

Lista de Exercícios

– Programação para Engenharia I-A –

1) Implementar um programa que leia um determinado número digitado pelo usuário e retorne a tabuada deste número (suas multiplicações de 1 até 10).

```
Private Sub cmdTab_Click()  
    Dim num As Integer  
    Dim cont As Integer  
    Dim mult As Integer  
  
    num = CInt(txtNumero.Text)  
  
    cont = 1  
  
    While cont <= 10  
        mult = cont * num  
        lstRel.AddItem (" " & cont & " x " & num & " = " & mult)  
        cont = cont + 1  
    Wend  
  
End Sub
```

2) Implementar um programa que calcule o fatorial de um número escolhido pelo usuário. O programa deve permitir o cálculo apenas de valores até 12, caso contrário, uma mensagem de erro deve ser apresentada.

```
Private Sub cmdFat_Click()  
    Dim num As Integer  
    Dim fat As Long  
    Dim cont As Integer  
  
    lstRel.Clear  
  
    num = CInt(txtNumero.Text)  
  
    If num <= 12 Then  
  
        fat = 1  
        cont = 1  
  
        While cont <= num  
            fat = fat * cont  
            lstRel.AddItem ("cont: " & cont & " fat: " & fat)  
            cont = cont + 1  
        Wend  
  
        lblResult.Caption = "Fatorial = " & CStr(fat)  
  
    Else  
        MsgBox "Erro: número maior que 12", vbCritical  
    End If  
End Sub
```

3) Implementar um programa que calcule o *i-ésimo* termo de uma PA. Para tanto, o usuário deve digitar o termo inicial, a razão da PA e o *i-ésimo* termo.

```
Private Sub cmdPa_Click()  
    Dim numero As Integer  
    Dim razao As Integer  
    Dim termo As Integer  
    Dim cont As Integer  
  
    numero = CInt(txtInicio.Text)  
    razao = CInt(txtRazao.Text)  
    termo = CInt(txtTermo.Text)  
  
    cont = 0  
  
    While cont <= termo  
        numero = numero + razao  
        cont = cont + 1  
    Wend  
  
    lblResult.Caption = "O termo " & termo & " é " & CStr(numero)  
  
End Sub
```

4) Implementar um programa que receba como entrada um número e descubra o número de divisores que este número possui (o programa também deve mostrar os divisores do número).

```
Private Sub cmdDivisores_Click()  
    Dim num As Integer  
    Dim divisores As Integer  
    Dim cont As Integer  
  
    lstRel.Clear  
  
    num = CInt(txtNumero.Text)  
  
    divisores = 0  
    cont = 1  
  
    While cont <= num  
        If num Mod cont = 0 Then  
            divisores = divisores + 1  
            lstRel.AddItem (cont & " é divisor de " & num)  
        End If  
        cont = cont + 1  
    Wend  
  
    lblResult.Caption = CStr(divisores) & " divisores"  
  
End Sub
```

5) Implementar um programa que verifique se um número digitado pelo usuário é primo ou não.

```
Private Sub cmdDivisores_Click()
    Dim num As Integer
    Dim divisores As Integer
    Dim cont As Integer

    lstRel.Clear

    num = CInt(txtNumero.Text)

    divisores = 0
    cont = 1

    While cont <= num
        If num Mod cont = 0 Then
            divisores = divisores + 1
            lstRel.AddItem (cont & " é divisor de " & num)
        End If
        cont = cont + 1
    Wend

    If divisores > 2 Then
        lblResult.Caption = "NÃO é um número primo"
    Else
        lblResult.Caption = "É um número primo"
    End If
End Sub
```

(**dica:** para os itens 4 e 5 utilize a função **Mod** para descobrir quando um número é divisível pelo outro)