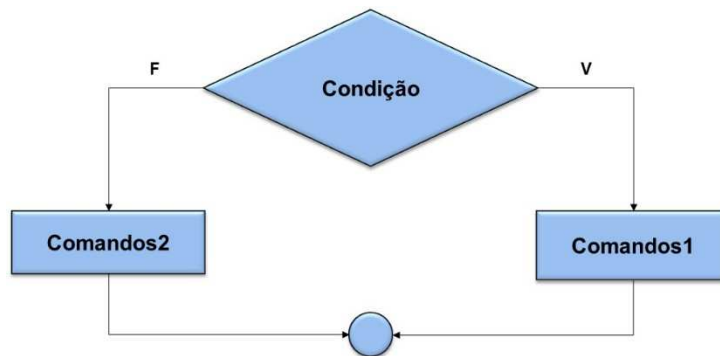
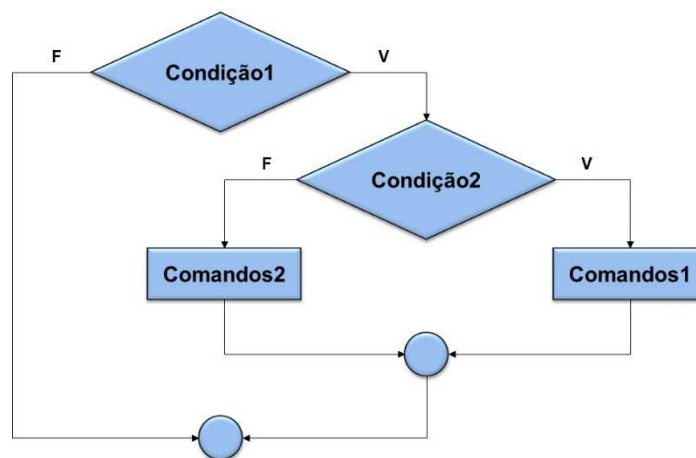


## LIÇÃO 1 - COMANDOS DE DECISÃO

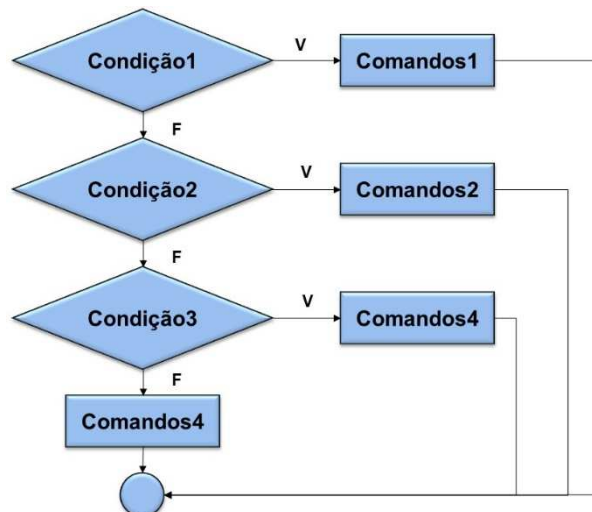
### • DECISÃO SIMPLES



### • DECISÃO ENCADEADA



### • DECISÃO MÚLTIPLA



**IF...THEN...ELSE:** Executa um bloco de instruções dependendo do valor de uma condição. As cláusulas **Else** e **ElseIf** são opcionais. Você pode ter quantas cláusulas **ElseIf** desejar em um bloco **If**, mas nenhuma pode aparecer depois de uma cláusula **Else**. As instruções do bloco **If** podem ser aninhadas, isto é, contidas uma dentro da outra.

```

If condição Then
    [Instruções]
ElseIf condição Then
    [Instruções]
Else
    [Instruções]
End If
  
```

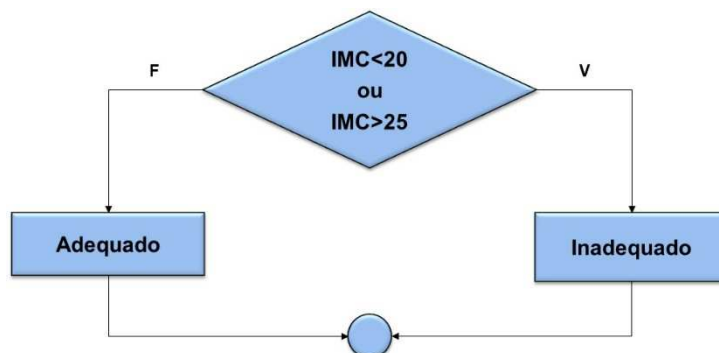
**SELECT CASE:** Executa um dos diversos grupos de instruções, dependendo do valor de uma expressão. Representa uma alternativa ao **If...Then...Else** já que permite testar várias condições.

```

Select Case expressão
    Case expressão1
        [Instruções]
    Case expressão2
        [Instruções]
    Case Else
        [Instruções]
End Select
  
```

Fonte: Ajuda do aplicativo

1. Abra a pasta **VBA2** e habilite seu conteúdo.
2. Na janela do editor, expanda **Módulos** e abra a janela de código do módulo **Exemplo1**.
3. Complete as linhas de código do procedimento **Exemplo1\_1** listado abaixo.

