

5. Subprogramas

Fundamentos de Informática

Dpto. Lenguajes y Sistemas Informáticos

Curso 2012 / 2013



Escuela Universitaria
de Ingeniería
Vitoria-Gasteiz

Ingeniaritzako
Unibertsitate Eskola
Vitoria-Gasteiz



Universidad
del País Vasco

Euskal Herriko
Unibertsitatea

Índice

Subprogramas

1. Cálculo de la función Coseno
2. Suma
3. Ecuación de 2º grado

1. Cálculo de la función Coseno

- **Título**

- Coseno

- **Nombre**

- PrgCoseno

$$\cos(x) = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$$

- **Descripción**

- Programa VB que lee un ángulo en radianes y calcule su coseno, utilizando el desarrollo de Taylor con un error inferior a 0,000001.

- **Observaciones**

- Descomposición en funciones
- Diseño con y sin funciones
- Diseño descendente - implementación ascendente

1.1 Análisis Coseno

$$\cos(x) = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$$

$$y = t_0 + t_1 + t_2 + t_3 + \dots + t_{\infty}$$

$$y = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots$$

Cada t_i reduce el error en $|t_i|$

$$y = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$$

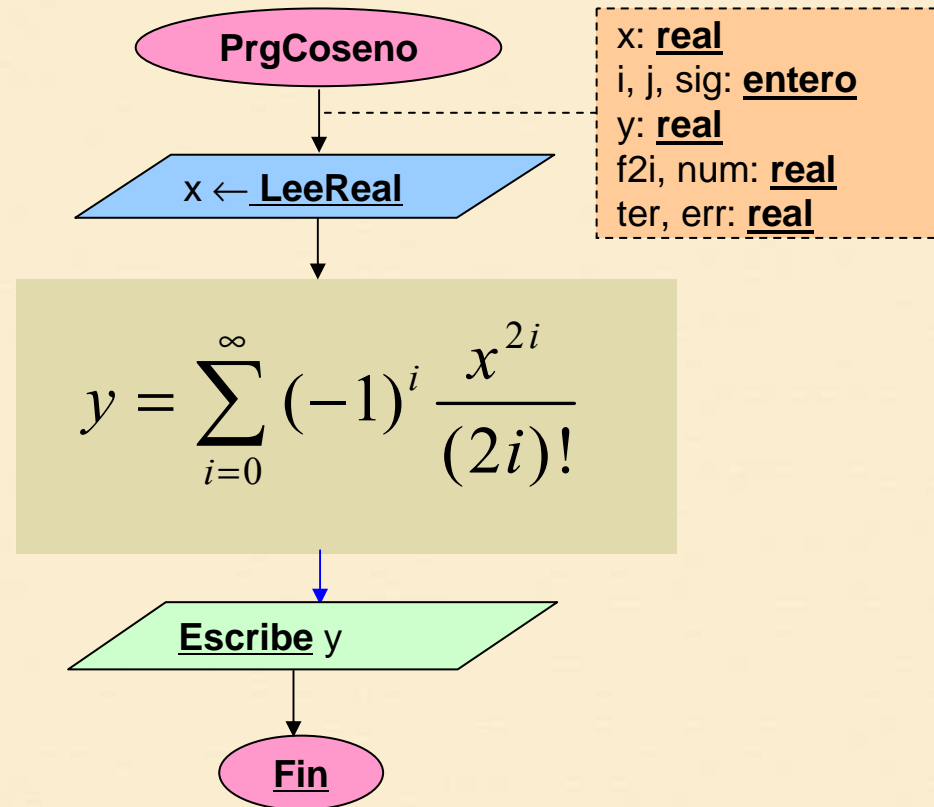
$$t_0 = -1^0 \cdot \frac{x^{2 \cdot 0}}{(2 \cdot 0)!} = 1$$

$$t_1 = -1^1 \cdot \frac{x^{2 \cdot 1}}{(2 \cdot 1)!} = -\frac{x^2}{2}$$

$$t_2 = -1^2 \cdot \frac{x^{2 \cdot 2}}{(2 \cdot 2)!} = \frac{x^4}{24}$$

$$t_3 = -1^3 \cdot \frac{x^{2 \cdot 3}}{(2 \cdot 3)!} = -\frac{x^6}{720}$$

1.2 Programa Coseno sin subprogramas

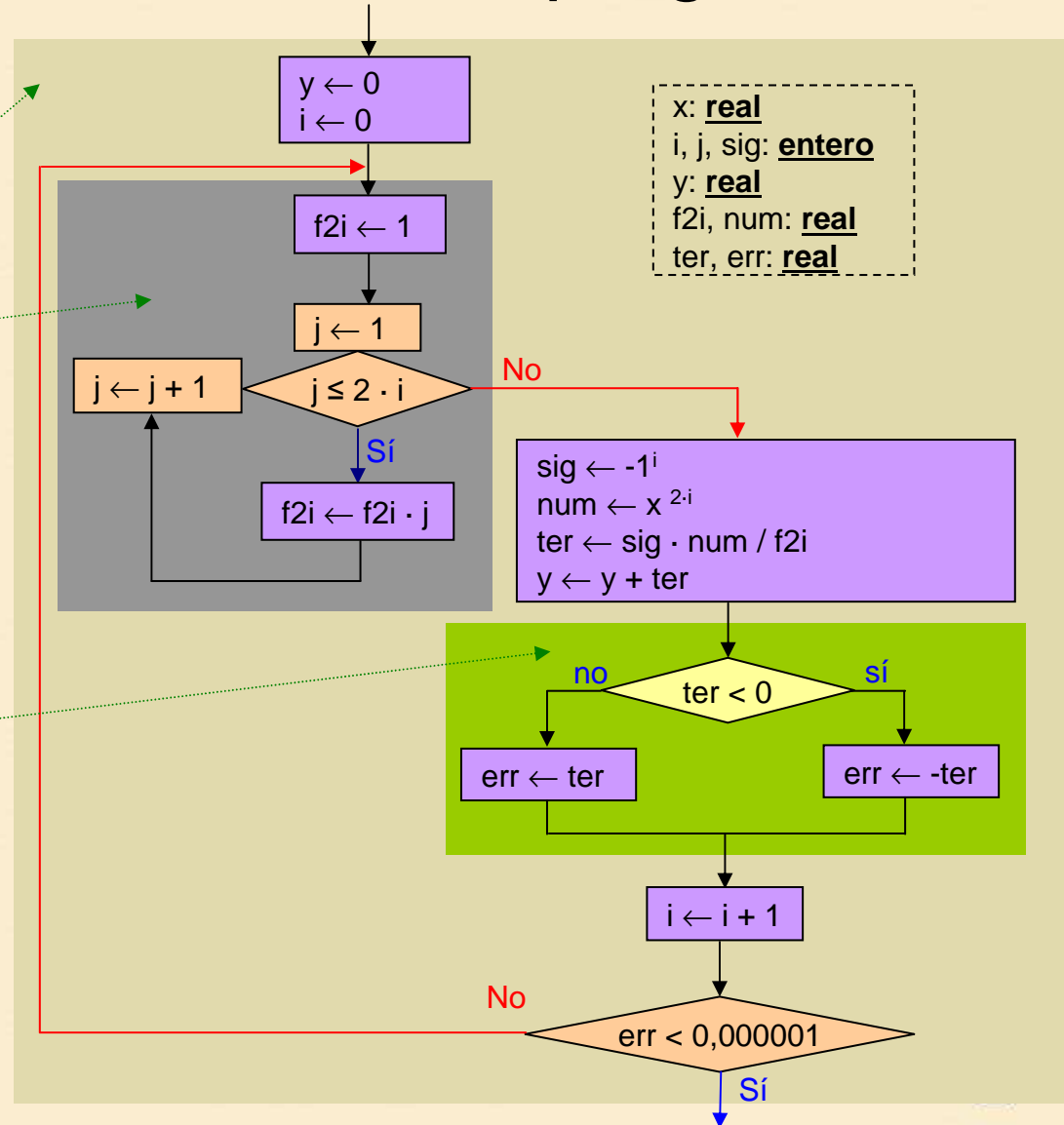


1.2 Programa Coseno sin subprogramas

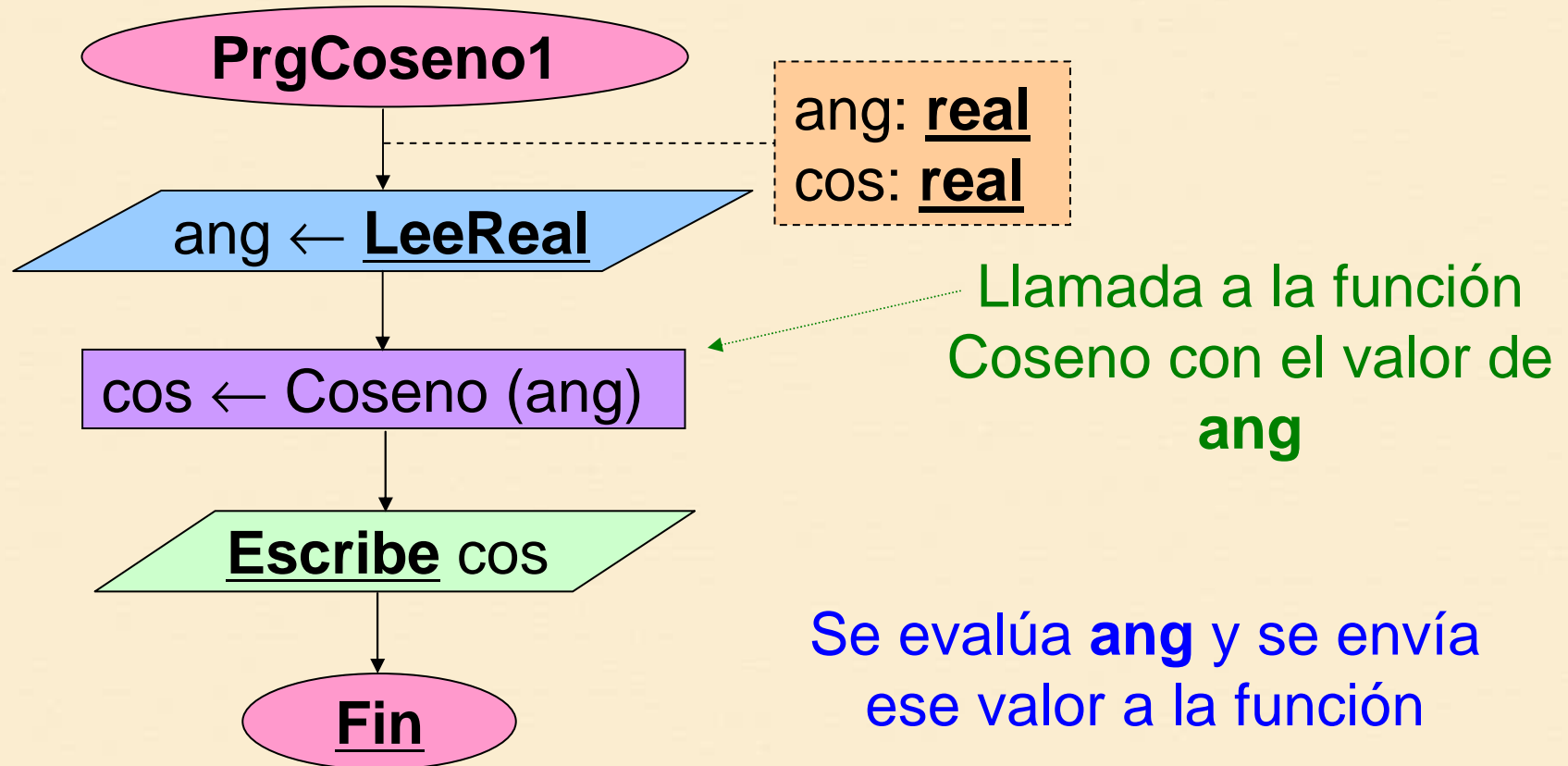
$$y = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$$

fact2i ← (2·i)!

