

## 2. Estructuras condicionales

### Fundamentos de Informática

Dpto. Lenguajes y Sistemas Informáticos

Curso 2012 / 2013

### Índice

Estructuras condicionales

1. Ej07: Cond. simple
2. Ej08: Cond. doble
3. Ej09: Cond. anidada
4. Ej10: Cond. cascada
5. Recomendaciones
6. Resumen
7. Operadores relacionales
8. Operadores booleanos o lógicos

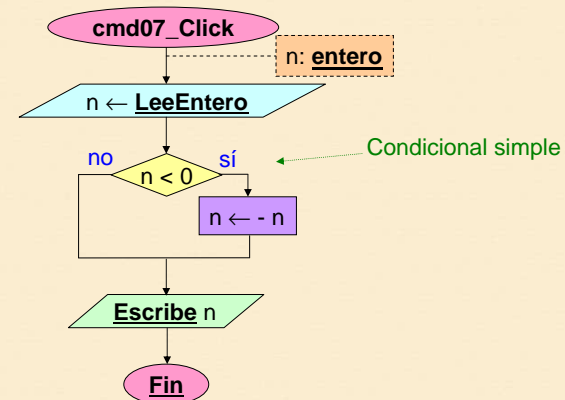
Condicionales | Ej07: Cond. simple

### Ejemplo 07

- **Título:**
  - Condicional simple
- **Nombre**
  - cmd07\_Click
- **Descripción**
  - Leer una variable entera, calcular su **valor absoluto** (en la misma variable) y mostrar el resultado
- **Observaciones**
  - Condicional simple

Conditionals | Ej07: Cond. simple

### Ej07: Diagrama de Flujo

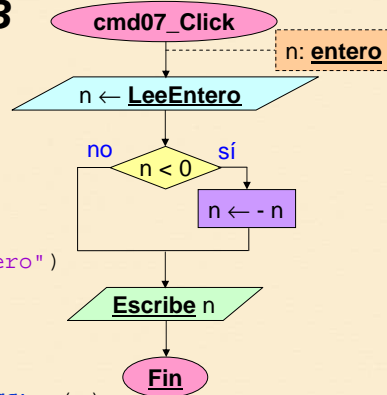


## Ej07: Codificación VB

```

Sub cmd07_Click()
    Dim s As String
    Dim n As Integer

    s = InputBox ("Introduce número")
    n = CInt (s)
    If n < 0 Then
        n = -n
    End If
    MsgBox "Valor absoluto: " & CStr (n)
End Sub
    
```

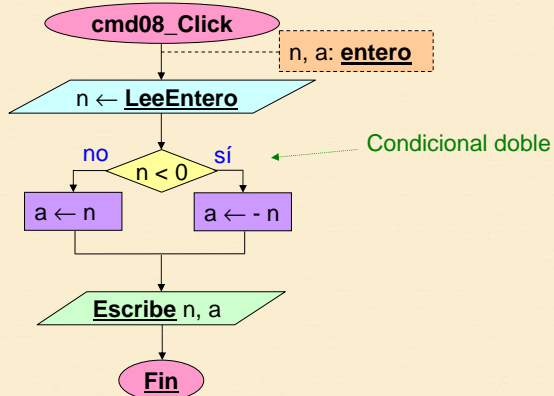


## 2. Ejemplo 08

- **Título:**
  - Condicional doble
- **Nombre**
  - cmd08\_Click
- **Descripción**
  - Leer una variable entera, calcular su valor absoluto en otra variable, y mostrar el resultado
- **Observaciones**
  - Condicional doble



## Ej08: Diagrama de Flujo

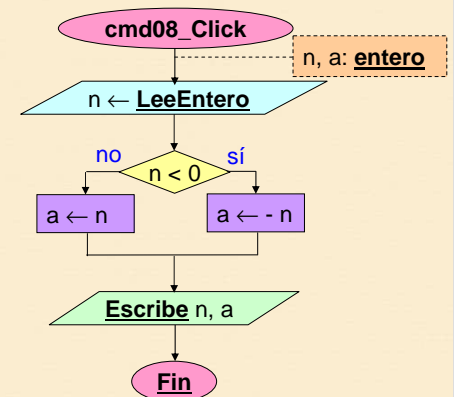


## Ej08: Codificación VB

```

Sub cmd08_Click()
    Dim s As String
    Dim n As Integer
    Dim a As Integer

    s = InputBox ("Número: ")
    n = CInt (s)
    If n < 0 Then
        a = -n
    Else
        a = n
    End If
    MsgBox "Valor absoluto de " & CStr (n) & _
        " es " & CStr (a)
End Sub
    
```

