

Exercícios sobre STRINGS

1. Faça um programa em C que inicialize a string `s[50]` com `n` vezes o caracter `ch`.
2. Faça um programa em C que inverta os caracteres de uma string. Por exemplo, se a string for “TELEINFORMATICA”, deve ser convertida a “ACITAMROFNIELET”.
3. Faça um programa em C que leia uma string `s`, um caracter `chAtual`, um caracter `chNovo` e substitua todo caracter `chAtual` de `s` pelo `chNovo`. O programa deve retornar também o número de substituições.
4. Faça um programa em C que leia uma string `s`, um caracter `ch`, um inteiro `pos` e insira o caracter `ch` na posição `pos` da string `s`.
5. Faça um programa em C que leia uma string `s1`, uma string `s2`, um inteiro `pos` e insira a string `s2` em `s1` na posição `pos`.
6. Faça um programa em C que leia uma string `s1`, uma string `s2`, um inteiro `n` e copie os `n` primeiros caracteres da string `s1` na string `s2`.
7. Faça um programa em C que leia uma string `s1`, uma string `s2`, um inteiro `n` e copie os `n` últimos caracteres da string `s1` na string `s2`.
8. Faça um programa em C que leia uma string `s1`, uma string `s2`, um inteiro `n`, um inteiro `inicio` e copie os `n` caracteres a partir da posição `inicio` da string `s1` na string `s2`.
9. Escreva um programa em C para ler uma palavra e escrever:
 - A primeira letra da palavra.
 - A última letra da palavra.
 - O número de letras existente na palavra.
10. Ler um valor `n` que representa o número de pares de palavras (2 palavras) que serão lidas. A seguir ler os `n` pares e imprimir para cada par:
 - IGUAIS se as palavras informadas (do par) forem iguais.
 - ORDEM CRESCENTE se as palavras (do par) foram informadas em ordem crescente.
 - ORDEM DECRESCENTE se as palavras (do par) foram informadas em ordem decrescente.
11. Escreva um programa em C para ler 3 palavras. A seguir imprimir as 3 palavras em ordem alfabética..
12. Escreva um programa em C para ler uma frase e imprimir o número de caracteres dessa frase (não utilizar a função `strlen`).
13. Escreva um programa em C para ler um caractere e logo após um número indeterminado de frases. Para cada frase informada imprimir o número de ocorrências do caractere na frase. O programa deve ser encerrado quando a frase digitada for a palavra "fim".

14. Escreva um programa em C para ler uma frase e contar o número de ocorrências de cada uma das 5 primeiras letras do alfabeto. Imprimir as contagens.

15. Escreva um programa em C para ler uma frase. A seguir converter todas as letras minúsculas existentes na frase para maiúsculas. Escrever a frase modificada.

16. Escreva um programa em C para ler uma frase e uma letra. A seguir retirar da frase, todas as letras iguais a informada. Imprimir a frase modificada.

17. Escreva um programa em C para ler uma frase e contar o número de palavras existentes na frase. Considere palavra um conjunto qualquer de caracteres separados por um conjunto qualquer de espaços em branco.

18. Elabore um programa que receba uma linha de texto e conte as vogais apresentando o respectivo histograma na seguinte forma: Exemplo:

Linha de texto passada: "Na próxima quarta-feira e feriado."

```
a :  * * * * *
e :  * * *
i :  * * *
o :  * *
u :  *
```

19. Implemente um programa que receba uma linha de texto, retire os espaços em excesso existentes deixando apenas um espaço entre as várias palavras.

20. Implemente um programa que receba um nome completo de uma pessoa e apresente de acordo com os exemplos abaixo:

Exemplo 01:

Entrada: Manuel Francisco Teixeira Duarte

Saída: DUARTE, F. T. Manuel

Exemplo 02:

Entrada: Joaquim José da Silva Xavier

Saída: XAVIER, J. J. da Silva (observar os casos em que aparecem, por exemplo, o "da")

21. Escreva um programa que leia 3 strings, depois crie uma 4ª string que contenha as 3 strings digitadas concatenadas, exibindo-a no final.

22. Escreva um programa que leia uma String e mostre a quantidade de vogais na String.

23. Escreva um programa que leia uma String e mostre a quantidade de palavras na String.

24. Escreva um programa que leia uma String correspondendo ao nome de uma pessoa e crie uma nova String contendo apenas as iniciais do nome. (Ex: "João da Silva" => "JdS")

25. Escreva um programa que leia uma String e crie uma outra String que seja o inverso da primeira.

26. Escreva um programa que receba uma string e imprima uma tabela que informe o número de ocorrências de cada caracter na string.

27. Construir um programa que seja capaz de embaralhar uma string s1 com uma string s2 e colocar o resultado em uma string s3. Para embaralhar s1 com s2 é necessário preencher os índices pares de s3 com os elementos de s1 e os ímpares com os elementos de s2 até que os elementos de uma das duas strings termine e os demais elementos de S3 serão preenchidos com os elementos da string restante. Considere o índice 0 (zero) como sendo par. Por exemplo:

- s1 = local
- s2 = misterio.
- Nova string s3 = lmoicsatlerio

28. Escreva um programa que receba uma string e converta todos os caracteres dos índices pares para maiúsculo e todos os caracteres dos índices ímpares para minúsculo – exibindo o resultado na tela.

(Ex: "arara" => "aRaRa").

29. Fazer um programa em "C" que lê um string qualquer de no máximo 80 caracteres e imprime:

- Quantos caracteres tem o string;
- Quantos caracteres são de pontuação;
- Quantos caracteres são números;
- Quantos caracteres são minúsculas.

30. Fazer um programa em "C" que lê um string contendo palavras separadas por um espaço em branco cada e as imprime uma abaixo das outras.

32. Fazer um programa em "C" que pergunta o nome, o endereço, o telefone e a idade de uma pessoa e monta um string com a seguinte frase:

"Seu nome é ..., você tem ... anos, mora na rua ... e seu telefone é"

33. Fazer um programa em "C" que solicita um número inteiro e soletra o mesmo na tela.

Entrada: 124

Saída: um, dois, quatro

34. Escreva um programa em "C" que receba um string um caracter e o índice de uma posição do string como parâmetro e insira o caracter na posição "empurrando" todos os demais para o lado.

35. Fazer um programa em "C" que recebe um string como parâmetro e devolve o endereço do primeiro caracter branco encontrado.