err ← |ter|



1.3 Programa Coseno con subprogramas

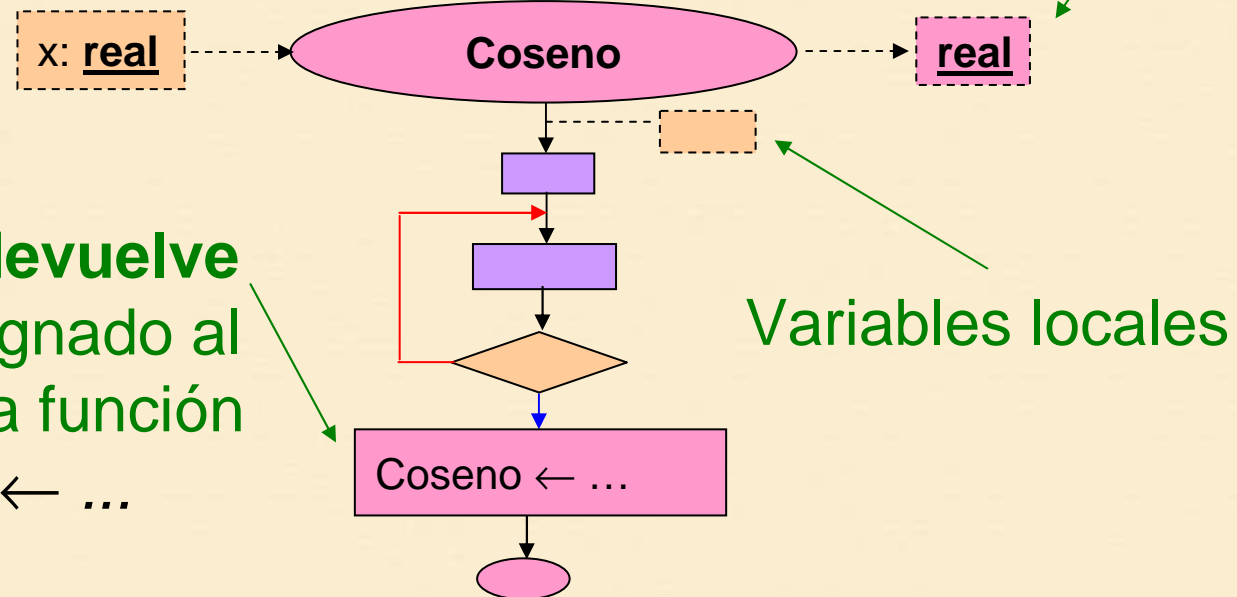


1.4 Interfaz de la función Coseno

Parámetro de entrada: x
Es una variable inicializada
en cada llamada

Tipo del valor devuelto
por la función

La función **devuelve**
un valor, asignado al
nombre de la función
 $\text{Coseno} \leftarrow \dots$



1.5 Llamada a la función Coseno

En la llamada se produce una asignación implícita

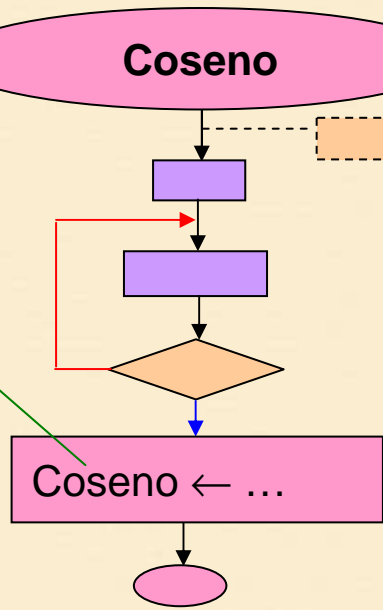
cos ← Coseno (ang)

x ← ang



La "evaluación" de la función es valor devuelto y asignado

Se calcula el valor del coseno



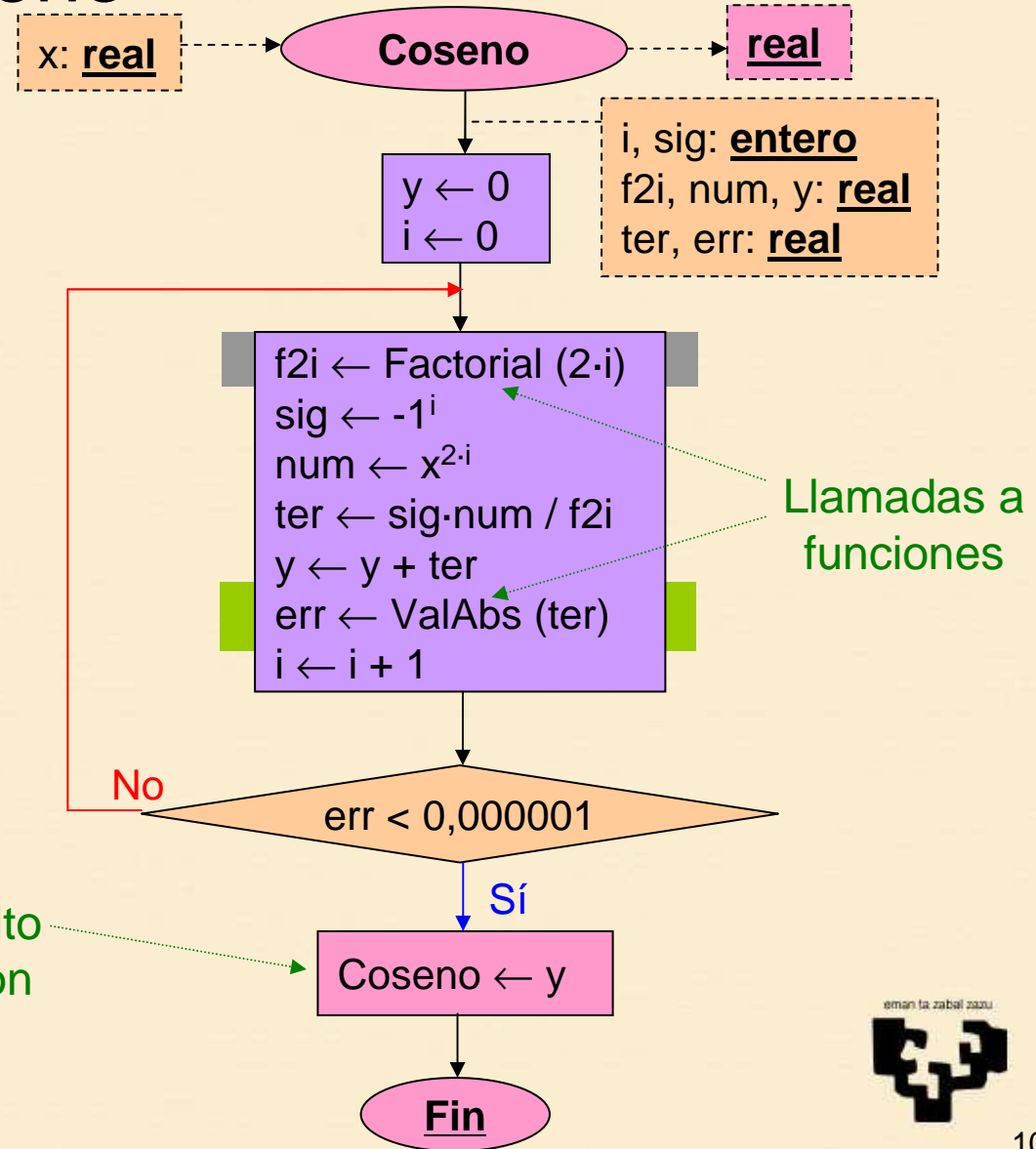
COS ← ...



1.6 Función Coseno

$$\cos(x) = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$$

$$y = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$$



1.7 Función Coseno VB

Function Coseno (ByVal x As Double)_

As Double

Dim i As Integer, sig As Integer

Dim f2i As Integer, num ...

y = 0

i = 0

Do

f2i = Factorial (2*i)

sig = (-1) ^ i

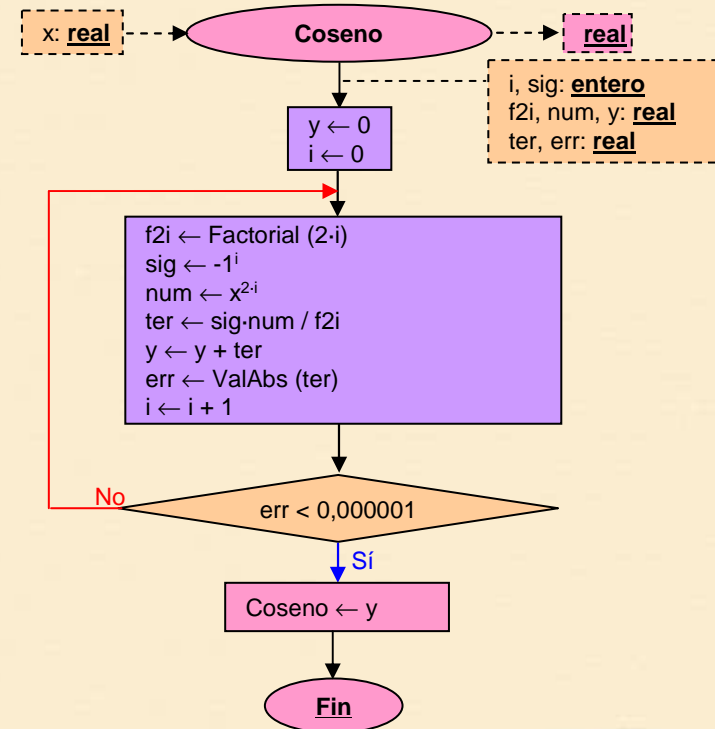
...

Loop Until err < 0.000001

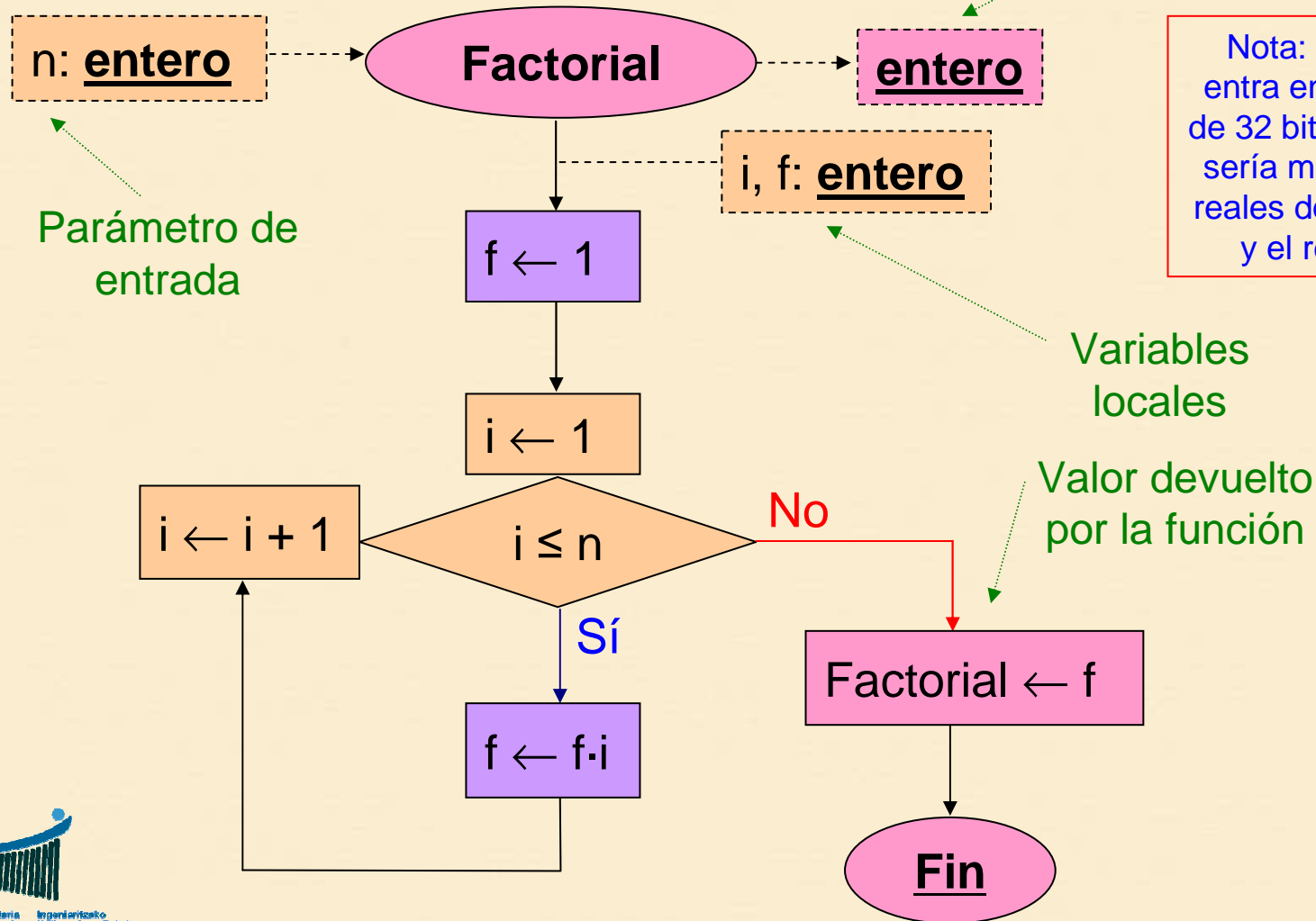
Coseno = y

End Function

$$\cos(x) = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$$



1.8 Función Factorial



Tipo del resultado

Nota: 13! ya no entra en un entero de 32 bits por lo que sería mejor utilizar reales dobles para f y el resultado

Parámetro de entrada

Variables locales

Valor devuelto por la función

1.8 Función Factorial VB

```
Function Factorial (ByVal n As Integer) As Double
```

```
    Dim i As Integer
```

```
    Dim f As Double
```

```
    f = 0
```

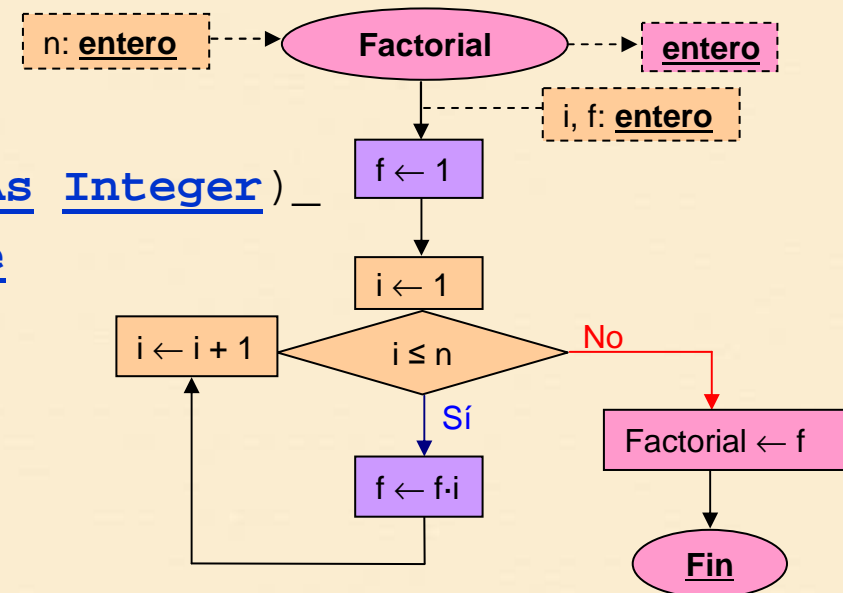
```
    For i = 1 To n Step 1
```

```
        f = f * i
```

```
    Next i
```

```
    Factorial = i
```

