

## Funções

Tratam-se de estruturas que permitem ao usuário separar seus programas em blocos. Também são conhecidas como sub-rotinas.

### Estrutura geral de uma função em Linguagem C

```
tipo_de_retorno           nome_da_função           (declaração_de_parâmetros)  
{  
    corpo_da_função  
}
```

O **tipo\_de\_retorno** é o tipo de dado devolvido pela função. Por default é devolvido o tipo **int**, ou seja, uma função para qual não declaramos o tipo de retorno considera-se que esteja sendo devolvido um inteiro.

A **declaração\_de\_parâmetros** é uma lista com a seguinte forma geral: *tipo nome1, tipo nome2, ... , tipo nomeN*. Note que o tipo deve ser especificado para cada uma das N variáveis de entrada.

É na declaração de parâmetros que informamos ao compilador quais serão as entradas da função (assim como informamos a saída no tipo-de-retorno).

No **corpo\_da\_função** são especificados os comandos para que a função realize o propósito de sua existência, ou seja, é nele que se descreve o que a função faz.

### Exemplificando

**Exemplo 1)** A função a seguir não recebe parâmetro algum e também não devolve informação alguma. Seu objetivo é calcular o quadrado de um número inteiro.

```
void           quadrado           (void)  
{  
    int    num;  
    printf ( " Digite um número: " );  
    scanf ( "%d", &num );  
    printf ( "%d ao quadrado é %d", num, num*num );  
}
```

Note que, nesta função, como ela não recebe parâmetro algum, foi informado o tipo de dado **void** na **declaração\_de\_parâmetros**. Analogamente, como ela também não retorna tipo de dado algum, foi informado o tipo de dado **void** no **tipo\_de\_retorno**.

### Observando o conjunto da obra

```
/* Bibliotecas utilizadas */  
#include <stdio.h>  
#include <conio.h>  
  
/* Protótipos de funções */  
int main ();  
void quadrado (void);  
  
/* Construção das funções */  
void quadrado (void)  
{  
    int    num;  
    printf ("\n Digite um número: ");  
    scanf  ("%d", &num);  
    printf ("\n O quadro de %d , %d", num, num*num);  
}
```

```

/* Corpo do programa */
int main ()
{
    printf ("\n Teste com funções");
    quadrado();
    getch();
    return 0;
}

```

**Exemplo 2)** A função a seguir recebe um parâmetro inteiro, mas não devolve informação alguma. Seu objetivo é calcular o quadrado de um número inteiro.

```

void          quadrado          ( int          num)
{

    printf ( "%d ao quadrado é %d", num, num*num );

}

```

### Observando o conjunto da obra

```

/* Bibliotecas utilizadas */
#include <stdio.h>
#include <conio.h>

/* Protótipos de funções */
int main ();
void quadrado (int num);

/* Construção das funções */
void quadrado (int num)
{
    printf ("\n O quadro de %d , %d", num, num*num);
}

/* Corpo do programa */
int main ()
{
    int      num;
    printf ("\n Teste com funções");
    printf ("\n Digite um número: ");
    scanf ("%d" , &num);
    quadrado( num );
    getch();
    return 0;
}

```

**Exemplo 3)** A função a seguir recebe um parâmetro inteiro e também devolve um valor inteiro. Seu objetivo é calcular o quadrado de um número inteiro.

```

int          quadrado          ( int          num)
{

    return ( num*num );

}

```

## Observando o conjunto da obra

```
/* Bibliotecas utilizadas */
#include <stdio.h>
#include <conio.h>

/* Protótipos de funções */
int main ();
int quadrado (int num);

/* Construção das funções */
int quadrado (int num)
{
    return ( num*num );
}

/* Corpo do programa */
int main ()
{
    int num;
    printf ("\n Teste com funções");
    printf ("\n Digite um número: ");
    scanf ("%d" , &num);
    printf ("%d ao quadrado , %d" , num, quadrado( num ) );
    getch();
    return 0;
}
```

**Observação:** Uma função do tipo **void**, quando executada, realiza seu propósito mas não devolve valor algum. Sendo assim, o compilador não permite que funções **void** sejam usadas em expressões. Por exemplo, a chamada a seguir causaria um erro de compilação:

```
x = quadrado();
```

### Exercícios –

**Exercício 1)** Codifique a função chamada **rodizio**, que recebe como parâmetro o número da *placa* de um veículo e exibe o dia em que ele está no rodizio, sem devolver retorno algum.

**Exercício 2)** Codifique um programa em Linguagem C, que utilize uma função chamada **menu**.

A função menu deverá funcionar de maneira tal que seja apresentada a tela a seguir até que o usuário digite uma das 5 opções disponíveis, e em caso contrário deverá ser mostrada a mensagem “Opção inválida! Tente outra vez.”

<p style="text-align: center;">Cadastro de alunos</p> <ol style="list-style-type: none"><li>1. incluir</li><li>2. consultar</li><li>3. alterar</li><li>4. excluir</li><li>5. sair</li></ol> <p style="text-align: center;">Digite a opção desejada:</p>
---

Quando for digitada uma das opções válidas, a função deverá retornar o valor da opção desejada.

Ficará a cargo do programa principal analisar qual foi o valor retornado pela função **menu** e exibir na tela uma das seguintes mensagens: “Opção de inclusão solicitada”, “Opção de consulta solicitada”, “Opção de alteração solicitada”, “Opção de exclusão solicitada”, “Encerrando operações”.