

Control de Flujo: caminos alternativos

Clase #04

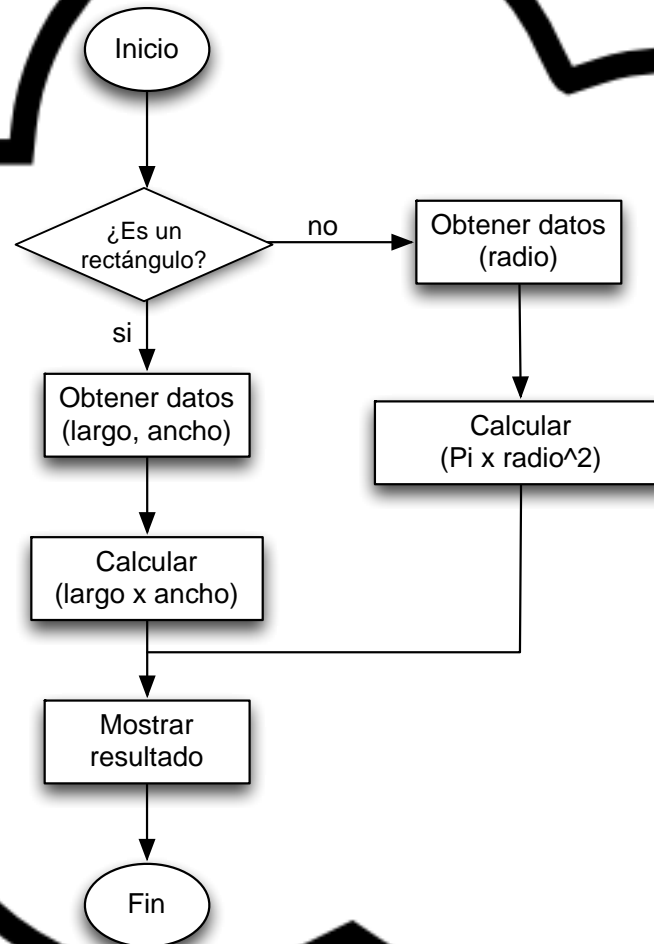
IIC1103 – Introducción a la Programación

Marcos Sepúlveda (marcos@ing.puc.cl)

Veremos hoy ...

Control de flujo:

- ▶ Instrucción ***if***
- ▶ Instrucción ***if / else***
- ▶ Instrucción ***if / elif / else***




**¿Recuerdan
caminos
alternativos?**

Control de Secuencia: *if*

► Sintaxis

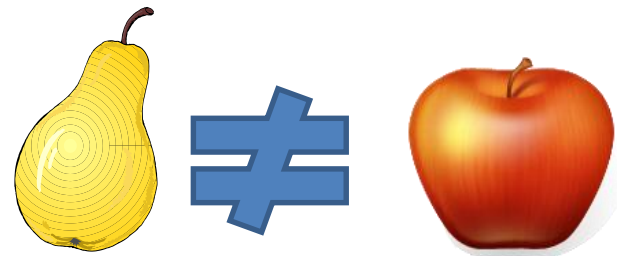
```
if condicion:  
    instruccion  
    ...
```



- Python evalúa la ***condicion***.
- Si el resultado de la evaluación es ***True***, entonces se ejecutan las instrucciones dentro del ***if***
- Si el resultado de la evaluación es ***False***, entonces NO se ejecutan las instrucciones dentro del ***if***

¿Qué es una condición?

- ▶ Las condiciones están compuestas por operandos, operadores lógicos y operadores relacionales que da como resultado un valor booleano (verdadero o falso)



Repaso: operadores relacionales

Operador	Descripción	Ejemplo
$a == b$	¿son iguales a y b ?	$5 == 5$ # True $6 == 8$ # False
$a != b$	¿son distintos a y b ?	$21 != 5$ # True $6 != 6$ # False
$a < b$	¿es a menor que b ?	$5 < 9$ # True $6 < 2$ # False
$a > b$	¿es a mayor que b ?	$6 > 5$ # True $6 > 6$ # False
$a <= b$	¿es a menor o igual que b ?	$5 <= 5$ # True $6 <= -10$ # False
$a >= b$	¿es a mayor o igual que b ?	$5 >= 5$ # True $6 >= 8$ # False

Repaso: operadores lógicos

► *and*

- Requiere que ambas expresiones sean ciertas

and		
operandos		resultado
izquierdo	derecho	
<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>True</i>	<i>False</i>
<i>False</i>	<i>False</i>	<i>False</i>