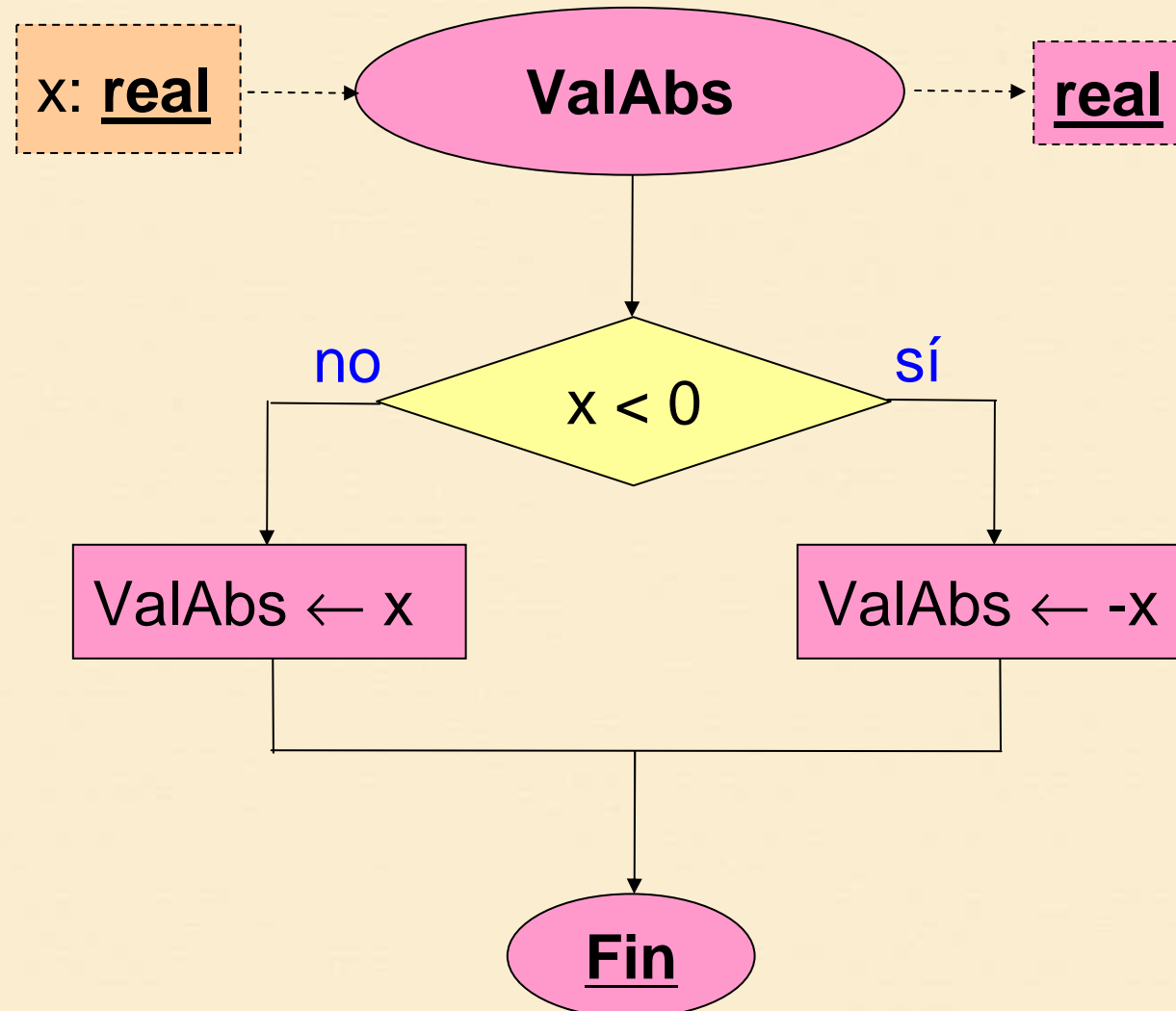
```
End Function
```



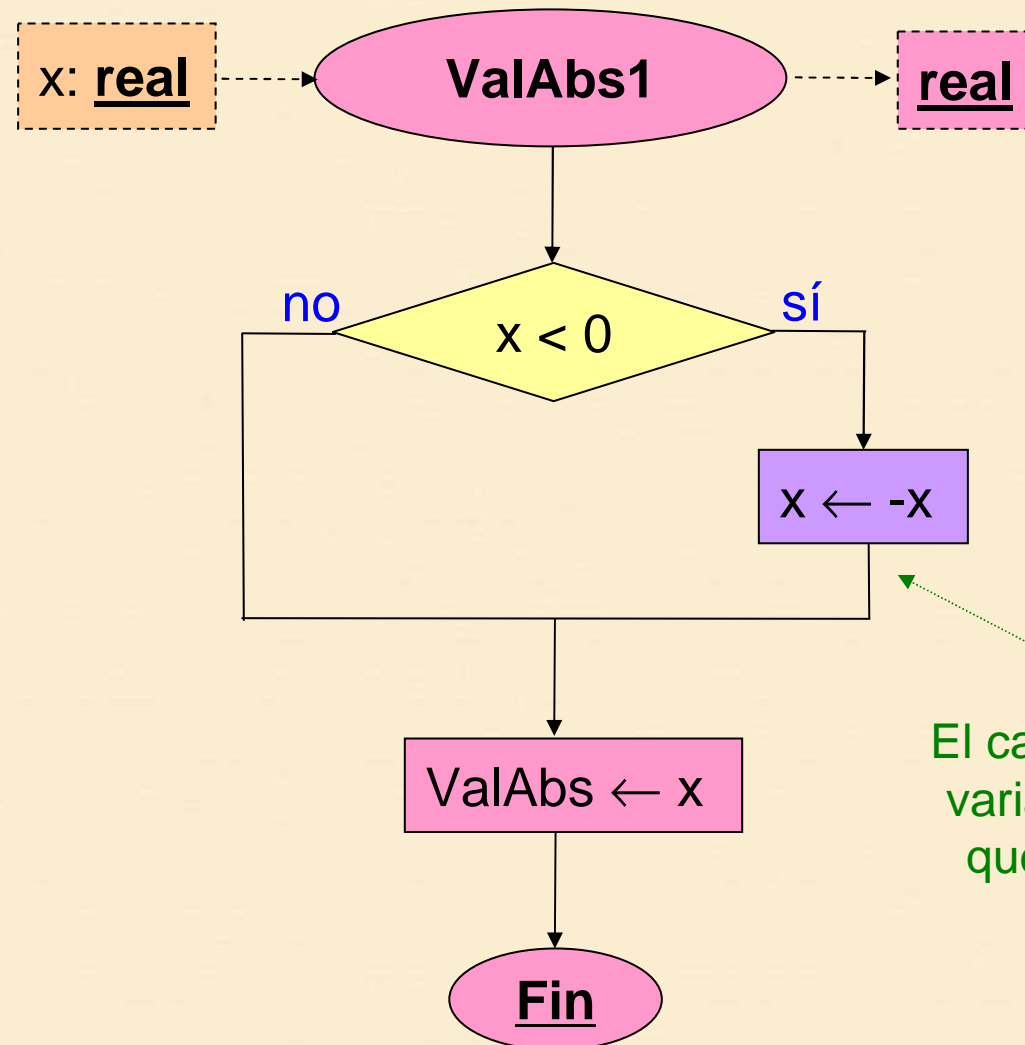
Nota: usamos
doubles en vez de
enteros



1.9 Función ValAbs

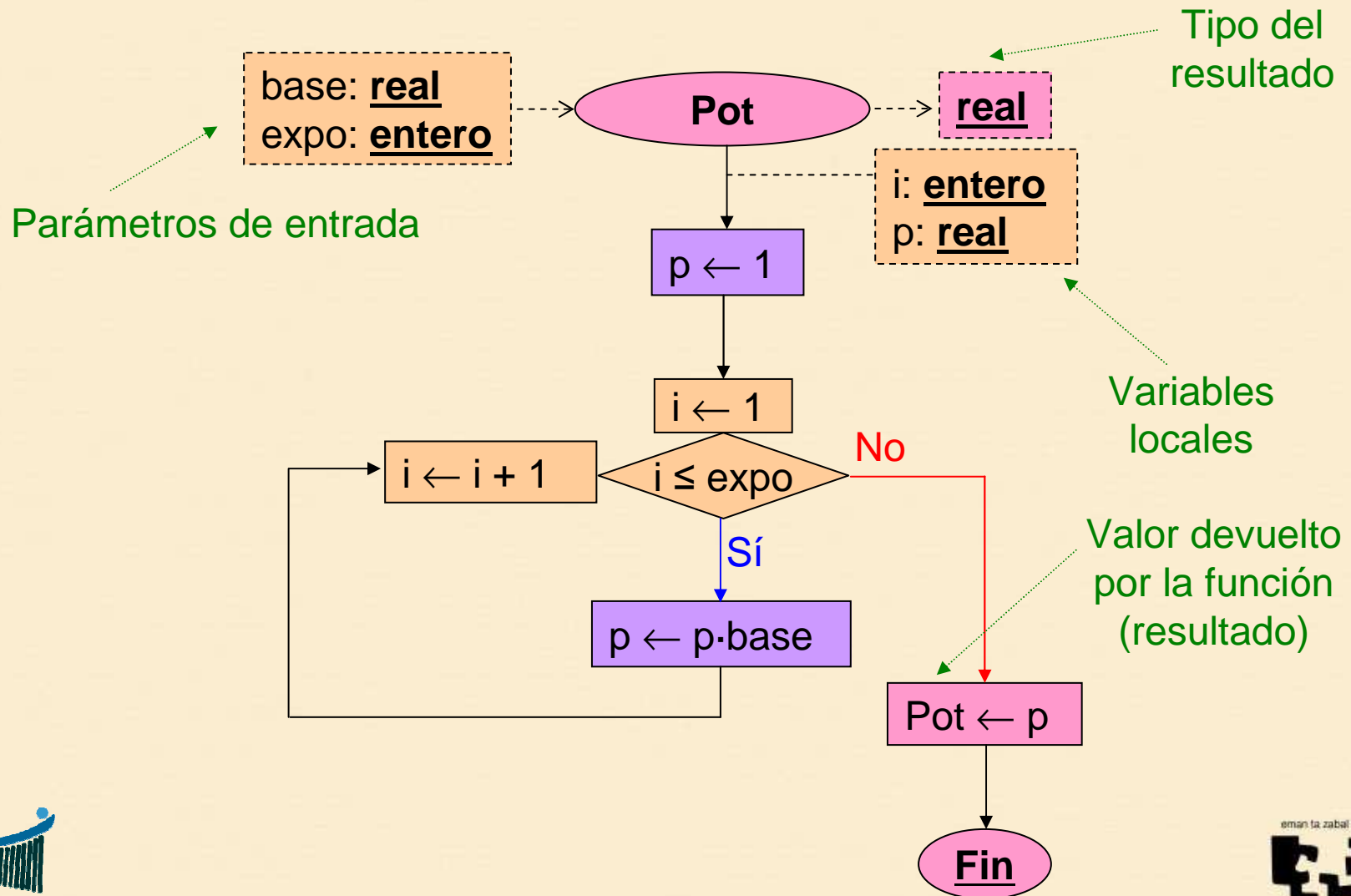


1.10 Función ValAbs1



El cambio no afecta a la variable de llamada ya que se copia el valor

1.11 Función Pot (no necesaria en VB)



2. Suma – Modelos de paso de parámetros

Descripción

Escribir un subprograma que calcule la suma de dos números

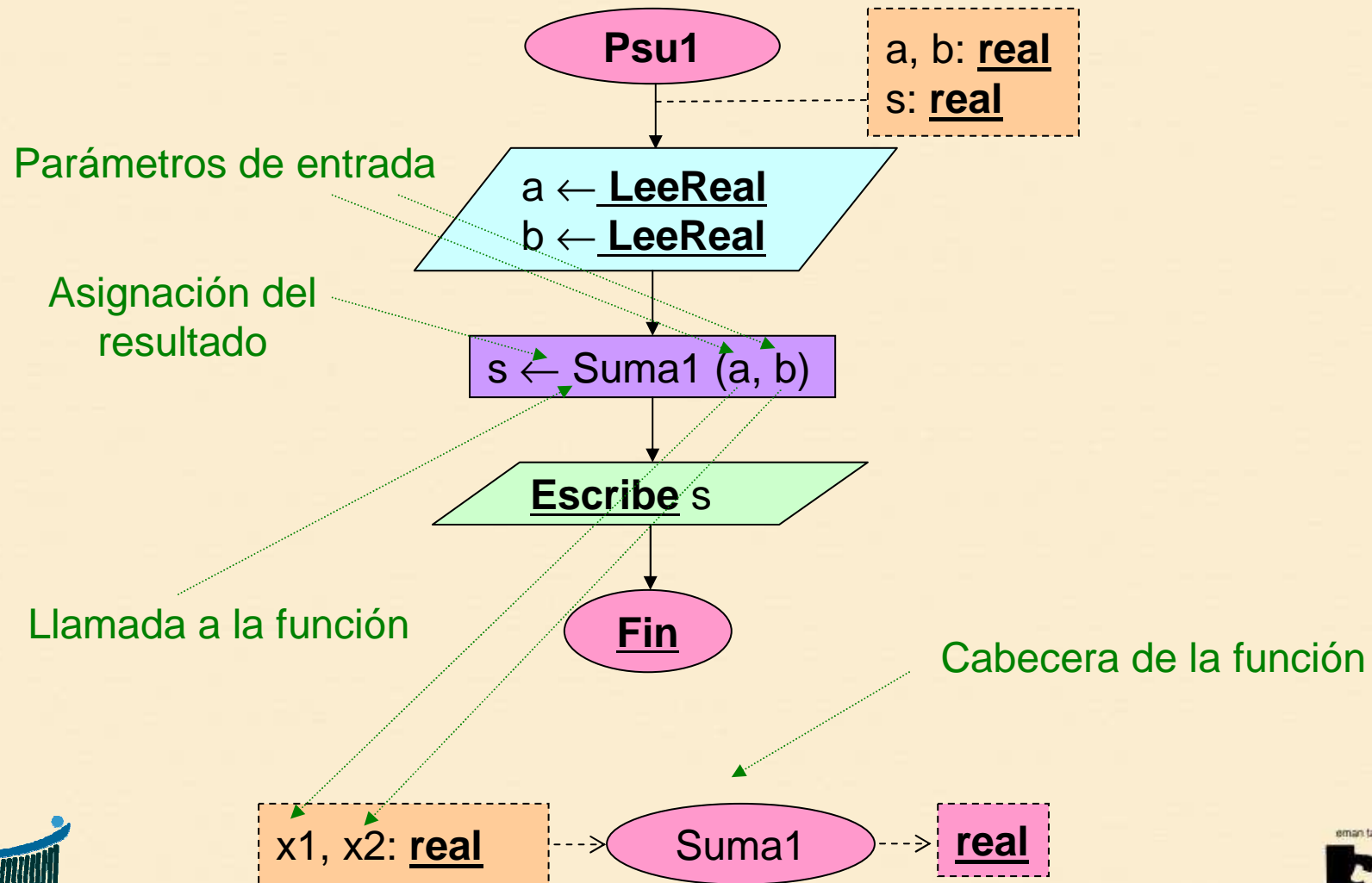
- **Versiones**

1. **función** con **dos** parámetros de **entrada** y **devuelve** el resultado
2. **procedimiento** con **dos** parámetros de **entrada** y **uno** de **salida**
3. **procedimiento** con **un** parámetro de **entrada** y **otro** de **entrada/salida**

