

# M11L: Funções

Murilo Dantas

**PROGRAMA EXEMPLO 1:** Faça um programa que calcule o imposto de renda de um contribuinte de um país imaginário onde as regras do imposto são as seguintes:

- Todos pagam a mesma alíquota, de 20%.
- São descontados da base de cálculo (proventos) as despesas com educação e despesas médicas.
- São descontados R\$ 1000,00 por dependente.
- O imposto devido ou a receber pode ser parcelado em até 6 vezes.
- Valores de imposto (devido ou a receber) abaixo de R\$100,00 não são cobrados nem pagos.

```
//Exemplo 1: "spaghetti"
#include <stdio.h>
#include <math.h>

#define ALIQUOTA 0.2f
#define DESC_POR_DEP 1000.f
#define PARC_MAX 6
#define PAGTO_MINIMO 100.f

void main()
{
    int numDep, parcelas;
    float proventos=0, descFonte, despMed, despEdu, impostoDevido;

    printf("+++++\n");
    printf("Programa Para Calculo do Imposto de Renda de Pessoa Fisica\n");
    printf("+++++\n");

    do {
        printf("\n\nEntre com os seguintes dados: \n\n");
        printf("Proventos (0 para encerrar): R$ ");
        scanf("%f", &proventos);

        if(proventos > 0)
        {
            do {
                printf("Total Descontado na Fonte: R$ ");
                scanf("%f", &descFonte);
            } while (descFonte < 0);

            do {
                printf("Total de Despesas Medicas: R$ ");
                scanf("%f", &despMed);
            } while (despMed < 0);

            do {
                printf("Total de Despesas com Educacao: R$ ");
                scanf("%f", &despEdu);
            } while (despEdu < 0);

            do {
                printf("Total de Dependentes: ");
                scanf("%d", &numDep);
            } while (numDep < 0);
```

```

        impostoDevido = proventos - numDep*DESC_POR_DEP - despMed -
despEdu;

        impostoDevido *= ALIQUOTA;
        impostoDevido -= descFonte;

        if (fabs(impostoDevido) <= PAGTO_MINIMO)
            printf("Nao hah imposto a pagar ou receber. \n");
        else if(impostoDevido > 0)
            printf("O imposto devido eh de: %.2f\n", impostoDevido);
        else
            printf("Imposto a ser restituído: %.2f\n", -impostoDevido);

        do {
            printf("No. de parcelas (max %d parcelas)? ", PARC_MAX);
            scanf("%d", &parcelas);
        }
        while(parcelas <= 0 || parcelas > PARC_MAX);

        printf("Parcela: R$ %.2f\n", fabs(impostoDevido/parcelas));
    }
} while(proventos>0);
}

```

//Exemplo 2: Com funções:

```
#include <stdio.h>
```

```
#include <math.h>
```

```
#define ALIQUOTA 0.2f
```

```
#define DESC_POR_DEP 1000.f
```

```
#define PARC_MAX 6
```

```
#define PAGTO_MINIMO 10.f
```

//Protótipos das Funções

```
void telaDeAbertura();
```

// Faz a apresentação do programa para o usuário

```
int dadosContribuinte(float *proventos, float *descFonte, float *despEdu, float *despMed, int
*numDep);
```

/\* Lê do usuário os dados do contribuinte dados como parâmetros:

\* proventos: valor anual recebido pelo contribuinte (valores <= 0 impedem a entrada dos outros dados)

\* descFonte: valor descontado na fonte durante o ano

\* despEdu: despesas com educação durante o ano

\* despMed: despesas médicas durante o ano

\* numDep: número de dependentes do contribuinte

\*/

```
float imposto(float proventos, float descFonte, float despEdu, float despMed, int numDep);
```

/\* Faz o cálculo do imposto, utilizando a alíquota estabelecida na constante ALIQUOTA

\* e descontando DESC\_POR\_DEP para cada dependente (numDep), além dos descontos

\* dados como parâmetro: despEdu e despMed.