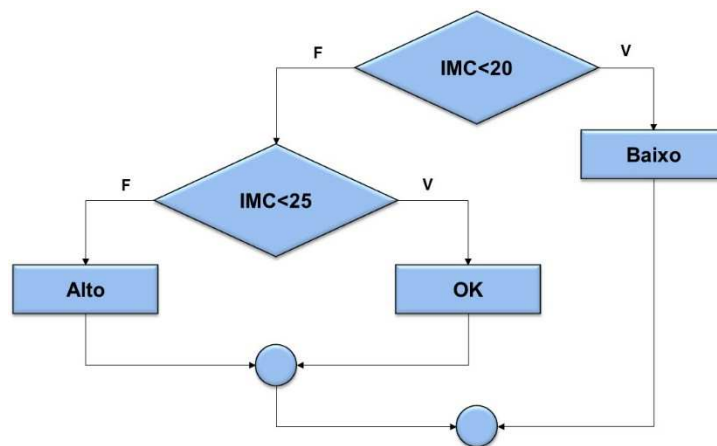


```

Sub Exemplo1_1()
'Declarar variável IMC
Dim IMC As Single
'Atribuir à IMC o valor digitado na InputBox
IMC = InputBox("Digite seu IMC")
'Se <20 ou >25 emitir msg Inadequado
If IMC < 20 Or IMC > 25 Then
    MsgBox "Inadequado"
'Senão(>=20 e <=25) emitir msg Adequado
Else
    MsgBox "Adequado"
'Fim SE <20 ou >25
End If
End Sub

```

4. Pressione **F5** para executar o procedimento e testar o novo código.
5. Complete as linhas de código do procedimento **Exemplo1\_2** listado abaixo.

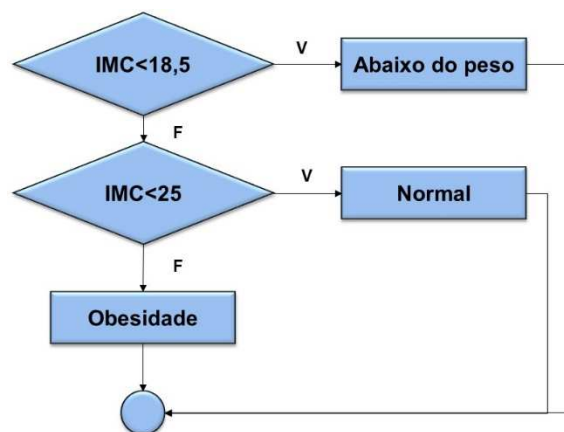


```

Sub Exemplo1_2()
'Declarar variável IMC
Dim IMC As Single
'Atribuir à IMC o valor digitado na InputBox
IMC = InputBox("Digite seu IMC")
'Se <20 emitir msg Baixo
If IMC < 20 Then
    MsgBox "Baixo"
'Senão(>=20)
Else
'Se <25 emitir msg OK
    If IMC < 25 Then
        MsgBox "OK"
'Senão(>=25) emitir msg Alto
    Else
        MsgBox "Alto"
'Fim SE < 25
    End If
'Fim SE <20
End If
End Sub

```

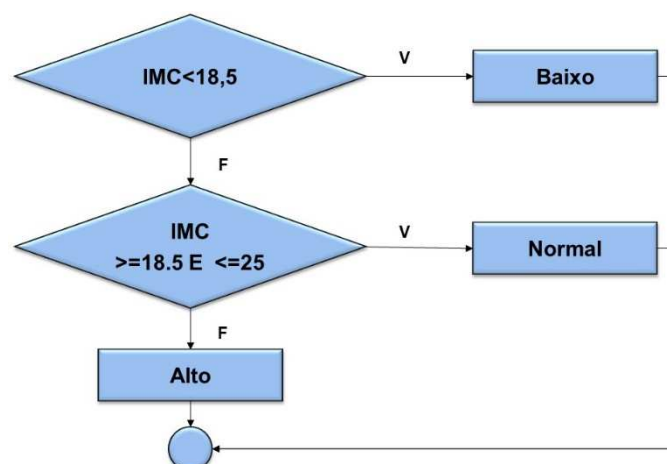
6. Pressione **F5** para executar o procedimento e testar o novo código.
7. Complete as linhas de código do procedimento **Exemplo1\_3** listado abaixo.



```

Sub Exemplo1_3()
'Declarar variável IMC
Dim IMC As Single
'Atribuir à IMC o valor digitado na InputBox
IMC = InputBox("Digite seu IMC")
'Se < 18.5 emitir msg Abaixo do peso
If IMC < 18.5 Then
    MsgBox "Abaixo do peso"
'Se >=18.5 e <25 emitir msg Normal
Elseif IMC < 25 Then
    MsgBox "Normal"
'Senão(>=25) emitir msg Obesidade
Else
    MsgBox "Obesidade"
'Fim SE <18.5
End If
End Sub
  
```

8. Pressione **F5** para executar o procedimento e testar o novo código.
9. Abra o código do módulo **Exemplo2**.
10. Complete as linhas de código do procedimento **Exemplo2\_1** listado abaixo.

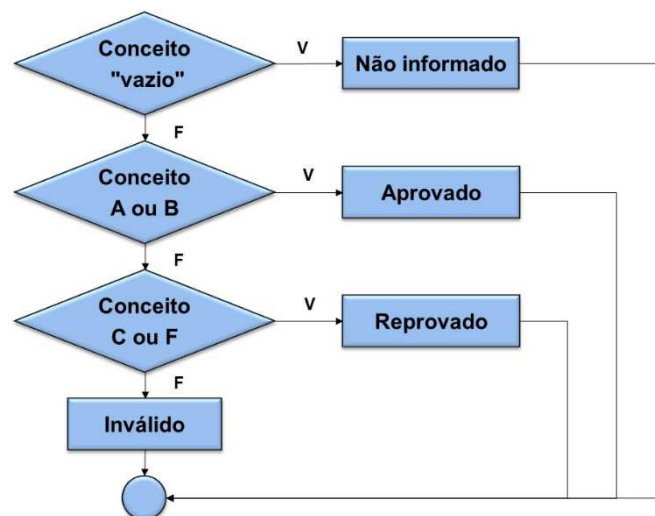


```

Sub Exemplo2_1()
'Declaração da variável IMC
Dim IMC As Single
'Atribuir à IMC o valor da InputBox
IMC = InputBox("Digite seu IMC")
'Iniciar Select Case IMC
Select Case IMC
'Se <18.5 emitir msg Baixo
    Case Is < 18.5
        MsgBox "Baixo"
'Se >=18.5 e <=25 emitir msg Normal
    Case 18.5 To 25
        MsgBox "Normal"
'Se >25 emitir msg Alto
    Case Else
        MsgBox "Alto"
'Fim Select Case IMC
End Select
End Sub

```

11. Pressione **F5** para executar o procedimento e testar o novo código.
12. Complete as linhas de código do procedimento **Exemplo2\_2** listado abaixo.



```

Sub Exemplo2_2()
'Declaração da variável Conceito
Dim Conceito As String
'Atribuir à Conceito o valor da InputBox
Conceito = InputBox ("Informe o conceito do aluno: ")
'Iniciar Select Case Conceito
Select Case Conceito
'Se não for digitado nenhum conceito
'Exibir msg "Não informado"
    Case Empty
        MsgBox "Não informado"
'Se digitado A ou B exibir msg Aprovado
    Case "A", "a", "B", "b "
        MsgBox "Aprovado"

```

```
'Se digitado C ou F exibir msg Reprovado
```

```
Case "C", "c", "F", "f"
```

```
    MsgBox "Reprovado"
```

```
'Se digitado outro valor exibir msg Inválido
```

```
Case Else
```

```
    MsgBox "Inválido"
```

```
'Fim Select Case Conceito
```

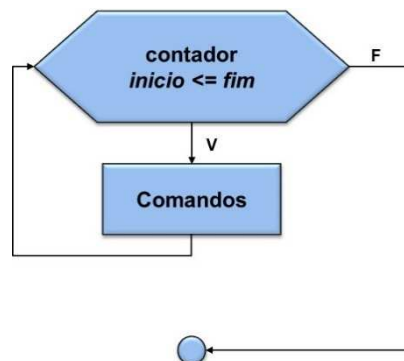
```
End Select
```

```
End Sub
```

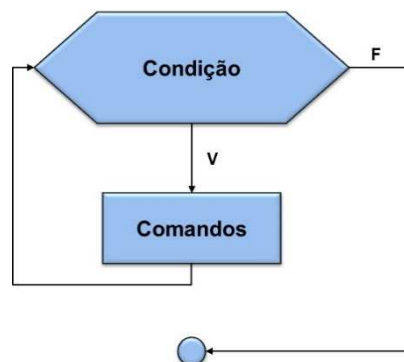
13. Pressione **F5** para executar o procedimento e testar o novo código.

## LIÇÃO 2 - COMANDOS DE REPETIÇÃO

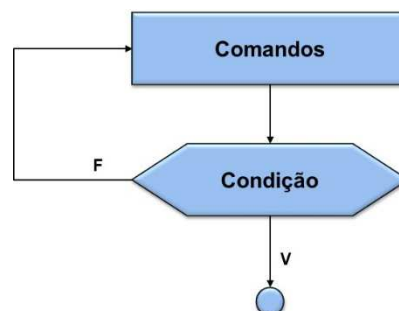
### • REPETIÇÃO COM CONTADOR



### • REPETIÇÃO COM PRÉCONDIÇÃO



### • REPETIÇÃO COM PÓSCONDIÇÃO



**FOR NEXT:** Repete um bloco de instruções um determinado número de vezes definido por uma variável contadora. A instrução **For** especifica o contador e seus valores inicial e final. A instrução **Next** incrementa o contador em uma unidade. Utilizando **Step**, podemos aumentar ou diminuir o contador pelo valor especificado. Para diminuir o contador lembre-se de definir um valor inicial maior que o valor final e utilizar um valor negativo em **Step**. Para sair de uma instrução **For...Next** antes que o contador atinja o seu valor final devemos utilizar a instrução **Exit for**.

**For** contador = início To fim [Step passo]

[Instruções]

[Exit For]

[Instruções]

**DO WHILE:** Permite realizar um bloco de instruções enquanto uma condição for verdadeira (**While**). A condição pode ser verificada antes de entrar no loop (**Do While...Loop**) ou depois de executar o loop pelo menos uma vez (**Do...Loop While**). Para sair de uma instrução **Do...Loop** podemos utilizar a instrução **Exit Do**.

**Do While** condição

[Instruções]

[Exit Do]

**Loop**

**DO UNTIL:** Permite realizar um bloco de instruções até que uma condição se torne verdadeira (**Until**). A condição pode ser verificada antes de entrar no loop (**Do Until...Loop**) ou depois de executar o loop pelo menos uma vez (**Do...Loop Until**). Para sair de uma instrução **Do...Loop** podemos utilizar a instrução **Exit Do**.

**Do Until** condição

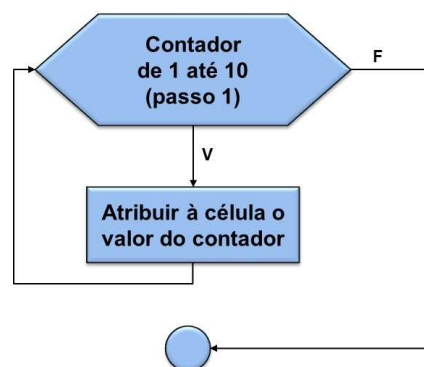
[Instruções]

[Exit Do]

**Loop**

Fonte: Ajuda do aplicativo

1. Abra o código do módulo **Exemplo3**.
2. Complete as linhas de código do procedimento **Exemplo3\_1** listado abaixo.



Sub Exemplo3\_1()

'Declaração da variável contador

Dim contador As Integer

'Para contador variando de 1 a 10 com passo 1 unidade

For contador = 1 To 10

'Atribuir à célula o valor de contador

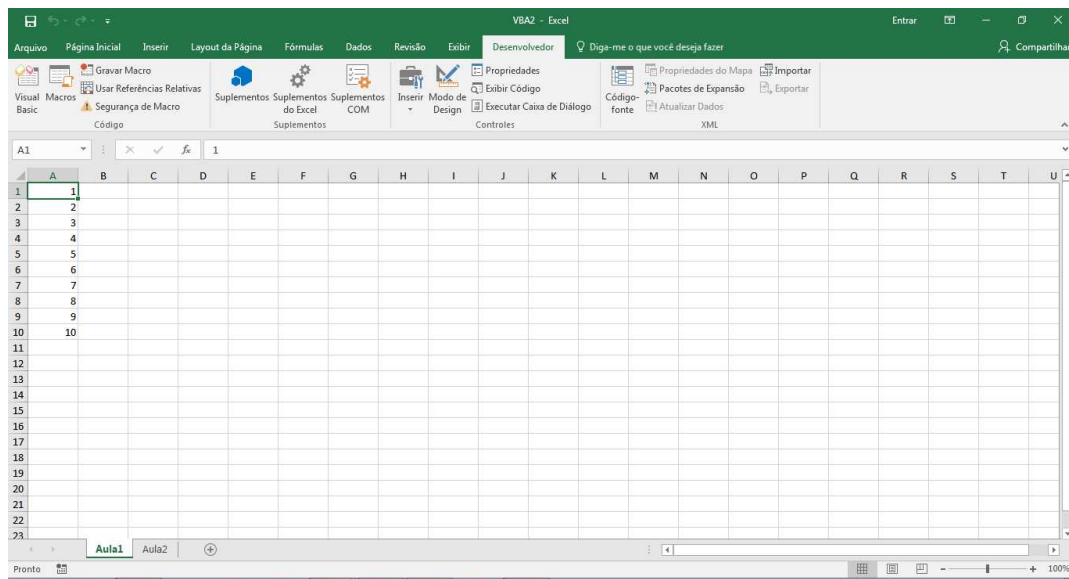
Cells(contador, 1).Value = contador

'Contador (+1)

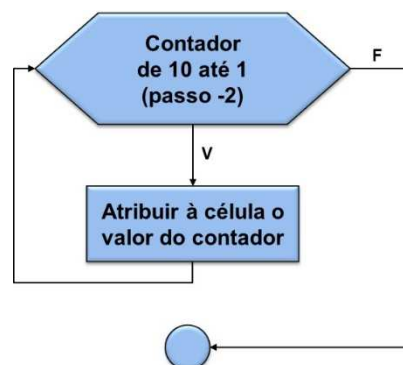
Next

End Sub

3. Pressione **F5** para executar o procedimento e testar o novo código.
4. Pressione **ALT+F11** para exibir a janela do **Excel** e observe o resultado na planilha ativa (**Aula1**).



5. Pressione **ALT+F11** para voltar para a janela do editor.
6. Complete as linhas de código do procedimento **Exemplo3\_2** listado abaixo.



Sub Exemplo3\_2()

'Declaração da variável contador

Dim contador As Integer

'Para contador variando de 10 a 1 com passo -2 unidades

For contador = 10 To 1 Step -2

'Atribuir à célula o valor do contador

Cells(contador, 2).Value = contador

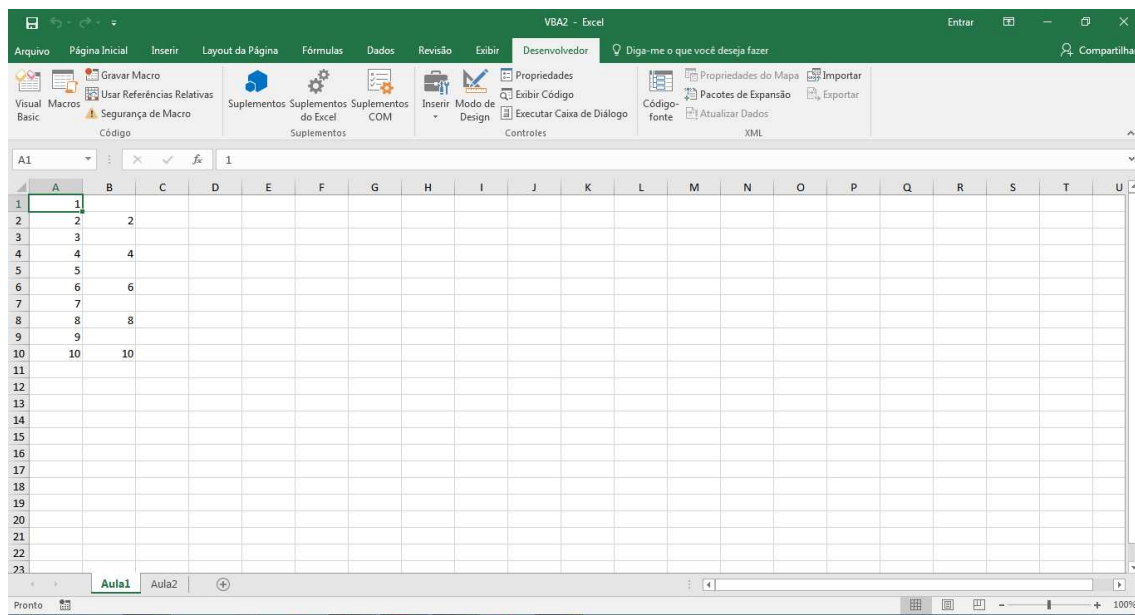
'Contador (-2)

Next

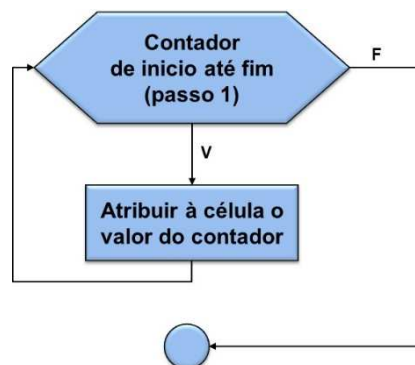
End Sub



7. Pressione **F5** para executar o procedimento e testar o novo código.
8. Pressione **ALT+F11** para exibir a janela do **Excel** e observe o resultado na planilha ativa (**Aula1**).



9. Pressione **ALT+F11** para voltar para a janela do editor.
10. Complete as linhas de código do procedimento **Exemplo3\_3** listado abaixo.