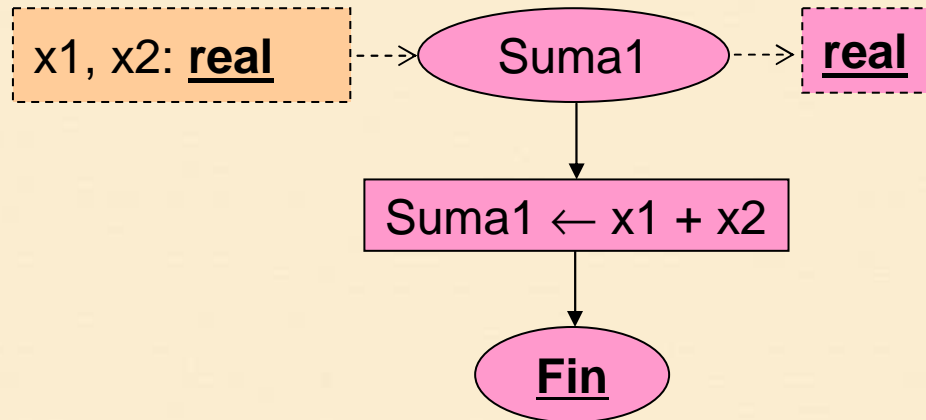
- **Observaciones**

- Paso de parámetros por **valor** y por **referencia**
- Los procedimientos no “devuelven” nada (pueden usar parámetros de salida)

2.1 Suma con función: programa y llamada



2.1 Función Suma1: DdF y VB



```
Function Suma1 (ByVal x1 As Double, ByVal x2 As Double)_  
    As Double  
    Suma1 = x1 + x2  
End Function
```

2.1 Llamadas a la función Suma1

Llamada con variables

```
s ← Suma1 (a, b)
```

...

```
s = Suma1 (a, b)
```

...

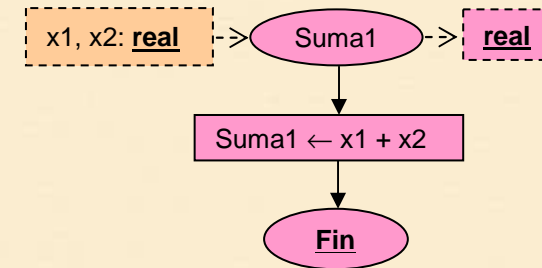
Llamada con constantes

```
s ← Suma1 (5, 7)
```

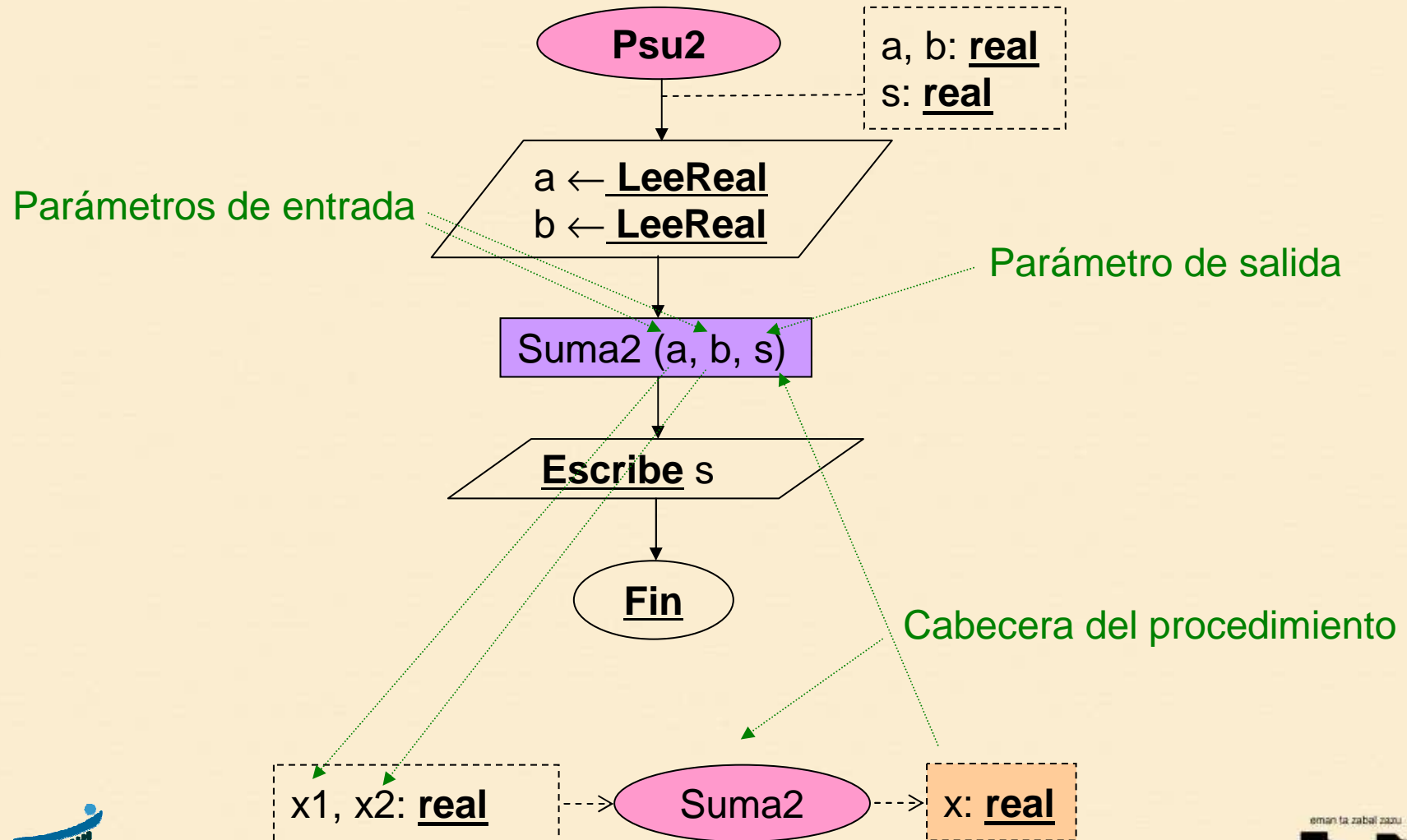
...

```
s = Suma1 (5, 7)
```

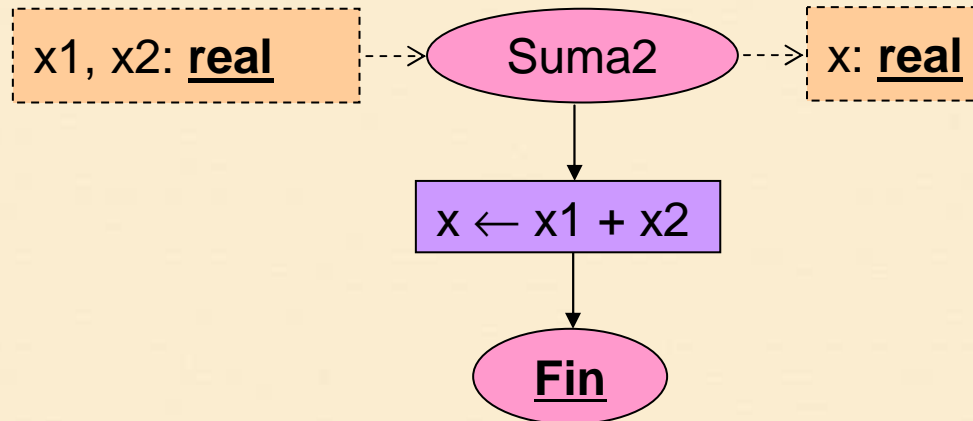
...



2.2 Procedimiento 1: programa y llamada



2.2 Procedimiento Suma2: DdF y VB



```

Sub Suma2 (ByVal x1 As Double, ByVal x2 As Double, _
            ByRef x As Double)

```

```

  x = x1 + x2

```

```

End Sub

```

Puede omitirse

Los cambios realizados sobre las variables pasadas por referencia afectan a las variables de la llamada, que pueden llamarse igual o distinto

2.2 Llamadas al procedimiento Suma2

Llamada con variables

Suma2 (a, b, s)

...

Call Suma2 (a, b, s)

...

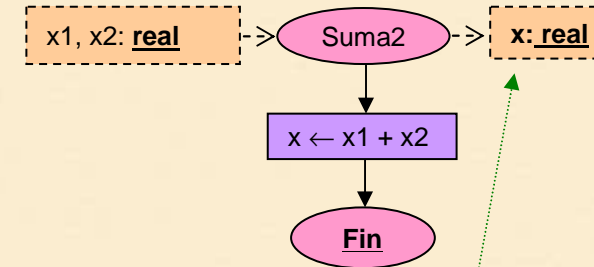
Llamada con constantes (por valor)

Suma2 (5, 7, s)

...

