
	<p>Ministério da Educação Universidade Federal de Santa Maria Curso Técnico em Informática para Internet Integrado ao Ensino Médio Colégio Técnico Industrial de Santa Maria - CTISM</p>	
---	--	---

**Disciplina: Algoritmos e Programação**  
**Profª: Luciana Lourega**

### Exercícios

01) Construa um programa inicializando uma variável qualquer mostrando ao final seu resultado através de um ponteiro que aponta para outro ponteiro.



02) Faça um programa que seja capaz de comparar o endereço de dois ponteiros. Caso o primeiro endereço seja maior que o segundo imprima o resultado da subtração do primeiro endereço menos o segundo. Caso contrário imprima a subtração realizada entre o segundo endereço menos o primeiro.

03) Qual o valor de y no final do programa? Tente primeiro descobrir e depois verifique no computador o resultado. A seguir, escreva um /\* comentário \*/ em cada comando de atribuição explicando o que ele faz e o valor da variável à esquerda do '=' após sua execução.

```
int main()
{
    int y, *p, x;
    y = 0;
    p = &y;
    x = *p;
    x = 4;
    (*p)++;
    x--;
    (*p) += x;
    printf ("y = %d\n", y);
    return(0);
}
```

04) Escreva um programa que declare uma matriz 100x100 de inteiros. Você deve inicializar a matriz com zeros usando ponteiros. Preencha depois a matriz com os números de 1 a 10.000 usando ponteiros.

05) Construa um programa que contenha dois vetores um do tipo int e outro do tipo float de 10 posição já inicializados (0 até 9). Após mostre os valores que se encontram dentro dos dois vetores. Faça esse procedimento utilizando ponteiros.

	<p>Ministério da Educação Universidade Federal de Santa Maria Curso Técnico em Informática para Internet Integrado ao Ensino Médio Colégio Técnico Industrial de Santa Maria - CTISM</p>	
---	--	---

06) Implemente um programa C que leia 2 *strings* digitados pelo usuário e verifique se seus conteúdos são iguais. Este programa deve imprimir os *strings* digitados e uma mensagem dizendo que as *strings* são iguais ou diferentes.

07) Escreva um procedimento em linguagem **C** que recebe 3 cadeia de caracteres de tamanho máximo 50, e as imprime na tela em ordem alfabética.

08) Faça um programa que leia quatro palavras pelo teclado, e armazene cada palavra em uma string. Depois, concatene todas as strings lidas numa única string. Por fim apresente esta como resultado ao final do programa. Para fazer uma concatenação utiliza-se a função `strcat`, a qual apresenta a seguinte sintaxe:

`strcat (string_destino,string_origem);`

A string de origem permanecerá inalterada e será anexada ao fim da string de destino. Um exemplo:



```
#include <stdio.h>
#include <string.h>
int main ()
{
    char str1[100],str2[100];
    printf ("Entre com uma string: ");
    gets (str1);
    strcpy (str2,"Voce digitou a string ");
    strcat (str2,str1);/* str2 armazenará "Voce digitou a string" + o conteúdo
de str1 */
    printf ("\n\n%s",str2);
    return(0);
}
```

09) Escreva um programa em C que recebe via teclado um **nome** (máximo 256 caracteres). Logo após a entrada do nome imprima: **número de letras maiúsculas**, **número de letras minúsculas**, **número de vogais** e o **número de consoantes**, conforme exemplo abaixo:

**Exemplo:** Nome: **CTISM** <enter>  
(3) Letras maiúsculas  
(26) Letras minúsculas  
(14) Vogais  
(15) Consoantes

OBS: Pode ser utilizada a função `strchr` que é apelido da função [`strstr\(\)`](#), a qual tem por objetivo encontrar a primeira ocorrência de uma string. Sua sintaxe é:

`strchr(string, character)`

	<p>Ministério da Educação  Universidade Federal de Santa Maria  Curso Técnico em Informática para Internet Integrado ao  Ensino Médio  Colégio Técnico Industrial de Santa Maria - CTISM</p>	
---	--	---

10) Desenvolva um algoritmo onde o usuário entra com uma string e retorne como saída a string invertida. Para isso utilize a função `strrev()`, a qual possui a sintaxe: `strrev(string)`.