\*/

```
void imprimeImposto(float impostoDevido);
```

// Imprime o imposto devido (impostoDevido positivo) ou a receber (impostoDevido negativo).

```
void parcelaImposto(float impostoDevido);
```

// Divide o imposto em um número que vai de 1 até PARC\_MAX

```

void main()
{
    int numDep;
    float proventos=0, descFonte, despMed, despEdu, impostoDevido;

    telaDeAbertura();
    while(dadosContribuinte(&proventos, &descFonte, &despEdu, &despMed, &numDep) != 0)
    {
        impostoDevido = imposto(proventos, descFonte, despEdu, despMed, numDep);
        imprimeImposto(impostoDevido);
        parcelaImposto(impostoDevido);
    }
}

// Implementação das Funções
void telaDeAbertura()
{
    printf("+++++\n");
    printf("Programa Para Calculo do Imposto de Renda de Pessoa Fisica\n");
    printf("+++++\n");
}

int dadosContribuinte(float *proventos, float *descFonte, float *despEdu, float *despMed, int
*numDep)
{
    printf("Entre com os os seguintes dados: \n\n");
    printf("Proventos (0 para encerrar): R$ ");
    scanf("%f",proventos);

    if(*proventos > 0)
    {
        do {
            printf("Total Descontado na Fonte: R$ ");
            scanf("%f", descFonte);
        } while (descFonte < 0);

        do {
            printf("Total de Despesas Medicas: R$ ");
            scanf("%f", despMed);
        } while (*despMed < 0);

        do {
            printf("Total de Despesas com Educacao: R$ ");
            scanf("%f", despEdu);
        } while (*despEdu < 0);

        do {
            printf("Total de Dependentes: ");
            scanf("%d", numDep);
        } while (*numDep < 0);
        return 1;
    }
    else
        return 0;
}

float imposto(float proventos, float descFonte, float despEdu, float despMed, int numDep)
{
    if(proventos > 0)
    {

```

```

        float impostoDevido;
        impostoDevido = proventos - numDep*DESC_POR_DEP - despMed - despEdu;
        impostoDevido *= ALIQUOTA;
        impostoDevido -= descFonte;
        return impostoDevido;
    }
    else
        return 0.f;
}

void imprimelImposto(float impostoDevido)
{
    if (fabs(impostoDevido) <= PAGTO_MINIMO)
        printf("Nao hah imposto a pagar ou receber. \n");
    else if(impostoDevido > 0)
        printf("O imposto devido eh de: %.2f\n", impostoDevido);
    else
        printf("O imposto a ser Restituído eh de: %.2f\n", -impostoDevido);
}

void parcelaImposto(float impostoDevido)
{
    int parcelas;
    do {
        printf("Dividir em quantas parcelas (max %d parcelas)? ", PARC_MAX);
        scanf("%d", &parcelas);
    }
    while(parcelas <= 0 || parcelas > PARC_MAX);

    printf("O valor de cada parcela eh de %.2f\n", fabs(impostoDevido/parcelas));
}

```

**Atenção: mesmo os exercícios onde isso não é explicitamente requisitado, faça programas para testar as suas funções.**

1. Escreva uma função que imprima seus dados pessoais: nome completo, endereço, telefone e e-mail. A seguir, faça um programa que utilize esta função.
2. Escreva uma função que imprima seus dados profissionais: nome, endereço e telefone da empresa, função, salário e data de admissão. A seguir faça um programa que apresente continuamente (em loop) na tela um menu com as seguintes opções (1-imprimir dados pessoais; 2 – imprimir dados profissionais; 3 – sair do programa). Utilize as funções criadas anteriormente para fazer as impressões.
3. Escreva uma função (FparaC) que receba uma temperatura em graus F e retorne a temperatura em graus C, sendo:  $C = \frac{5}{9}(F - 32)$ . A seguir, faça um programa que, em loop, leia um valor para F da entrada padrão e o imprima o valor de C correspondente, utilizando a função FparaC.
4. Escreva as seguintes funções:

- a. CparaF – faz a conversão de uma temperatura em graus C para graus F.
- b. CparaK – faz a conversão de uma temperatura em C para Kelvin ( $C=K-273$ )
- c. KparaC – faz a conversão de K para C.
- d. KparaF – faz a conversão de K para F (dica: utilize as funções anteriores)
- e. FparaK – faz a conversão de F para K.

A seguir, faça um programa que apresente continuamente um menu na tela com todas as opções de conversão que você implementou. Uma vez feita a opção, o programa lê do teclado o valor a ser convertido e imprime o resultado.

5. A multiplicação entre dois números inteiros pode ser definida como uma repetição da adição de um deles. Exemplo:  $3 \times 4 = 4 + 4 + 4$   
 Escreva uma função que multiplique dois números inteiros utilizando esse método. A seguir, escreva um programa que peça ao usuário um número inteiro e imprima a tabuada para aquele número (de 1 à 10) utilizando a função construída.