**Sub Exemplo3\_3()**

**'Declaração das variáveis contador, inicio e fim**

**Dim contador As Integer, inicio As Integer, fim As Integer**

**'Atribuir às variáveis inicio e fim os valores de InputBox**

**inicio = InputBox("Digite número inicial")**

**fim = InputBox("Digite número final")**

**'Para contador variando de inicio a fim com passo 1 unidade**

**For contador = inicio To fim**

**'Atribuir à célula o valor do contador**

**Cells(contador, 3).Value = Contador**

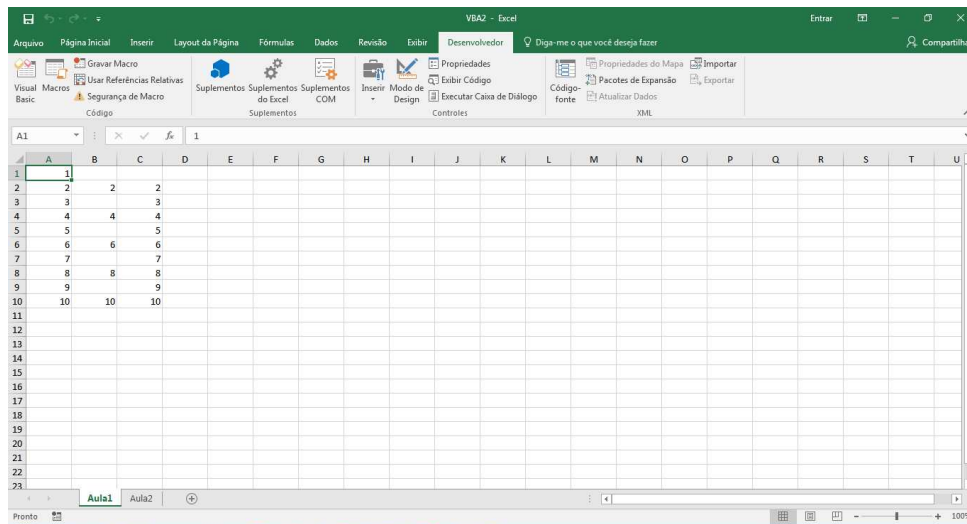
**'Contador (+1)**

**Next**

**End Sub**

11. Pressione **F5** para executar o procedimento e digite os valores **2** e **10** para **inicio** e **fim** respectivamente.

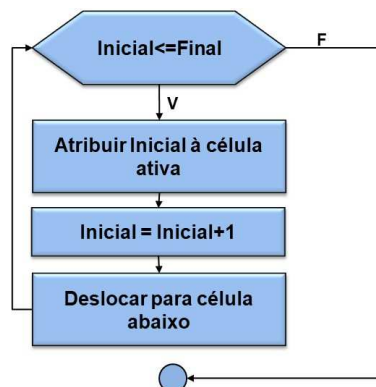
12. Pressione **ALT+F11** para exibir a janela do **Excel** e observe o resultado na planilha ativa (**Aula1**).



13. Pressione **ALT+F11** para voltar para a janela do editor.

14. Abra a janela código do módulo **Exemplo4**.

15. Complete as linhas de código do procedimento **Exemplo4\_1** listado abaixo.

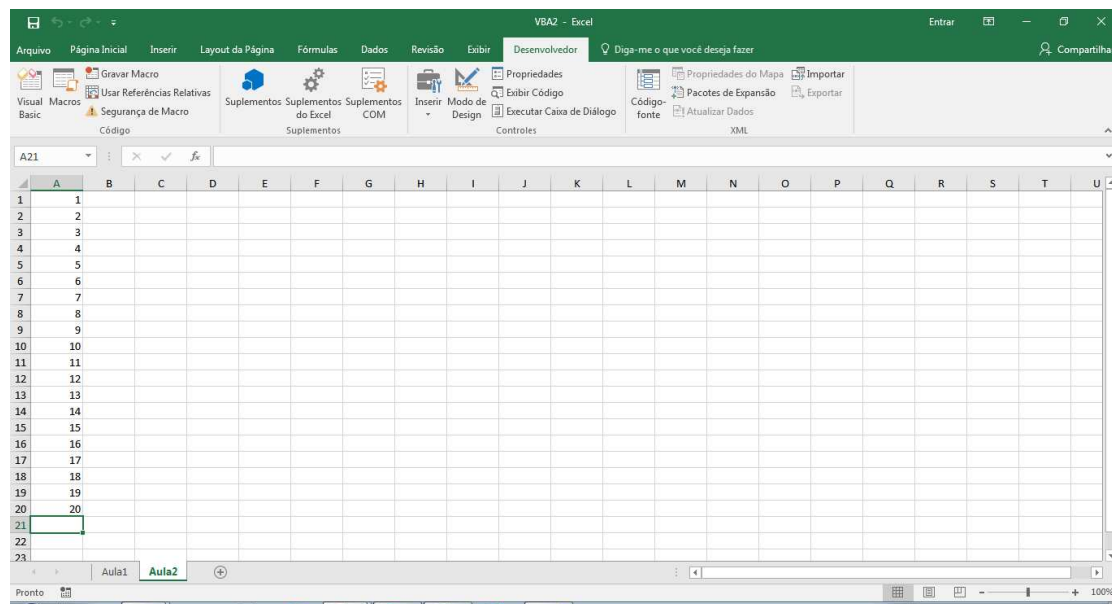


```

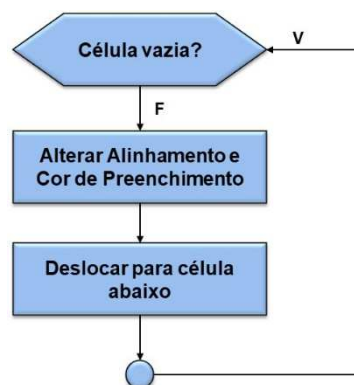
Sub Exemplo4_1()
'Declaração das variáveis Inicial e Final
Dim Inicial As Integer, Final As Integer
'Atribuir os valores às variáveis Inicial e Final
Inicial = 1
Final = 20
'Selecionar a planilha Aula2
Sheets("Aula2").Select
'Selecionar a célula A1
Range("A1").Select
'Início laço Do While
Do While Inicial <= Final
'Atribui o valor da variável Inicial à célula ativa
    ActiveCell.Value = Inicial
'Adiciona uma unidade à variável Inicial
    Inicial = Inicial + 1
'Selecionar a célula da linha abaixo
    ActiveCell.Offset(1, 0).Select
'Fim do laço Do While
Loop
End Sub
  