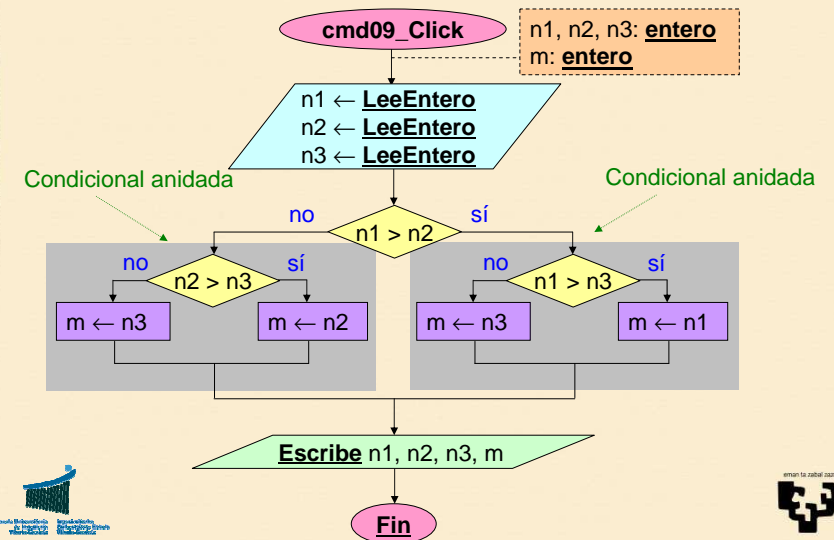


### 3. Ejemplo 09

- **Título:**
  - Condicional anidada
- **Nombre**
  - cmd09\_Click
- **Descripción**
  - Leer tres variables enteras, calcular cuál es la mayor de ellas y mostrar el resultado
- **Observaciones**
  - Condicional anidada (condicional dentro de una condicional)



### Ej09: Diagrama de Flujo



### Ej09: Codificación VB (I)

```

Sub cmd09_Click()
    Dim s As String
    Dim n1 As Integer, n2 As Integer, n3 As Integer
    Dim m As Integer
    s = InputBox ("Introduce primer número: ")
    n1 = CInt (s)
    s = InputBox ("Introduce segundo número: ")
    n2 = CInt (s)
    s = InputBox ("Introduce tercer número: ")
    n3 = CInt (s)
    ...
    MsgBox "El mayor de " & CStr (n1) & ", " & _
           CStr (n2) & " y " & CStr (n3) & _
           " es: " & CStr (m)
End Sub
    
```



### Ej09: Codificación VB (II)

```

...
n3 = CInt (s)
If n1 > n2 Then
    If n1 > n3 Then
        m = n1
    Else
        m = n3
    End If
Else
    If n2 > n3 Then
        m = n2
    Else
        m = n3
    End If
End If
MsgBox "El mayor de " & CStr (n1) & ", " & _
...
    
```

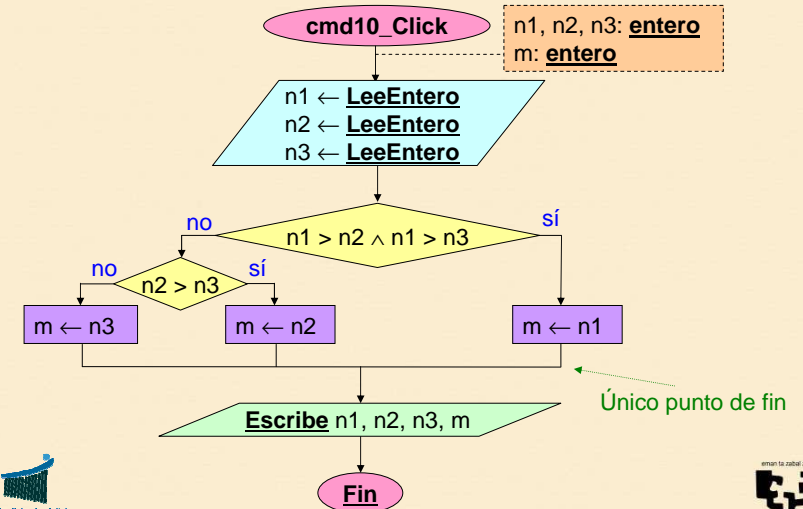


## 4. Ejemplo 10

- **Título:**
  - Condicional en cascada
- **Nombre**
  - cmd10\_Click
- **Descripción** (Mismo problema que el ejemplo anterior)
  - Leer tres variables enteras, calcular cuál es la mayor de ellas y mostrar el resultado
  - Variante distinta de la solución
- **Observaciones**
  - Condicional en cascada
  - Tras la alternativa Else inmediatamente viene una nueva condición, convirtiéndose en ElseIf



