



Ponteiros

1. Faça um programa que entre com duas string's e verifique se a segunda está contida na primeira. Utilize uma função que recebe dois ponteiros, um para cada string. Através de operações com ponteiro, informar em qual a posição de memória se encontra o início da string2 em string1.
A função deve retornar um inteiro indicando a posição da string2 em string1. Caso não for encontrada, retornar -1.
2. Faça um programa que entre com o raio de uma circunferência e apresente em tela o perímetro e área da circunferência. Utilize uma função para fazer o cálculo que recebe as variáveis passadas por referência. Os valores de perímetro e área devem ser impressos na função main.
3. Faça um programa que receba como argumento da função main o dia, mês e ano (todos valores inteiros) e imprima a data por extenso. Exemplo:
Chamada: programa 27 08 1971
Saída em tela: 27 de agosto de 1971

4. Mostre o resultado e explique o que acontece no programa a seguir:

```
1. #include <stdio.h>
2. #include <stdlib.h>
3.
4. #define Y 2
5. #define X 2
6. int main(int argc, char *argv[])
7. {
8.     int **A;
9.     int i, j;
10.    A = (int **)malloc(Y * sizeof(int *));
11.    for (i = 0; i < Y; i++)
12.        A[i] = (int *)malloc(X * sizeof(int));
13.    for (i = 0; i < Y; i++)
14.        for (j = 0; j < X; j++)
15.            A[i][j] = i + j;
16.    for (i = 0; i < Y; i++)
17.        for (j = 0; j < X; j++)
18.            printf("%d \t", A[i][j]);
19.    for (i = 0; i < Y; i++)
20.        free(A[i]);
21.    free(A);
22.    return 0;
23. }
```

5. Faça um programa que entre com dois valores inteiros pelo teclado e utilizando apenas ponteiros, troque os valores das variáveis em uma função.