6. O seno de um ângulo em radianos, no intervalo de 0 à  $\frac{\pi}{2}$  pode ser calculado através

da série de McLaurin, apresentada a seguir:  $\sin x = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$

- a. Escreva uma função que converta um ângulo em graus para seu valor em radianos ( $180^\circ = \pi \text{ rad}$ )
- b. Escreva uma função que receba como parâmetro um ângulo em graus, a precisão requerida para o cálculo e retorne o seu seno, utilizando a função de conversão graus-radiano feita anteriormente
- c. Faça um programa que teste a sua função para cálculo do seno.

7. O valor aproximado de  $\pi$  pode ser calculado a partir da série:  $\pi = \frac{4}{1} - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \dots$

Escreva uma função que calcule o valor de  $\pi$ , com precisão dada como parâmetro.

**PROGRAMA EXEMPLO 2:** Faça uma função que receba duas variáveis como parâmetros e troque os valores dessas duas variáveis entre si.

```
#include <stdio.h>

//funções de troca
void trocaErrada(int a, int b);
void trocaCerta(int *a, int *b);

// Programa para teste de troca errada e troca certa
void main()
{
    int x, y;

    printf("Valor de x? ");
    scanf("%d", &x);

    printf("Valor de y? ");
```

```

scanf("%d", &y);

trocaErrada(x, y);
printf(" x = %d e y = %d apos troca errada.\n ", x, y);

trocaCerta(&x, &y);
printf(" x = %d e y = %d apos troca certa.\n ", x, y);
}

//Implementação das Funções
void trocaErrada(int a, int b)
{
    int aux;
    aux = a;
    a = b;
    b = aux;
}

void trocaCerta(int *a, int *b)
{
    int aux;
    aux = *a;
    *a = *b;
    *b = aux;
}

```

8. Uma equação do segundo grau é escrita  $ax^2 + bx + c = 0$  e a sua solução é dada em função dos valores de a, b e c. Podendo ter duas raízes, uma ou nenhuma. Escreva uma função que resolva a equação do segundo grau, retornando o número de raízes encontradas. Os valores dessas raízes devem ser retornados em parâmetros.
9. Faça um programa que apresente na tela um menu com as seguintes opções: 1 – converter um ângulo em graus para radiano; 2 - calcular o seno de um ângulo, 3 – calcular o valor de  $\pi$ . 4 - resolver uma equação do segundo grau; 0 - sair. Depois de feita a opção, o programa deve chamar uma função que leia do usuário os parâmetros necessários para o cálculo escolhido e a seguir usar uma das funções que você já implementou.

**PROGRAMA EXEMPLO 3:** Escreva uma função que receba como parâmetro um vetor de números reais de tamanho qualquer e retorne a soma dos elementos do vetor.

```

#include <stdio.h>
//Protótipos das Funções
float somaVetor(float *v, int tam); //soma os "tam" elementos do vetor
void imprimeVetor(float *v, int tam); // imprime os "tam" elementos do vetor

//programa de teste
void main()
{
    float vetorExemplo[] = {3.5f, 6, 7.2f, -2.456f, 9, 0, 4.23f};

    printf("A soma dos elementos do vetor ");
    imprimeVetor(vetorExemplo, 7);
    printf(" eh de %.3f\n", somaVetor(vetorExemplo, 7));
}

```

```
//Implementação das Funções
float somaVetor(float *v, int tam)
{
    int i;
    float soma=0;
    for(i=0; i<tam; i++)
        soma += *(v+i);

    return soma;
}

void imprimeVetor(float *v, int tam)
{
    int i;

    printf("{%.1f", v[0]);
    for(i=1; i<tam; i++)
        printf(", %.1f", v[i]);
    printf("}");
}
```

10. Escreva uma função que receba como parâmetro a referência de um vetor de números reais de tamanho qualquer e retorne a média dos elementos do vetor.
11. Escreva uma função que receba como parâmetro a referência de um vetor de reais de tamanho qualquer e imprima os elementos desse vetor na forma { e1, e2, e3, ....., en}
12. Escreva uma função que receba como parâmetro a referência de um vetor de números reais de qualquer tamanho e retorne, através de parâmetros, o maior e o menor valores encontrados no vetor.
13. Utilizando as três funções anteriores, escreva um programa que leia do teclado um vetor de 15 números reais através de uma função, calcule a soma dos seus elementos, a média, o maior e o menor elementos utilizando as funções anteriores.