

## Lista de Exercícios de Estruturas de Dados e Algoritmos

Q1) Escreva um programa que, infinitamente, leia um número inteiro  $n$ , que representa o tamanho, seguida de uma sequência de  $n$  números inteiros, e determine o comprimento máximo de um segmento crescente destes  $n$  números. Exemplos:

- Na sequência 5, 10, 3, 2, 4, 7, 9, 8, 5 o comprimento do segmento crescente máximo é 4.
- Na sequência 10, 8, 7, 5, 2 o comprimento de um segmento crescente máximo é 1.

Seu programa para quando  $n$  for menor ou igual a zero.

Q2) Implemente um programa que, infinitamente, receba, como parâmetro de entrada, um número  $n$  e retorne os  $n$  primeiros números primos existentes. Seu programa para quando  $n$  for menor ou igual a zero.

Q3) Implemente um programa que, infinitamente, receba, como parâmetro de entrada, um número  $n$  e um número  $m$ , e retorne todas as potências de  $m$ , menores que  $n$ . Por exemplo, se  $n = 1000$  e  $m = 2$ , seu programa deve imprimir os seguintes números: 1, 2, 4, 8, 16, 32, 64, 128, 256 e 512. Seu programa para quando  $n$  for menor que 1 ou quando  $m$  for 0, ou 1 ou -1.

Q4) Implemente um programa que, infinitamente, receba, como parâmetro de entrada, um número  $n$  e retorne os  $n$  primeiros números primos existentes depois de  $n$ . Por exemplo, se  $n = 2$ , a resposta será os primos 3 e 5. É necessário salientar que  $n$  não precisa ser primo. Seu programa para quando  $n$  for menor ou igual a zero.

Q5) Implemente um programa que, infinitamente, receba, como parâmetro de entrada, um número  $n$  e retorne a representação binária de  $n$ . Por exemplo, se  $n$  é igual a 12, a resposta deste programa deve ser "1100". Seu programa para quando  $n$  for menor que zero.

Q6) Supondo as seguintes declarações:

```
int x, y, z;  
int teste;  
x = 5;  
y = x++;  
z = x--;
```

Informe os valores de `teste`, `x`, `y` e `z` depois da avaliação das seguintes expressões:

- (a) `teste = !y == !x;`
- (b) `teste = ((x++ > y) || (--z <= y));`
- (c) `teste = ((!x) || (!(!z)));`
- (d) `teste = ((x + y) > z) && (x++);`
- (e) `teste = x && y + !3 || 4;`