Call Suma2 (5, 7, s)

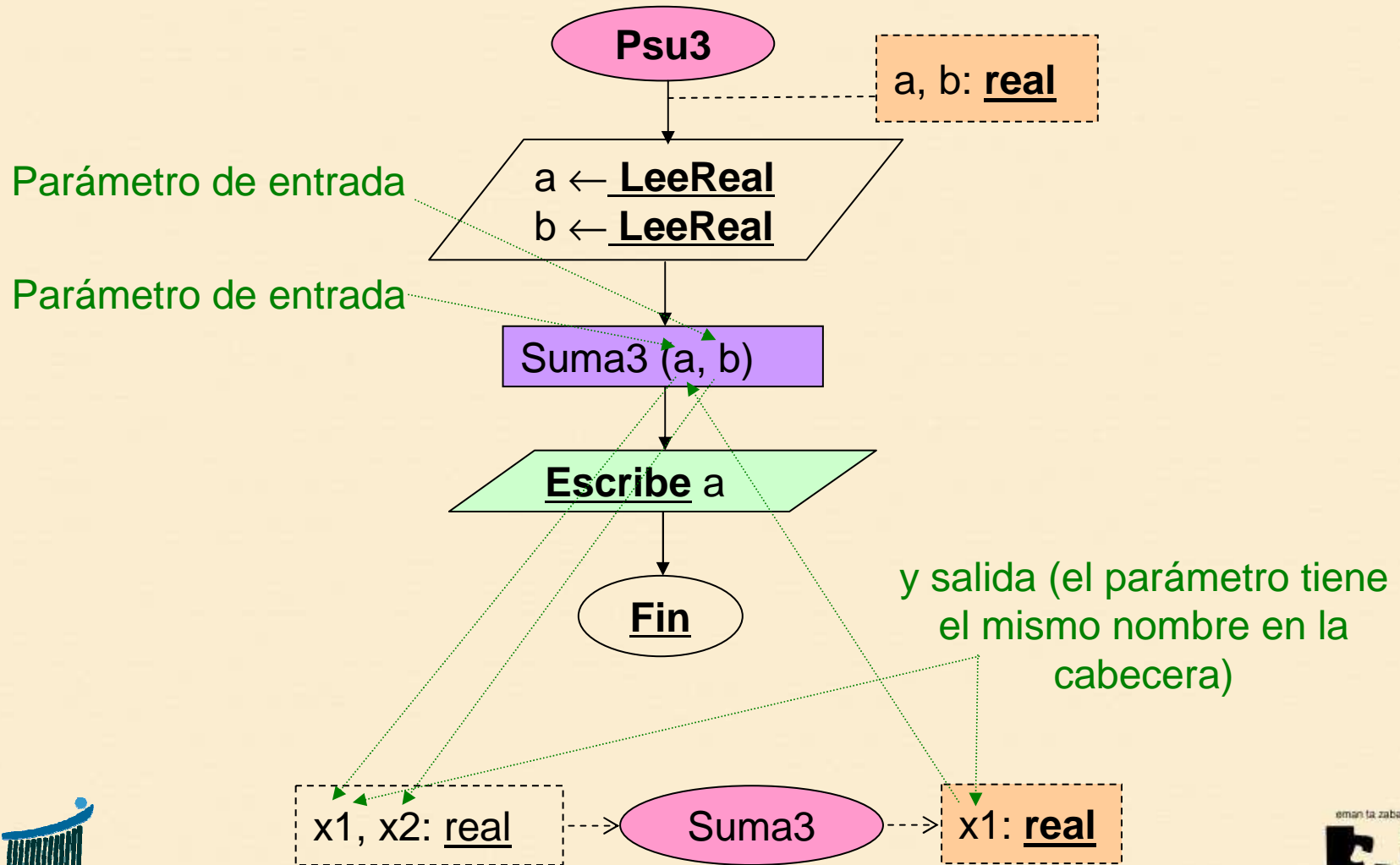
...



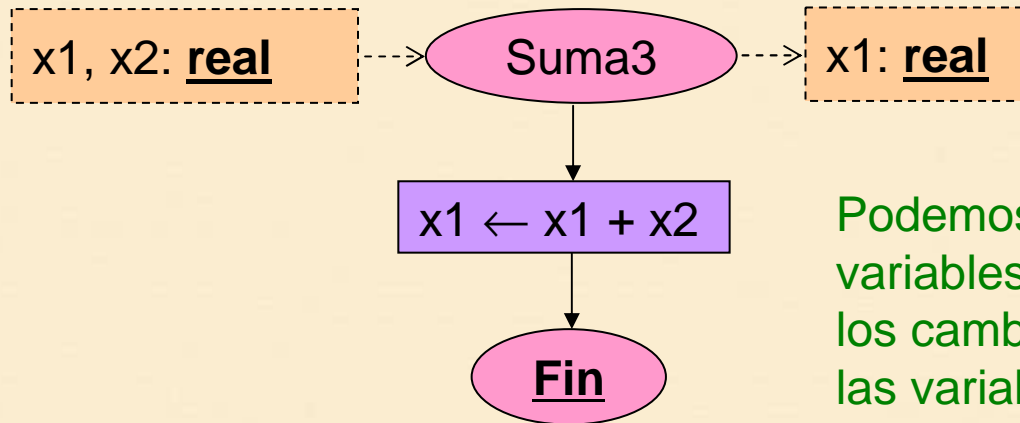
Ha de ser una variable
para recoger el resultado



2.3 Procedimiento 2: programa y llamada



2.3 Procedimiento Suma3: DdF y VB



Podemos leer el valor de las variables pasadas por referencia y los cambios que hagamos afectan a las variables de la llamada

```

Sub Suma3 (ByRef x1 As Double, ByVal x2 As Double)
  x1 = x1 + x2
End Sub
  
```

Puede omitirse

Por la cabecera VB no podemos saber si x1 es de salida o de entrada/salida

2.3 Llamadas al procedimiento Suma3

Llamada con variables

Suma3 (a, b)

...

Call Suma3 (a, b)

...

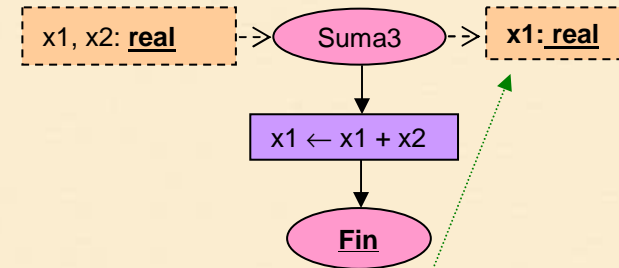
Llamada con constantes

Suma3 (a, 7)

...

Call Suma3 (a, 7)

...



Ha de ser una variable
para recoger el resultado



3. Ecuación de 2º grado

$$ax^2 + bx + c = 0$$

- **Descripción**

- Calcular las raíces de una ecuación de 2º grado

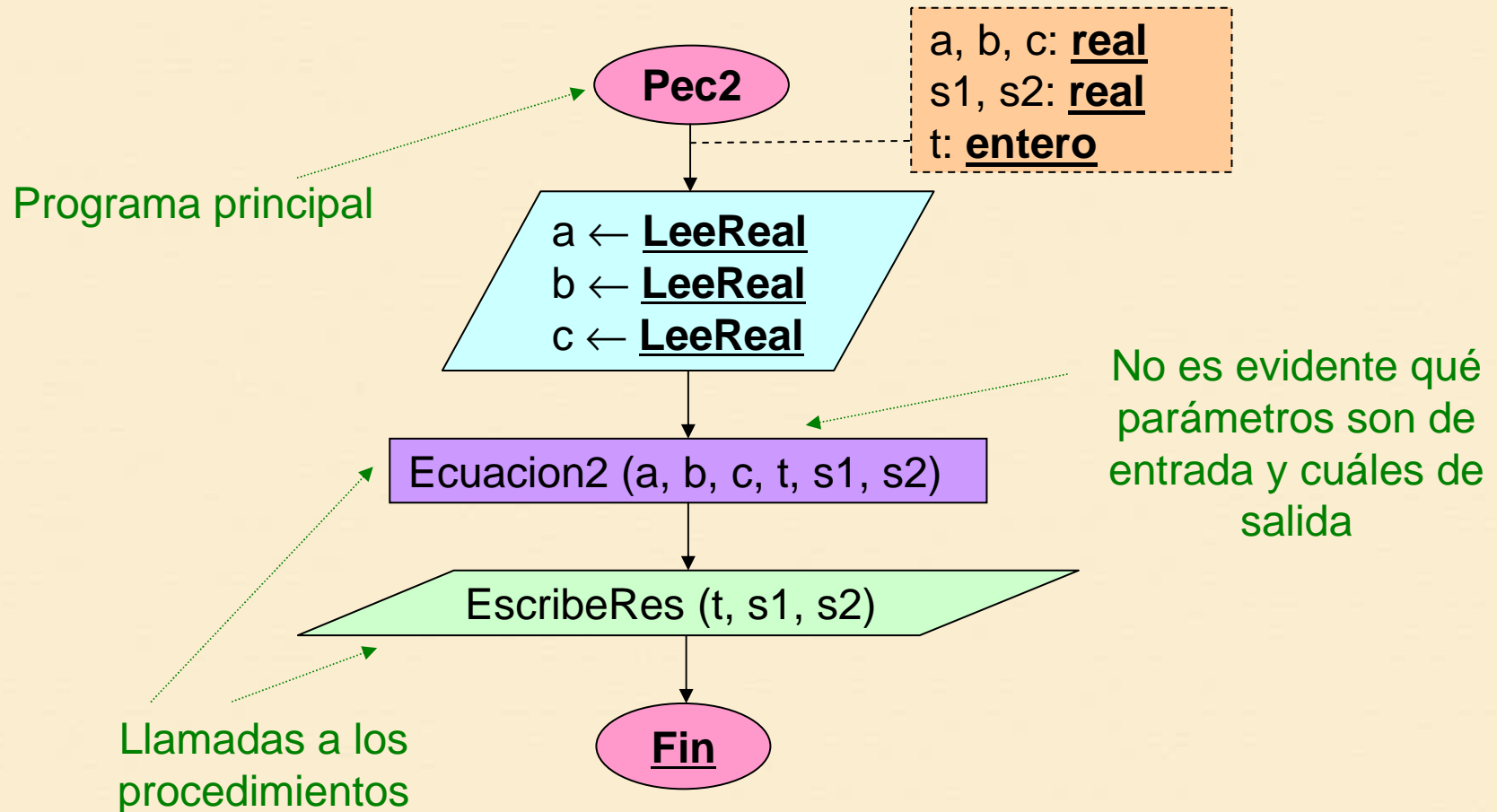
- Tipo 0: No es una ecuación
- Tipo 1: Ecuación lineal
- Tipo 2: Soluciones reales
- Tipo 3: Soluciones complejas

