

## Prática 02 – Programação Estruturas de Seleção, Repetição, E/S Formatada

### Desenvolva os seguintes algoritmos em linguagem de programação C:

1. Escreva um programa para ler um valor inteiro entre 1 e 12 e assumir que esse valor representa um mês. Escreva o nome do mês na tela.
2. Escreva um programa para classificar um triângulo, de lados de comprimentos dados pelo usuário, em *escaleno* (os três lados de comprimentos diferentes), *isósceles* (dois lados de comprimentos iguais) ou *equilátero* (os três lados de comprimentos iguais).
3. Leia dois números reais. Se o primeiro for maior, escreva "Maior"; se for menor, escreva "Menor"; e se forem iguais escreva "Iguais".
4. Escreva um programa para gerar um número aleatório e mostrá-lo na tela. Para gerar um número aleatório use as seguintes funções do C:

```
srand(time (NULL));
```

```
numero = rand();
```

Estabelece um número base para gerar todos os números aleatórios no programa; no caso, o número base será a hora atual em segundos, dada pela função **time**.

Após executar a função `srand` execute a função `rand` tantas vezes quantas forem necessárias para gerar números. Exemplo: se for preciso gerar 3 números aleatórios executar a função `rand` 3 vezes:

```
Num1 = rand();
```

```
Num2 = rand();
```

```
Num3 = rand();
```

5. Escreva um jogo de adivinhação. O programa vai gerar um número aleatório entre 0 (zero) e 100 (cem) e o usuário (jogador) tentará acertar o número secreto. Em cada tentativa, o usuário introduz o número palpitado e o computador indica se acertou, ou então se é maior ou menor que o número secreto. O jogo prossegue até o utilizador acertar, ou até atingir um número  $n$  (dado) de tentativas..

```
numero = rand()%10 + 1;
```

Gera um número – `rand`  
Calcula o resto da divisão do número por 10 - `%10`  
→ o resto será um número entre 0 e 9  
Soma 1 porque queremos entre 1 e 10

6. Escreva um programa que determine a soma dos quadrados dos  $n$  primeiros números naturais, dado  $n$ .

7. Num frigorífico existem 90 bois. Cada boi tem um número de identificação e um peso. Fazer um programa que leia o número de identificação e o peso de cada boi e ao final escreva qual é o boi mais gordo e o boi mais magro.
8. Escrever um programa em C que lê 20 valores, um de cada vez, e conta quantos deles estão em cada um dos intervalos  $[0, 25]$ ,  $(25, 50]$ ,  $(50, 75]$ ,  $(75, 100]$ , escrevendo esta informação.
9. Utilizando um laço "while", escreva um programa que exiba a mensagem "Senha inválida. Bomba não desarmada!" até que a senha "S" seja digitada. Tente incrementar seu programa com outras mensagens.
10. Deseja-se efetuar um levantamento da produção de  $n$  reservatórios de petróleo de determinada região e determinar qual produziu mais e qual produziu menos.
  - a. Ler do usuário o código do reservatório e sua produção;
  - b. Funcionar em um "loop" até que o usuário digite código 0 (zero) para reservatório.

Observações:

### FUNÇÃO GERADORA DE NÚMEROS ALEATÓRIOS

*rand()*;

- O protótipo da função *rand()* encontra-se no arquivo *stdlib.h*

A função *rand()* gera números aleatórios de 0 a *RAND\_MAX* (definido no cabeçalho *stdlib.h*). Se existe a necessidade de gerar valores randômicos em uma determinada faixa numérica, basta fazer: **numero= rand() % valor\_maximo;**

Os números "aleatórios" que o computador gera são baseados em um valor inicial fornecido, chamado *seed* (semente). Se o programador não fornecer esse número, ou se fornecer sempre o mesmo, a seqüência será sempre a mesma. Isto acontece porque a inicialização do gerador de números aleatórios é constante (valor). Para fornecer uma semente aleatória, use a função *srand()* com *time()*, onde o protótipo da função *time()* está no arquivo *time.h*. Assim: **srand(time(NULL));**

### FUNÇÕES PARA TRATAR CARACTERES:

**tolower()**; converte caractere maiúsculo em minúsculo. Se o caractere de entrada for minúsculo o retorno é o mesmo caractere.

**toupper()**; converte caractere minúsculo em maiúsculo. Se o caractere de entrada for minúsculo o retorno é o mesmo caractere.

- O protótipo das funções *tolower()* e *toupper()* encontram-se no arquivo *cctype.h*