## Ej10: Diagrama de Flujo



## Ej10: Codificación VB (I)

```

Sub cmd10_Click()
    Dim s As String
    Dim n1 As Integer, n2 As Integer, n3 As Integer
    Dim m As Integer
    s = InputBox "Introduce primer número: "
    n1 = CInt (s)
    s = InputBox "Introduce segundo número: "
    n2 = CInt (s)
    s = InputBox "Introduce tercer número: "
    n3 = CInt (s)
    ...
    MsgBox "El mayor de " & CStr (n1) & ", " & _
           CStr (n2) & " y " & CStr (n3) & _
           " es: " & CStr (m)
End Sub
    
```



## Ej10: Versión anidada (II)

```

...
n3 = CInt (s)
If n1 > n2 And n1 > n3 Then
    m = n1
Else
    If n2 > n3 Then
        m = n2
    Else
        m = n3
    End If
End If
MsgBox "El mayor de " & CStr (n1) & ", " & _
...
    
```

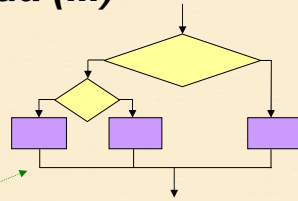


### Ej10: Versión en cascada (III)

```

...
n3 = CInt (s)
If n1 > n2 And n1 > n3 Then
    m = n1
ElseIf n2 > n3 Then
    m = n2
Else
    m = n3
End If
MsgBox "El mayor de " & CStr (n1) & ", " & _
...
    
```

Único punto de fin



### 5. Recomendaciones (I)

- El cuerpo de las instrucciones condicionales normalmente irá sangrado dos espacios en cada anidamiento, a añadir al cuerpo del subprograma
- Es por legibilidad, ya que Visual Basic lo entenderá de igual manera
- **Ejemplo:**

```

Sub ejemplo ()
    . . .
    If a > b Then
        t = a
        a = b
        b = t
    End If
End Sub
    
```



### 5. Recomendaciones (II)

- Para facilitar la programación es conveniente identificar conjuntos disjuntos y la condición que las distingue
- Es mejor no volver a verificar condiciones excluidas
- **Ejemplo:**

```

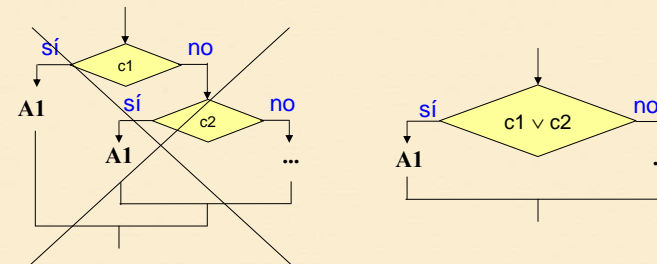
If nota < 5 Then
    cal = "suspenseo"
ElseIf nota < 7 Then
    cal = "aprobado"
ElseIf nota < 9 Then
    cal = "notable"
Else
    cal = "sobresaliente"
End If
    
```

No volvemos a mirar si es mayor o igual a 5



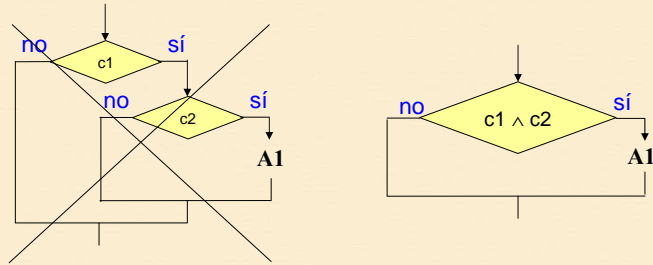
### 5. Recomendaciones (III)

- Cuando hay dos condiciones a las que queremos asociar la misma acción se agruparán en una sola



## 5. Recomendaciones (IV)

- Cuando han de cumplirse dos condiciones simultaneas no pondremos dos instrucciones condicionales sino una sola con la conjunción de las condiciones



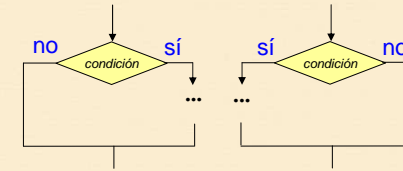
## 6. Resumen (I)

- **Condicionales simple**

If condición Then

...

