
O caractere 'd' aparece 2 vez;
O caractere ' ' aparece 4 vez; // entenda-se o espaço
6. Faça um programa que retire todos os caracteres repetidos de uma string.
7. Faça um programa no qual o usuário informe uma string e um numero inteiro positivo (N) e o programa mostre na tela os caracteres da string correspondentes de 0 à N. O programa deve verificar se o valor N é compatível com o tamanho da string e informar ao usuário pedindo para informar o número novamente.
8. Faça um programa que receba uma palavra e mostre na tela a palavra em ordem inversa.
9. Faça um algoritmo e implemente um programa em C que verifique se a string é um palíndromo, ou seja é uma frase ou palavra que pode ser lida tanto da esquerda para direita quanto da direita para esquerda. Ex. OVO. Obs. Ignore a diferença entre letras maiúsculas e minúsculas.
10. Faça um algoritmo e implemente um programa em C que receba como entrada uma frase e mostre na tela apenas as vogais.
11. Faça um programa que faça a troca de uma letra de uma frase. O usuário deve digitar a string e a letra à ser substituída.
Ex. :Digite uma string : **“Essa é a string digitada”**
Digite a letra que será substituída: **s**
Digite a letra substituta: **G**

O programa deve mostrar na tela : “EGGa é a Gtring digitada”

12. Faça um programa que :

1. Crie e leia duas palavras (Com o mesmo tamanho, o programa deve fazer essa verificação e informar o usuário)
2. Crie e construa uma nova palavra que seja a intercalação letra a letra da primeira com a segunda palavra Ex:
s1: “Olá mundo”
s2: “Olá world”
resultado: “OOLLáá mwuonrdlod”

13. Faça um programa em C que leia duas palavras e diga qual delas vem primeiro na ordem alfabética. Bônus: verifique caso o usuário não digite apenas letras.

14. Faça um programa que receba uma frase, armazene em uma string e retire todos os espaços em branco, mostrando o resultado na tela em seguida.

15. Desenvolva um algoritmo e implemente em C de um programa que receba uma string e dois números inteiros positivos (x e y) e mostre na tela o trecho da string entre os caracteres x....y.

Ex: “**Essa é a string digitada**”

O valor de x é 1 e y é 5.

O resultado apresentado na tela deve ser: “**ssa é**”

Obs.: O programa deve conferir se o intervalo x e y digitados estão de acordo com o tamanho da string e caso não estejam o programa deve informar ao usuário e pedir que o mesmo digite outros valores. Bônus implemente o programa em um laço perguntado ao usuário se ele deseja continuar ou sair.