**Função**

“Conjunto de instruções desenhadas para cumprir uma tarefa particular e agrupadas numa unidade com um nome para referenciá-la.” Treinamento em Linguagem C. Victorine Viviane Mizrahi. 2ed. Pág. 110.

Um programa em C pode conter várias funções, sendo que uma delas deve ser a função **main()**.

**Chamada de uma função**

Chamar uma função significa referenciar seu nome em alguma parte do programa principal, ou de outra função. Quando isso acontece o fluxo de execução, ou o controle do programa, é passado para a função, até que ela o ‘devolva’ ao ponto de chamada.

Exemplo:

O protótipo da função (ou cabeçalho da função) deve ser incluído no programa antes da sua chamada, uma vez que uma função não pode ser chamada antes de ser declarada. O protótipo permite que o compilador verifique a sintaxe de chamda da da função.

#include <stdio.h>

int primo(int); //protótipo da função

main(){

int n;

printf("digite um numero para verificar se o mesmo e primo: ");

scanf("%d", &n);

if (primo(n)) //chama a função e depois avalia o retorno lógico da função 1(verdadeiro) ou 0(falso)

printf("\no numero %d e primo", n);

A chamada da função é a referência ao seu nome, juntamente com os parâmetros que a mesma pede. Neste momento o controle de execução do programa passa para a função, e depois da execução da função (e do seu retorno), volta para o ponto da chamada.

else

printf("\no numero %d nao e primo", n);

}

int primo(n)

{

O cabeçalho da função, ou assinatura da função, possui o formato:

tipo\_da\_função nome\_da\_função(parâmetros da função)

int x, p;

p = 0;

x = n;

while (x >= 1) {

if (n % x == 0)

p++;

O retorno da função é o ponto da mesma onde ‘uma resposta’ é enviada ao ponto de chamada, indicado pela palavra **return**. O retorno deve ser compatível com o tipo da função. Neste exemplo a função primo é do tipo inteiro, e os dois possíveis retornos também.

x--;

}

if (p == 2)

return 1; //VERDADEIRO ou TRUE

else

return 0; //FALSO ou FALSE

}

**Tipo da função**

É determinado pelo tipo de dado que a função retorna

**Comando return**

Pode retornar somente um valor para o ponto de chamada. Pode retornar um valor constante, uma variável ou uma expressão, desde que sempre do tipo da função.

**Parâmetros de uma função**

São as variáveis da função que recebem os dados da mesma. São especificadas na assinatura da função, e devem ser ‘declarados’ da mesma forma que variáveis, no formato

tipo\_de\_dado nome\_do\_parâmetro

Uma função pode ter nenhum ou vários argumentos.

**Passagem de parâmetro por valor**

Indica que o valor que é passado para a função, no momento da chamada, é copiado para o parâmetro que o está recebendo. Desta forma, qualquer alteração na variável na função não será refletida na variável na chamada.

**Passagem de parâmetros por referência**

Indica que o parâmetro na função recebe o endereço de memória da variável na chamada da função. Desta forma, toda alteração que for executada na variável na função será refletida na variável na chamada.

**Funções que não retornam valor (VOID)**

São funções que não retornam nenhum tipo de valor (em Pascal são chamadas de PROCECIMENTOS ou PROCEDURES).