End If



- **Condicionales doble**

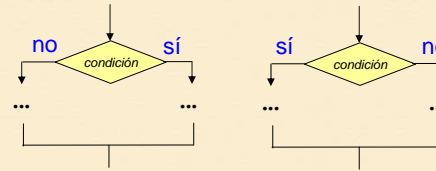
If condición Then

...

Else

...

End If



## 6. Resumen (II)

- **Condicionales anidada**

If condición Then

...

Else

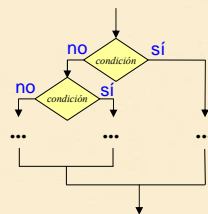
If condición Then

...

Else

End If

End If



- **Condicionales en cascada**

If condición Then

...

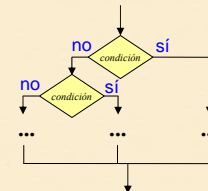
ElseIf condición Then

...

Else

...

End If



## 6. Resumen (III)

- **Condicionales simple incorrecta**

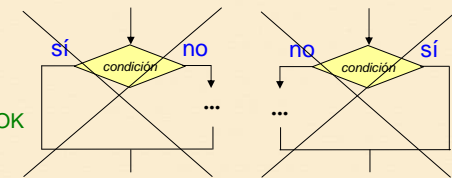
If condición Then

Else

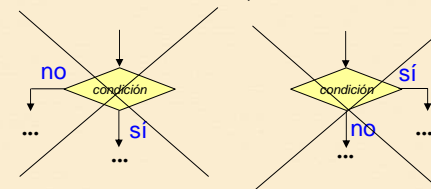
...

End If

- Negar la condición y OK



- **Representaciones incorrectas** (confusión con bucles)





## 7. Operadores relacionales

- Comparan valores y tienen resultado lógico **True** ó **False**
- Los operadores relacionales son 6

DdF	VB	Descripción
>	>	Mayor
≥	>=	Mayor o igual
=	=	Igual
≠	<>	Distinto
<	<	Menor
≤	<=	Menor o igual

- Hay que tener cuidado porque no se pueden formar expresiones matemáticas de acotamiento



## 7. Operaciones booleanas o lógicas

- Relacionan expresiones lógicas
- El resultado es de tipo lógico (**Boolean**): **True** ó **False**
- Los operandos lógicos son 4:

DdF	VB	Descripción
¬	<b>Not</b>	Negación, no
∧	<b>And</b>	Conjunción, y
∨	<b>Or</b>	Disyunción, o
⊕	<b>Xor</b>	Disyunción exclusiva, o exclusiva



### • Operadores booleanos o lógicos

- El resultado se expresa mediante **tablas de verdad**

DdF	VB	Descripción
¬	<b>Not</b>	Cambia de cierto a falso y viceversa
∧	<b>And</b>	Cierto sólo si ambos operandos son ciertos
∨	<b>Or</b>	Cierto si algún operando o ambos son ciertos
⊕	<b>Xor</b>	Cierto si los operandos son distintos

		Negación	Conjunción	Disyunción	Disyunción exclusiva
		¬a	a ∧ b	a ∨ b	a ⊕ b
a	b	<b>Not</b> a	a <b>And</b> b	a <b>Or</b> b	a <b>Xor</b> b
Falso	Falso	Cierto	Falso	Falso	Falso
Falso	Cierto	Cierto	Falso	Cierto	Cierto
Cierto	Falso	Falso	Falso	Cierto	Cierto
Cierto	Cierto	Falso	Cierto	Cierto	Falso



### • Leyes de De Morgan (*Augustus De Morgan 1806-1871*)

- La **negación de la disyunción** es equivalente a la conjunción de la negaciones (cambia ∨ por ∧)

$$\neg (a \vee b) \equiv \neg a \wedge \neg b$$

$$\text{Not } (a \text{ Or } b) \equiv \text{Not } a \text{ And Not } b$$

Ejemplo:

$$\text{DdF: } \neg (a > 0 \wedge b > 0) \equiv a \leq 0 \vee b \leq 0$$

$$\text{VB: } \text{Not } (a > 0 \text{ And } b > 0) \equiv a \leq 0 \text{ Or } b \leq 0$$

- La **negación de la conjunción** es equivalente a la disyunción de las negaciones (cambia ∧ por ∨)

$$\neg (a \wedge b) \equiv \neg a \vee \neg b$$

$$\text{Not } (a \text{ And } b) \equiv \text{Not } a \text{ Or Not } b$$

Ejemplo:

$$\text{DdF: } \neg (a \geq 0 \wedge b > 0) \equiv a < 0 \vee b \leq 0$$

$$\text{VB: } \text{Not } (a \geq 0 \text{ And } b > 0) \equiv a < 0 \text{ Or } b \leq 0$$