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

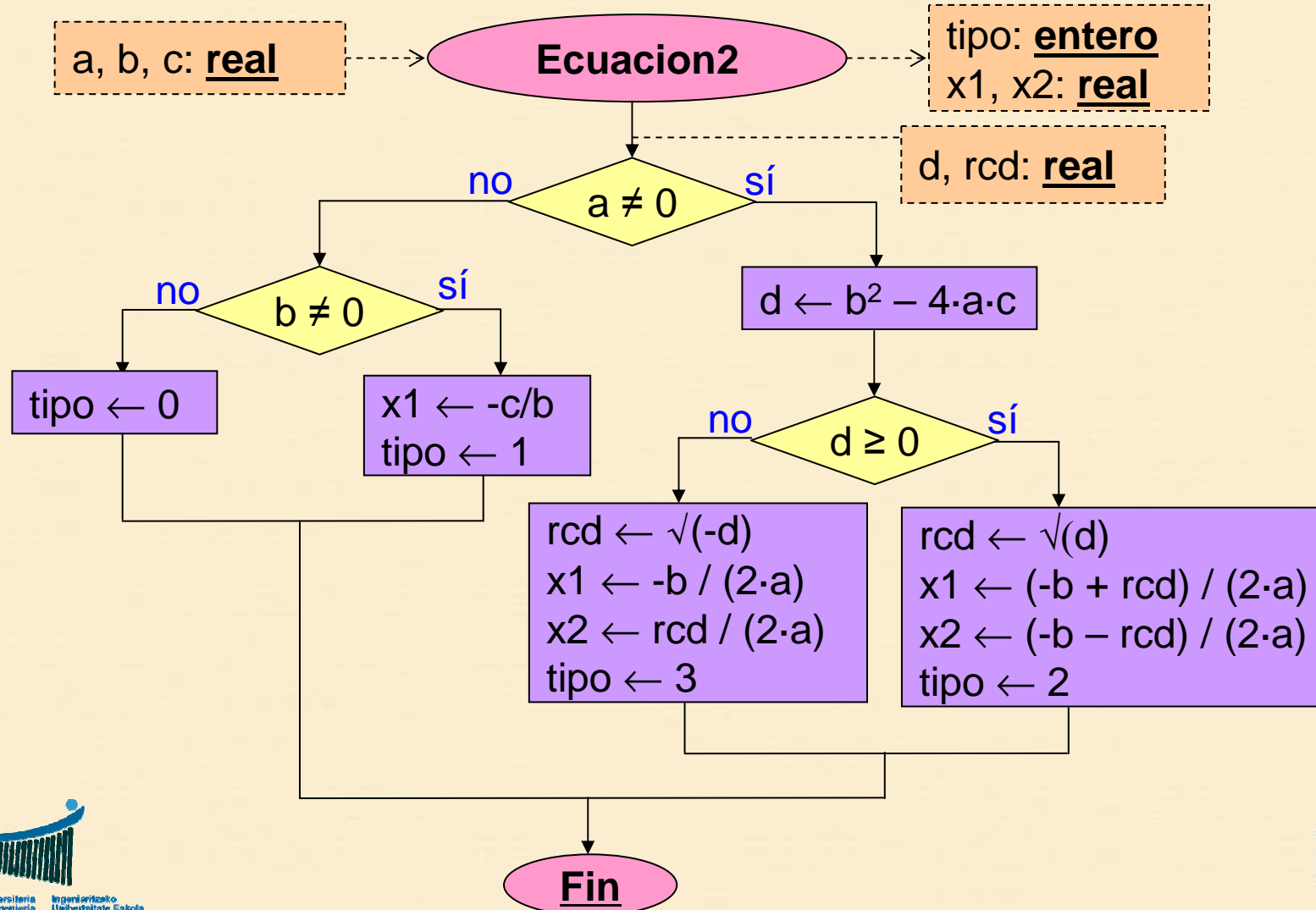
- **Observaciones**

- Paso de parámetros por referencia

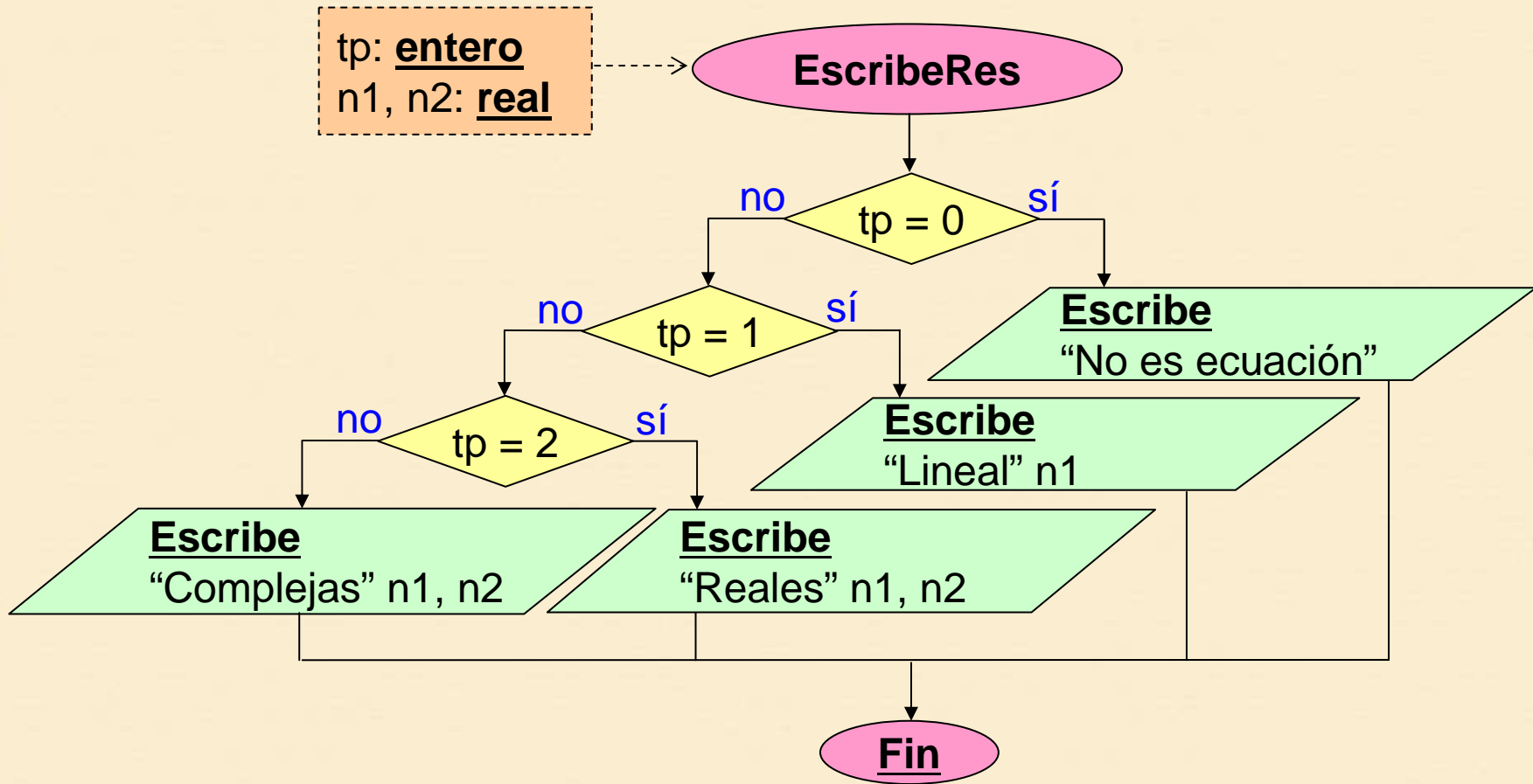
3.1 Programa ecuación de 2º grado - DdF



3.2 Procedimiento Ecuación2 – DdF



3.3 Procedimiento EscribeRes – DdF

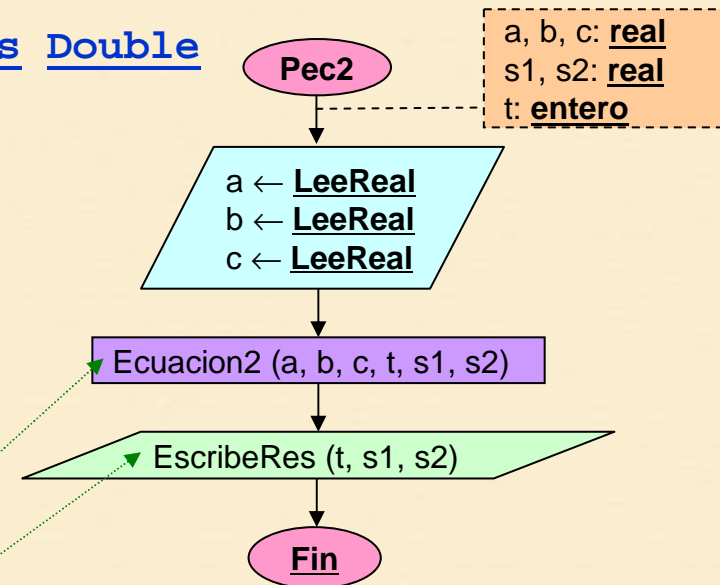


3.4 Programa VB ecuación 2º grado

```

Sub Pec2_Click()
    Dim s As String
    Dim a As Double, b As Double, c As Double
    Dim s1 As Double, s2 As Double
    Dim t As Integer
    s = InputBox ("Coeficiente A")
    a = Cdbl (s)
    s = InputBox ("Coeficiente B")
    b = Cdbl (s)
    s = InputBox ("Coeficiente C")
    c = Cdbl (s)
    Call Ecuacion2 (a, b, c, t, s1, s2)
    Call EscribeRes (t, s1, s2)
End Sub

```



Llamadas a procedimientos



3.5 Procedimiento VB Ecuacion2

```
Sub Ecuacion2 (ByVal a As Double, ByVal b As Double, ByVal c As Double, _
              ByRef tipo As Integer, ByRef x1 As Double, ByRef x2 As Double)
```

```
Dim d As Double, rcd As Double
```

```
If a <> 0 Then
```

```
    d = b*b - 4*a*c
```

```
    If d >= 0 Then
```

```
        rcd = Sqr(d)
```

```
        x1 = (-b + rcd)/(2*a)
```

```
        x2 = (-b - rcd)/(2*a)
```

```
        tipo = 2
```

```
    Else
```

```
        rcd = Sqr(-d)
```

```
        x1 = -b / (2*a)
```

```
        x2 = rcd / (2*a)
```

```
        tipo = 3
```

```
    End If
```

```
Else
```

```
    If b <> 0 Then
```

```
        x1 = -c/b
```

```
        tipo = 1
```

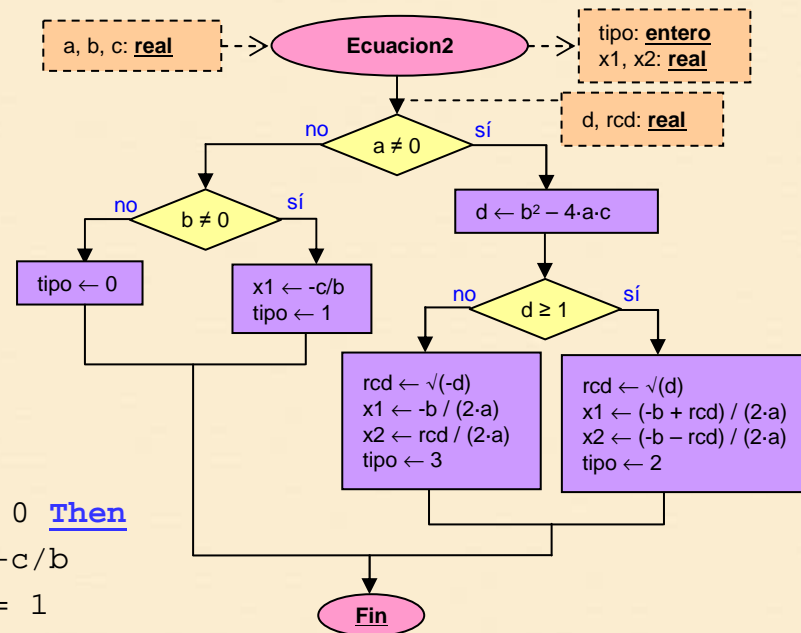
```
    Else
```

```
        tipo = 0
```

```
    End If
```

```
End If
```

```
End Sub
```



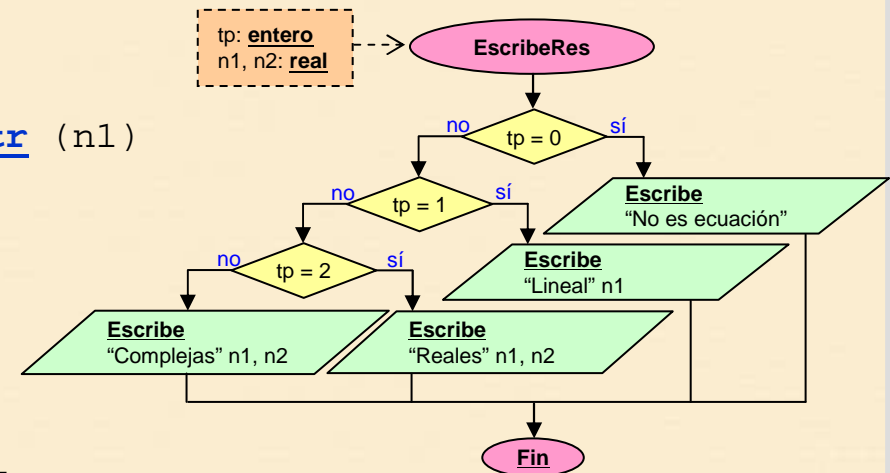
3.6 Procedimiento VB EscribeRes

```

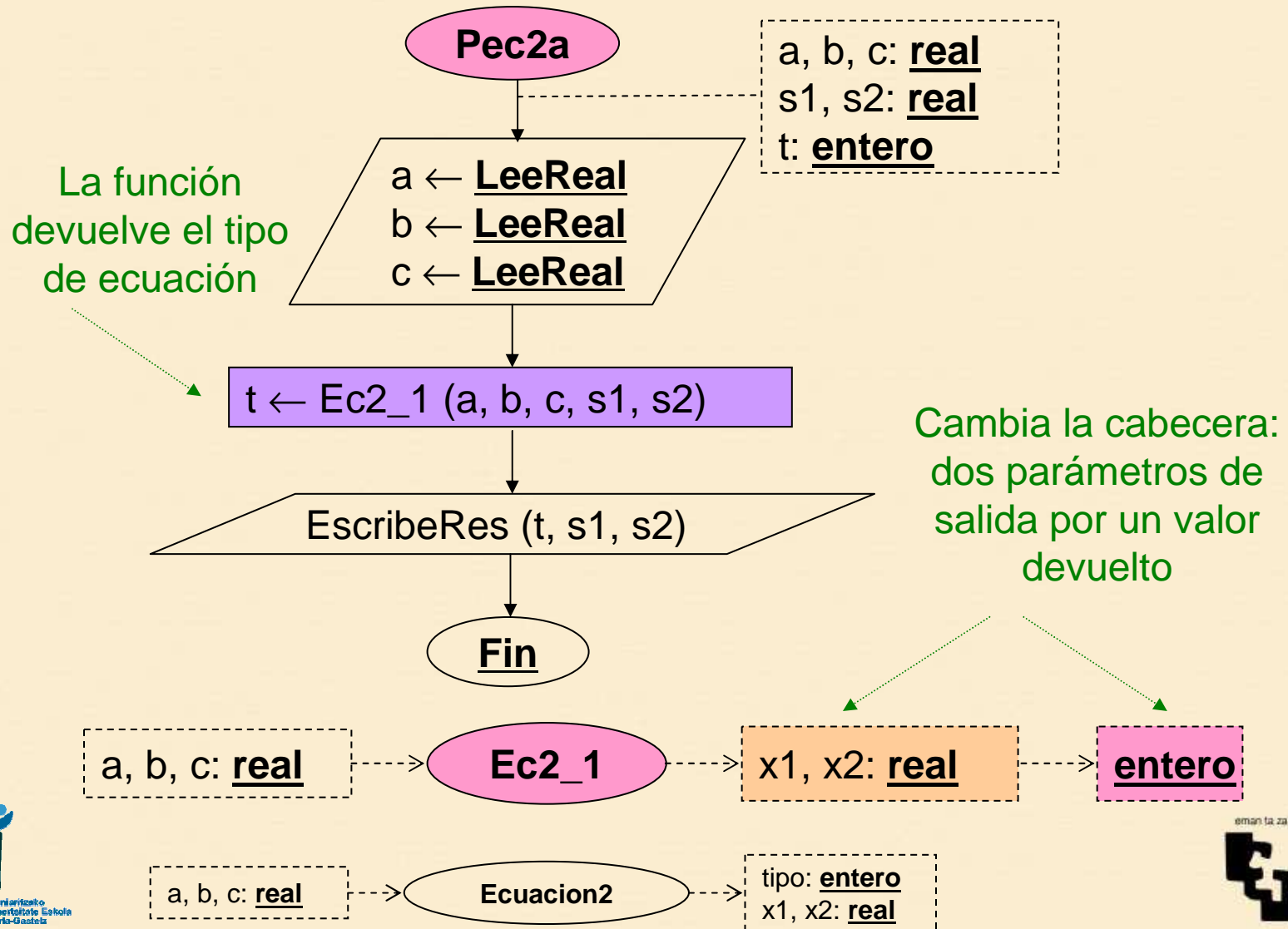
Sub EscribeRes (ByVal tp As Integer, _
               ByRef n1 As Double, ByRef n2 As Double)

    If tp = 0 Then
        MsgBox "No es una ecuación"
    ElseIf tp = 1 Then
        MsgBox "Ecuación lineal. X: " & CStr (n1)
    ElseIf tp = 2 Then
        MsgBox "Soluciones reales. " & _
            " x1: " & CStr (n1) & _
            " x2: " & CStr (n2)
    Else
        MsgBox "Soluciones complejas. " & _
            " x1: " & CStr (n1) & "+" & CStr (n2) & "i" & _
            " x2: " & CStr (n1) & "-" & CStr (n2) & "i"
    End If
End Sub

```



3.7 Programa alternativo con función

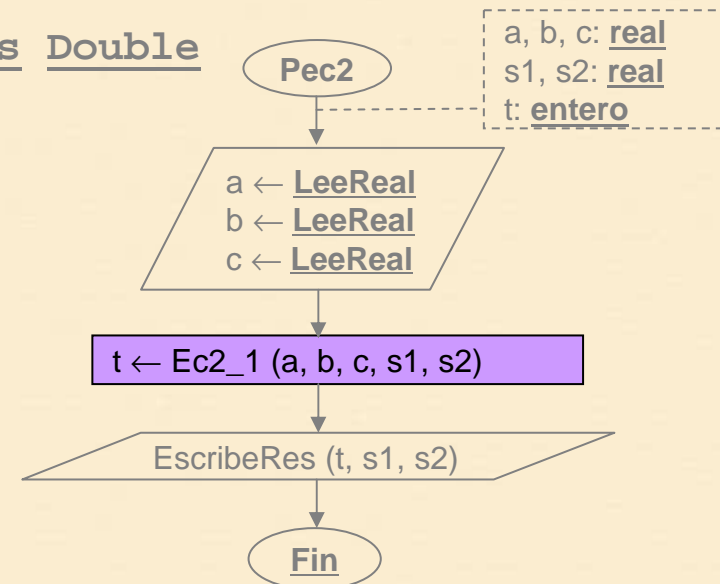


3.8 Alternativa VB ecuación 2º grado

```

Sub Pec2a_Click()
  Dim s As String
  Dim a As Double, b As Double, c As Double
  Dim s1 As Double, s2 As Double
  Dim t As Integer
  s = InputBox ("Coeficiente A")
  a = Cdbl (s)
  s = InputBox ("Coeficiente B")
  b = Cdbl (s)
  s = InputBox ("Coeficiente C")
  c = Cdbl (s)
  t = Ec2_1 (a, b, c, s1, s2)
  Call EscribeRes (t, s1, s2)
End Sub

```



3.9 Función VB Ec2_1

```
Function Ec2_1 (ByVal a As Double, ByVal b As Double, ByVal c As Double, _  
                ByRef x1 As Double, ByRef x2 As Double) As Integer  
Dim d As Double, rcd As Double  
If a <> 0 Then  
    d = b*b - 4*a*c  
    If d >= 0 Then  
        rcd = Sqr (d)  
        x1 = (-b + rcd)/(2*a)  
        x2 = (-b - rcd)/(2*a)  
        Ec2_1 = 2  
    Else  
        rcd = Sqr (-d)  
        x1 = -b / (2*a)  
        x2 = rcd / (2*a)  
        Ec2_1 = 3  
    End If  
Else  
    x1 = -c/b  
    Ec2_1 = 1  
Else  
    Ec2_1 = 0  
End If  
End Function
```

