

## Uso de linguagem C ANSI para programação (função)

### Definição:

Basicamente o nome já diz tudo é uma função. Uma boa analogia é comparar uma estrutura de uma empresa, ela possui vários cargos como: financeiro, vendedor, almoxarife e outros. Cada cargo exerce uma ou mais funções específicas como o vendedor, que tem a função de fazer orçamento e vender o produto.

A função em programação tem o objetivo de segmentar a codificação para ficar mais fácil o entendimento de programas maiores, cada função tem só suas codificações específicas e assim reduzindo a codificação, pois poderá ser chamada em outros pontos do programa.

### Estrutura:

- Declaração do protótipo da função

[tipo de dado de saída da função] [nome da função] ([tipo de dados de entrada da função]);

- Declaração da função

[tipo de dado de saída da função] [nome da função] ([tipo de dados de entrada da função]){  
    return [dado de saída];  
}

### Exemplo:

```
#include <stdio.h>

// função sem entrada, sem saída
void funcao1 (void);
// função com entrada, sem saída
void funcao2 (int x);
// função sem entrada, com saída
int funcao3 (void);
// função com entrada, com saída
int funcao4 (int x);

int main (void){
    funcao1();
    funcao2(funcao3()+funcao4(3));
    getch();
    return 0;
}

// função sem entrada, sem saída
void funcao1 (void){
    printf("OLA\n");
}

// função com entrada, sem saída
void funcao2 (int x){
    printf("%i",x);
}

// função sem entrada, com saída
int funcao3 (void){
    return 3;
}
```

```

}

// função com entrada, com saída
int funcao4 (int x){
    return x+3;
}

```

**ou**

```

#include <stdio.h>

// função valida data
int ValidaData(int d, int m, int a);

int main (void){
    int D,M,A;
    printf("Digite a data (DD/MM/AAAA):");
    scanf("%i%c%i%c%i",&D,&M,&A);
    printf("Data: %i/%i/%i\n",D,M,A);
    if (ValidaData(D,M,A) == 0){
        printf("Data valida\n");
    }else{
        printf("Data invalida\n");
    }
    getch();
    return 0;
}

int ValidaData(int d, int m, int a){
    if (a >= 0){
        if (m >= 1 && m <= 12) {
            if (d >= 1 && d <= 31){
                return 0;
            }else{
                return 1;
            }
        }else{
            return 1;
        }
    }else{
        return 1;
    }
}

```

## Exercícios

1. Faça um programa que leia um número, passe este número como parâmetro para uma função que retorne 0 se o número for par ou 1 se o número for ímpar.  
R.:
2. Faça uma função que desenhe linhas de caracteres na tela, a função receberá como argumento o tipo de caractere e o número de linhas que deverá imprimir.  
R.:
3. Fazer um programa que possibilite várias opções de cálculos a partir de um menu. O programa chamará a função correspondente a cada cálculo.  
[ a ]  $S = 1/1 + 3/2 + 5/3 + \dots + 99/50$

[ b ]  $S = 1/1 - 2/2 + 3/3 - \dots - 10/10$

[ c ]  $S = 1000/1 - 997/2 + 994/3 - \dots$

[ d ]  $S = 480/10 - 475/11 + 470/12 - \dots$

[ f ] FIM

Obs.: Nas opções [c] e [d] fazer os cálculos para os 20 primeiros termos.

R.:

4. Faça uma função que receba como argumento os valores dos lados de um triângulo, a função deverá retornar 0 se triângulo for equilátero, 1 se for isósceles ou 2 se for escaleno.

R.:

5. Faça um programa que leia uma data, passe para uma função que reajuste esta data em cinco dias.

R.: