

# Declaración prototipo

- Una declaración prototipo de algún modulo tiene una sintaxis similar a la de su encabezado.
- Se declara en la sección de **Variables**



- Es necesario hacer una declaración forward cuando se hace el llamado del modulo antes de haberlo definido.

Principal()

Inicio

Variabales: Entero m1(), r

r ← m1()

Regresa()

Fin\_Principal

Entero m1()

Inicio

m2()

....

Regresa(4)

Fin\_m1

# Ejemplo

Principal()

Inicio

coment: declaración de un modulo prototipo

Variables:

Real SumaR(Real x, Real y) coment: Real SumaR(Real, Real)  
Real SumaR(Real k, Real m)

Real r, a  $\leftarrow$  3.14, b

r  $\leftarrow$  SumaR(a,b) coment: llamada al modulo

Escribir("el resultado es ",r)

Regresa()

Fin\_Principal

coment: definición del modulo

Real SumaR(Real x, Real y)

Inicio

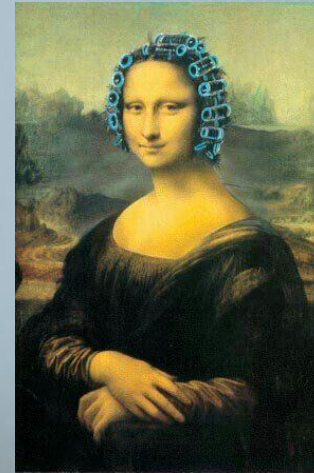
Regresa(x + y)

Fin\_SumaR

# Paso de parámetros

## Por valor

- El módulo sólo recibe una copia del valor del parámetro
- No puede acceder al parámetro



En algún  
lugar



Museo del  
Louvre en  
París,  
Francia



# Sintaxis

## Definición

tipo nombre\_módulo(tipo d1, tipo d2)

nombre\_módulo(tipo d1, tipo d2)

nombre\_módulo()

## Llamado

nomvar ← nombre\_módulo(d1, d2)

nombre\_módulo(d1, d2)

nombre\_módulo()

# Ejemplo

Entero SumaDig(Entero num, Entero condig)<sup>514</sup><sup>0</sup>

Variables: r, dig: Entero

Inicio

dig  $\leftarrow$  0

Repetir

r  $\leftarrow$  num mod 10

condig  $\leftarrow$  condig+1

dig  $\leftarrow$  dig + r

num  $\leftarrow$  num / 10

Hasta(num  $\leq$  0)

Regresa(dig)

0 1 2 3

Fin\_SumaDig



10

# Ejemplo----- SumaDig

Principal()

Inicio

Entero: Sum, num, condig

condig  $\leftarrow$  0

Leer(num)

Sum  $\leftarrow$  SumaDig(num, condig)

Regresa()

Fin\_Principal



# Por referencia

- Pasa la dirección de memoria del parámetro (variable)
- Se puede acceder y modificar su valor durante la ejecución del módulo





# Sintaxis

## Definición

tipo nombre(tipo var d1, tipo var d2)

## Llamado

nomvar ← nombre\_módulo(dir d1, dir d2)

# Ejemplo-----SumaDig

Entero SumaDig(Entero num, Entero **var** condig)

Variables: Entero r, dig

Inicio

dig  $\leftarrow$  0

Repetir

r  $\leftarrow$  num mod 10

**var** condig  $\leftarrow$  **var** condig+1

dig  $\leftarrow$  dig + r

num  $\leftarrow$  num / 10

Hasta(num $\leq$ 0)

Regresa(dig)

Fin\_SumaDig



0 1 2 3

10

# Ejemplo----- SumaDig

Principal()

Inicio

Variables: Entero Sum, num,  
                  condig

condig  $\leftarrow$  0

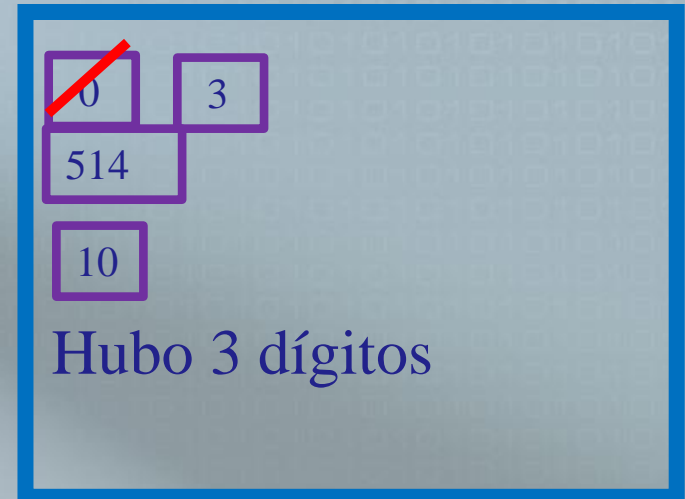
Leer(num)

Sum  $\leftarrow$  SumaDig(num, dir condig)

Escribir ("hubo", condig, "dígitos")

Regresa()

Fin\_Principal



# Ejemplo---GenreaFibo

Entero GeneraFibo(Entero N, Entero **var** S)

Variables: Entero Fn-1, Fn-2, Fn, i

Inicio

Fn-1  $\leftarrow$  0, Fn-2  $\leftarrow$  1

Para (i $\leftarrow$ 1 hasta N, incremento 1)

Fn  $\leftarrow$  Fn-1 + Fn-2

**var** S  $\leftarrow$  **var** S + Fn

Fn-1  $\leftarrow$  Fn-2

Fn-2  $\leftarrow$  Fn

Fin\_Para

Regresa(Fn)

Fin\_GeneraFibo

$$f_n = f_{n-1} + f_{n-2}$$

$$f_0 = 0$$

$$f_1 = 1$$

$$f_2 = f_1 + f_0 = 1$$

$$f_3 = f_2 + f_1 = 2$$

...



# Ejemplo----- GeneraFibo

Principal()

Inicio

Entero: F, x,y, Sum

Sum  $\leftarrow$  0

Leer(x)

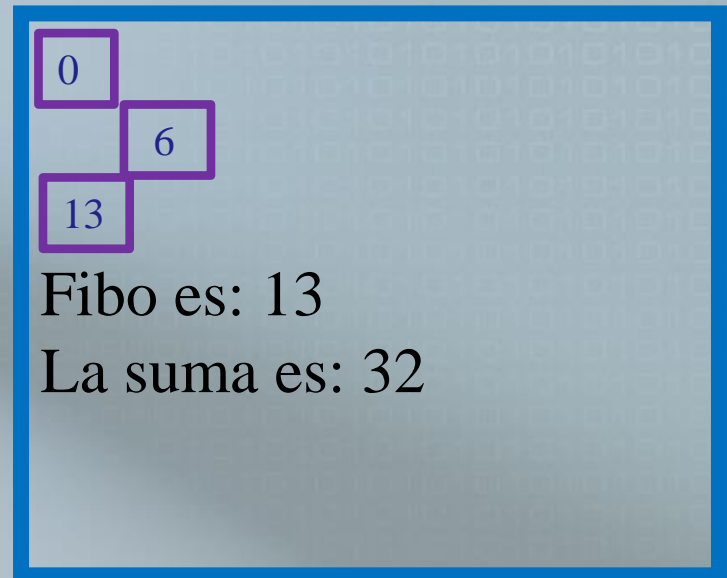
F  $\leftarrow$  GeneraFib(x, dir Sum)

Escribir("Fibo es: ", F)

Escribir("La suma es: ", Sum)

Regresa()

Fin\_Principal



# Práctica-----1

Entero QueHago(Entero **var** A)

Inicio

Variables: Entero res

res  $\leftarrow$  1

Mientras (**var** A > 1)

    res  $\leftarrow$  res \* **var** A

**var** A  $\leftarrow$  **var** A - 1

Fin\_Mientras

Regresa(res)

Fin\_QueHago

Principal()

Inicio

Variables: Entero: F, x

Leer(x)

F ← QueHago(dir x)

Escribir("Se calculó para:", x, "y dio  
como resultado " F)

Regresa()

Fin\_Principal

1. ¿Qué hace el  
algoritmo?

2. ¿Qué valor regresa si x  
vale 5 ?

3. ¿Cuál es el error?

# Práctica-----2

Entero QueHare(Entero A, Real p)

Inicio

Variables: Real num[100], s←0.0, s1← 0.0 Entero res, i, R ← 0

res ← 1

i ← 1

Mientras (A >=1) AND (A <= 100)

Leer(num[i])

10 10 8 9 10

s1 ← s1 + num[i]

i ← i+1

Fin\_Mientras

p ← s1/A

Mientras (res<=A)

s ← s + num[res]

res ← res + 2

R ← R+1

Fin\_Mientras

Regresa(s/R)

Fin\_QueHare



Principal()

Inicio

Variables: Entero x Real res, F

Leer(x)

F ← QueHare(x , res)

Escribir("Un resultado es ", F)

Escribir("y el otro resultado es ", res)

Regresa()

Fin\_Principal

1. ¿Qué hace el algoritmo?

2. ¿Qué valores se imprimen si x vale 5 y los valores del arreglo?

3. ¿Cuál es el error?