



PONTIFICIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

Instituto de Ciências Exatas e Informática

Departamento de Ciência da Computação

PUC Minas

Disciplina	Curso	Turno	Período
Algorítmos e Estruturas de Dados I	Ciência da Computação	Manhã	1º
Professor	Felipe Cunha (felipe@pucminas.br)		

Lista de Exercícios 09 - “*string.h*”

1. Crie um programa em C que leia uma string e determine o seu comprimento, sem utilizar a função **strlen**.
2. Crie um programa em C que leia um nome e exiba-o somente se a primeira letra for a letra ‘a’ ou ‘A’.
3. Crie um programa em C que leia duas strings e compare se elas são iguais, sem utilizar a função **strcmp**.
4. Repita o exercício anterior, mas agora utilizando a função **strcmp** para realizar a comparação entre as duas strings.
5. Crie um programa em C que leia duas strings e as concatene em uma terceira string, sem utilizar a função **strcat**.
6. Repita o exercício anterior, mas agora utilizando a função **strcat** para realizar a concatenação.
7. Crie um programa em C que leia uma string e a copie para outra variável, sem utilizar a função **strcpy**.
8. Repita o exercício anterior, mas agora utilizando a função **strcpy** para realizar a cópia da string.
9. Crie um programa em C que leia uma palavra e conte quantas vogais (a, e, i, o, u) ela possui. Em seguida, leia um caractere (vogal ou consoante) e substitua todas as vogais da palavra por esse caractere.
10. Crie um programa em C que leia uma string e a exiba na tela de forma invertida (de trás para frente).
11. Crie um programa em C que leia uma string e verifique se ela é um palíndromo ? ou seja, se pode ser lida da mesma forma da esquerda para a direita e da direita para a esquerda (por exemplo: **osso, arara, reviver**).
12. Crie um programa em C que leia uma cadeia de caracteres no formato ‘DD/MM/AAAA’ e extraia o dia, o mês e o ano para três variáveis inteiras. Antes disso, verifique se as barras estão nas posições corretas e se os valores de DD, MM e AAAA são numéricos.
13. Crie um programa em C que leia uma string contendo um endereço de e-mail e verifique se ele é válido. Considere um e-mail válido aquele que contém exatamente um caractere ‘@’ e pelo menos um ponto (‘.’) apóis o caractere ‘@’.
14. Crie um programa em C que leia uma string e substitua todas as ocorrências de um caractere específico por outro caractere fornecido pelo usuário.

15. Crie um programa em C que leia uma string e conte o número de palavras presentes nela. Considere que as palavras são separadas por espaços em branco.
16. Crie um programa em C que leia uma string e converta todas as letras minúsculas para maiúsculas, sem utilizar a função **strupr**.
17. Crie um programa em C que leia uma string e converta todas as letras maiúsculas para minúsculas, sem utilizar a função **strlwr**.