```

16. Pressione **F5** para executar o procedimento.

17. Pressione **ALT+F11** para exibir a janela do **Excel** e observe o resultado na planilha **Aula2**.



18. Volte para o editor e complete as linhas de código do procedimento **Exemplo4\_2** listado abaixo.

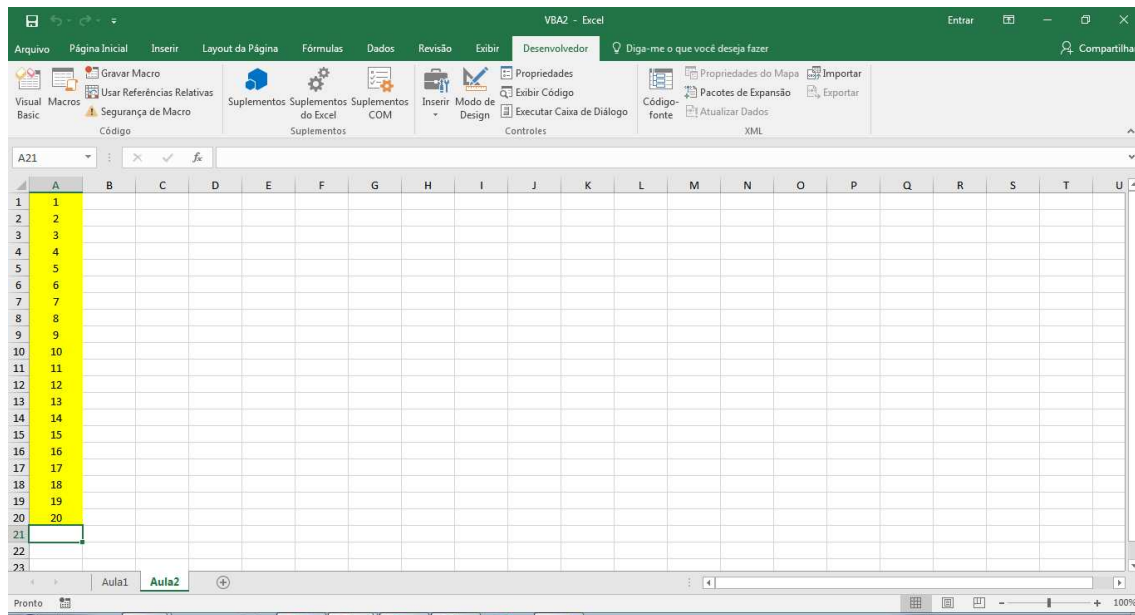


```

Sub Exemplo4_2()
'Selecionar a planilha Aula2
Sheets("Aula2").Select
'Selecionar a célula A1
Range("A1").Select
'Início do laço Do Until
Do Until IsEmpty(ActiveCell)
'Ajustar o alinhamento horizontal da célula para centralizado
ActiveCell.HorizontalAlignment = xlCenter
'Ajustar o sombreamento da célula ativa para amarelo
ActiveCell.Interior.Color = vbYellow
'Ativar a célula da linha abaixo
ActiveCell.Offset(1, 0).Activate
'Fim do laço Do Until
Loop
End Sub
  
```

19. Pressione **F5** para executar o procedimento.

20. Pressione **ALT+F11** para exibir a janela do **Excel** e observe o resultado na planilha **Aula2**.



**WITH...END WITH:** Executa uma série de instruções que referenciam repetidamente um único objeto ou estrutura de modo que as instruções possam usar uma sintaxe simplificada para acessar membros do objeto ou estrutura. Usando **With...End With**, você pode executar uma série de instruções em um objeto especificado sem especificar o nome do objeto várias vezes.

