**Exercícios de Fixação**

**Algoritmos com estruturas de iteração aninhadas**

1. Escrever um algoritmo que leia 20 valores para uma variável N e, para cada um deles, calcule a tabuada de 1 até N. Mostre a tabuada na forma:

**1 x N = N**

**2 x N = 2N**

**3 x N = 3N**

**......**

**N x N = N2**

1. Crie um programa que mostre a seguinte seqüência na tela, de 1 até 10:

**1 X o numero 1:**

**1**

**2 X o numero 2:**

**2**

**2**

**3 X o numero 3:**

**3**

**3**

**3**

**... ( e assim por diante até 10)**

1. Monte um programa que simule um relógio digital como o anterior porém só imprima os “quartos” de hora, ou seja:

**0:0**

**0:15**

**0:30**

**0:45**

**1:0**

**... ( e assim por diante até 23:45)**

1. Crie um programa que solicite liste todos os possíveis divisores de um número, de 1 até ele mesmo. Entende-se aqui que “divisor” é um número pelo qual pode-se dividir o valor desejado obtendo um quociente inteiro e resto zero. O número deverá ser solicitado ao usuário. Após a listagem dos divisores deste número o programa deverá pedir outro número até que o usuário digite o valor zero ou um valor negativo.
2. Escrever um algoritmo que gera e escreve os 4 primeiros números perfeitos. Um número perfeito é aquele que é igual a soma dos seus divisores. (Ex.: 6 = 1+2+3; 28= 1+2+4+7+14 etc).
3. Elabore um algoritmo que imprima todos os primeiros N números primos existentes, sendo que N é um número natural fornecido pelo usuário.
4. Escrever um algoritmo que lê um número não determinado de valores para M, todos inteiros e positivos, um de cada vez. Se M for par, verificar quantos divisores possui e escrever esta informação. Se M for ímpar e menor do que 10 calcular e escrever o fatorial de M. Se M for ímpar e maior ou igual a 10 calcular e escrever a soma dos inteiros de 1 até M.