

# Programação Procedimental - Aulas Práticas usando a Linguagem C

Universidade Federal de Uberlândia - 2013

## 1 Criando e Executando Programas

O programa clássico a ser testado em qualquer linguagem de programação é aquele que imprime a mensagem *Hello World!*. Este pequeno programa, mostrado a seguir em linguagem C, permite conhecer sobre a edição do programa, a compilação, a carga, a execução e o local no qual o resultado será exibido. A execução deste programa depende do sistema operacional em uso.

```
#include <stdio.h>
int main()
{
    printf("Hello World!\n");
    return 0;
}
```

O exemplo define a criação de uma função chamada *main*. Normalmente podem-se criar funções usando nomes de livre escolha, mas o nome *main* define a função principal, a primeira a ser executada. Assim, todo programa deve ter uma função *main*. A função *main* pode chamar outras funções definidas pelo usuário e usar bibliotecas para realizar suas instruções. A primeira linha do programa

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

informa o compilador sobre a inclusão da biblioteca padrão para comandos de entrada/saída. O nome de uma função deve sempre ser seguido por uma lista de argumentos entre parênteses. Neste exemplo a função *main* não possui argumentos. Os comandos contidos na função são delimitados pelos símbolos { e }. Apenas uma instrução foi definida na função *main*, que é a impressão de uma mensagem usando-se a função *printf*, da biblioteca *stdio.h*. A sequência \n é a notação em C para o caracter de nova linha.

As funções que resumem os comandos de entrada e saída formatadas no C são as funções *printf* e *scanf*. O conhecimento destas funções é fundamental ao programador. Em

```
printf("Hello World!\n");
```

o único argumento da função é a mensagem *Hello World!* e corresponde a uma cadeia de caracteres, que será lida e em seguida escrita na tela. Quando são necessárias conversões, podemos ter:

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int i;
    i=10;
    printf("%d", i);
    return 0;
}
```

O trecho acima, especifica a conversão de um argumento do tipo `int`, que corresponde aos números inteiros. O número inteiro (10) será lido e os caracteres correspondentes ao número serão escritos na tela.

Em <http://programacaodescomplicada.wordpress.com/indice/linguagem-c/> podemos encontrar duas video-aulas sobre as instruções `printf` (3a. Aula) e `scanf` (4a. Aula).

## 1.1 Exercícios Propostos

1. Seja o programa abaixo:

```
#include<stdio.h>
int main()
{
    int a,b;
    printf("Entre com dois valores inteiros :");
    scanf("%d %d",&a,&b);

    printf("\nA soma de %d e %d resulta em %d",a,b,a+b);
    printf("\nA subtracao de %d e %d resulta em %d",a,b,a-b);
    printf("\nO produto de %d por %d resulta em %d",a,b,a*b);
    printf("\nO quociente de %d por %d resulta em %f\n",a,b,
           (float)a/(float)b);
    return 0;
}
```

Teste o programa e verifique se o mesmo funciona corretamente. Explique como funcionam as funções `printf` e `scanf`.

2. Escreva um programa que solicita ao usuário o valor do raio de uma circunferência e retorna os valores da área e perímetro.
3. Escreva um programa que resolva equações do segundo grau ( $ax^2 + bx + c = 0$ ). O usuário deve informar os valores de  $a, b$  e  $c$ , e o programa deverá retornar o número de soluções e os valores.
4. Escreva um programa que receba um número real  $x$  e um número inteiro  $n$ , e retorna o resultado de  $x$  elevado a potência  $n$ .
5. Escreva um programa que calcula a resistência equivalente à três valores de resistência  $R_1$ ,  $R_2$  e  $R_3$  (informados como números reais) considerando:
  - Valor para ligação em série:  $R_{eq} = R_1 + R_2 + R_3$
  - Valor para ligação em paralelo:  $R_{eq} = \frac{R_1 * R_2 * R_3}{R_1 * R_2 + R_1 * R_3 + R_2 * R_3}$
6. Escreva um programa que calcula a soma de quatro números inteiros, informados pelo usuário utilizando:
  - 5 variáveis (os valores informados são memorizados)
  - 2 variáveis (os valores informados são perdidos)
7. Escreva um programa que calcula e exiba a distância entre dois pontos A e B do plano, cujas coordenadas  $(X_a, Y_a)$  e  $(X_b, Y_b)$  são informadas pelo usuário.

## Referências

- [1] Salvetti, D. D., Barbosa M. B. *Algoritmos*, Makron Books, 1993.