**With** *objeto*

[Instruções]

**End With**

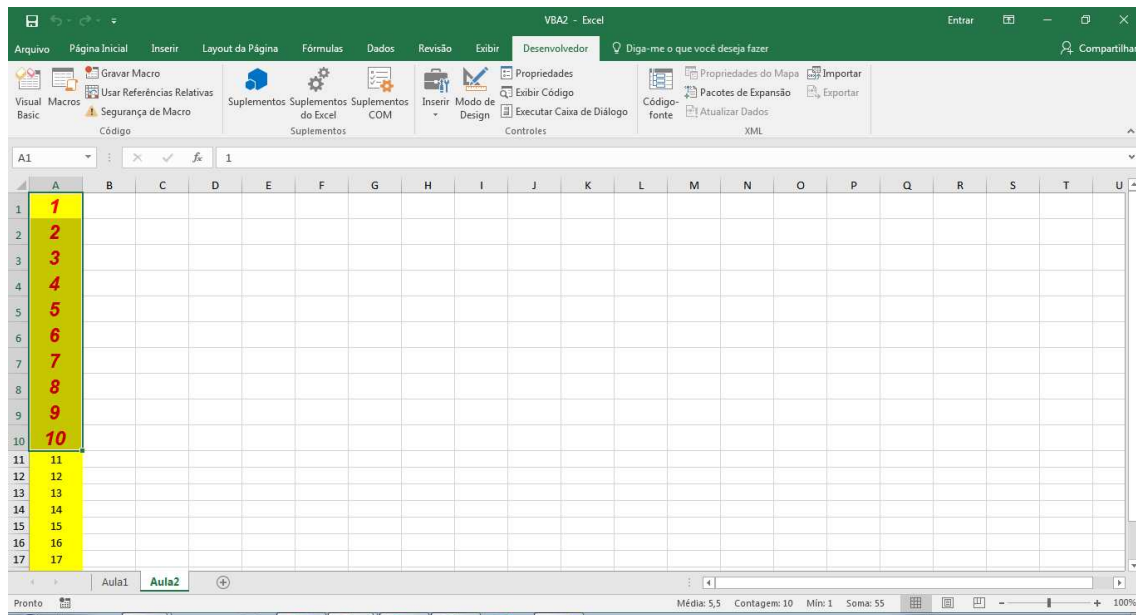
**Fonte:** Ajuda do aplicativo

1. Volte para o editor e complete as linhas de código do procedimento **Exemplo4\_3** listado abaixo.

```
Sub Exemplo4_3()
'Selecionar o intervalo A1:A10 da planilha Aula2
Sheets("Aula2").Range("A1:A10").Select
'Início With (para a fonte do intervalo)
With Selection.Font
'Formatar negrito, itálico, vermelho, 18 e Arial
.Bold = True
.Italic = True
.Color = RGB(255, 0, 0)
.Size = 18
.Name = "Arial"
'Fim With
End With
End Sub
```

2. Pressione **F5** para executar o procedimento.
3. Pressione **ALT+F11** para exibir a janela do **Excel** e observe o resultado na planilha **Aula2**.

## 4. Salve as alterações e feche o arquivo.



### CONTEÚDO COMPLEMENTAR (EXTRAÍDO DA AJUDA DO APLICATIVO)

Função **IsEmpty(expressão)**: retorna um valor booleano que indica se uma expressão foi inicializada.

Função **RGB(verm,verde,azul)**: representa um valor de cor RGB.

Método **Activate**: ativa o objeto.

Método **Select**: seleciona o objeto.

Objeto **Range**: representa uma célula, uma linha, uma coluna, uma seleção de células contendo um ou mais blocos contíguos de células ou um intervalo 3D.

Objeto **Sheets**: Representa uma coleção de todas as planilhas na pasta de trabalho ativa ou especificada. Use **Sheets( índice)**, onde índice é o nome ou o número da planilha, para retornar um único objeto **Chart** ou **Worksheet**.

Propriedade **ActiveCell**: retorna um objeto **Range** que representa a célula ativa na janela ativa (a janela visível) ou na janela especificada.

Propriedade **Cells**: retorna um objeto **Range** que representa as células no intervalo especificado: **Cells(linha,coluna)**.

Propriedade **Font**: representa a fonte do objeto especificado.

Propriedade **HorizontalAlignment**: representa o alinhamento horizontal para o objeto especificado.

Propriedade **Offset(linha,coluna)**: retorna um objeto **Range** que representa um intervalo que é deslocado do intervalo especificado.

Propriedade **Value**: representa o valor do intervalo especificado.

Propriedades **Interior.Color**: representa a cor de sombreamento da célula ou a cor de preenchimento do objeto de desenho.

**Fonte: Ajuda do aplicativo**