► *or*

- Basta con que alguna de las expresiones sea cierta

or		
operandos		resultado
izquierdo	derecho	
<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>False</i>

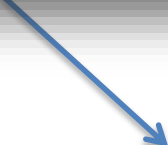
Repaso: operadores lógicos

Operador	Descripción	Ejemplo
<i>a and b</i>	¿se cumplen <i>a</i> y <i>b</i> ?	<i>5 == 5 and 4 != 1 # True</i> <i>9 == 8 and 10 != 9 # False</i>
<i>a or b</i>	¿se cumple al menos uno de <i>a</i> o <i>b</i> ?	<i>9 == 8 or 10 != 9 # True</i> <i>5 == 7 or 10 != 10 # False</i>
<i>not a</i>	¿es <i>a</i> falso?	<i>not (5 == 1) # True</i> <i>not (10 == 10) # False</i>

Control de Secuencia: *if*

► Sintaxis

```
if condicion:  
    instruccion  
    ...
```



Nota:

- Existen espacios delante de las instrucciones. Esto se llama indentación.
- Las instrucciones que queremos estén dentro del *if* deben estar indentadas.
- Python IDLE lo realiza automáticamente.

Ejemplo – ¿eres mayor de edad?

```
edad = int(input("Ingresa tu edad: "))  
if edad >= 18:  
    print("Eres mayor de edad")
```

Control de Secuencia: *if-else*

► Sintaxis

```
if condicion:  
    instruccion  
    ...  
else:  
    instruccion  
    ...
```

- A la sintaxis de *if* se puede agregar ***else***, que ejecuta instrucciones (que están dentro del ***else***) cuando el resultado de la evaluación de la condición del *if* es falsa.

Ejemplo – ¿eres mayor de edad?

```
edad = int(input("Ingresa tu edad: "))
if edad >= 18:
    print("Eres mayor de edad")
else:
    print("Eres menor de edad")
```

Ejemplo – área de un rectángulo o un círculo

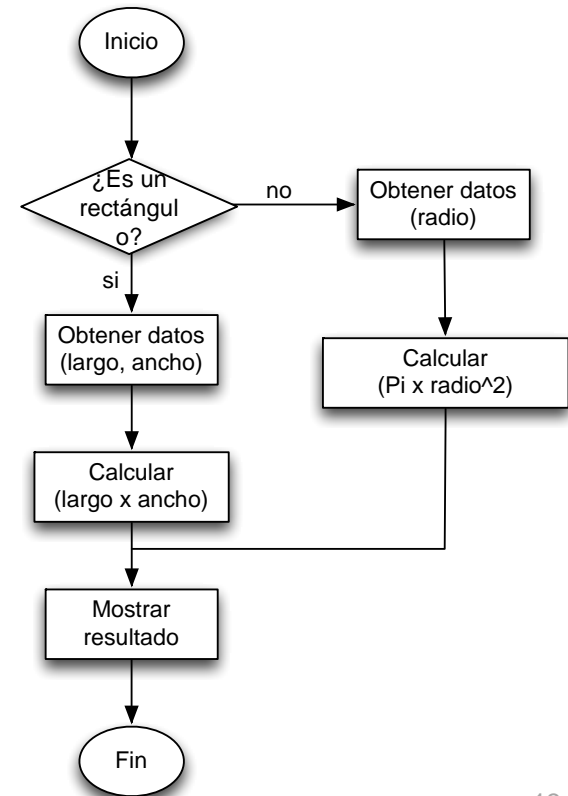
```
import math

print("Este programa calcula el área de una rectángulo o un círculo")

respuesta = int(input("¿Es un rectángulo? (1:sí; otro número:no): "))

if respuesta == 1:
    largo = float(input("Ingrese largo: "))
    ancho = float(input("Ingrese ancho: "))
    area = largo * ancho
else:
    radio = float(input("Ingrese radio: "))
    area = math.pi * radio ** 2

print("El área es: " + str(area))
```



Control de Secuencia: *if-elif-else*

► Sintaxis

```
if condicion:  
    instruccion  
    ...  
elif condicion:  
    instruccion  
    ...  
else:  
    instruccion  
    ...
```

- A la sintaxis de *if* se puede agregar *elif*, que ejecuta instrucciones (que están dentro del *elif*) cuando el resultado de la evaluación de la condición del *if* es falsa. *elif* también tiene una condición que evaluar, que debe ser verdadera.

Control de Secuencia: *if-elif-else*

► Sintaxis

```
if condicion:  
    instruccion  
    ...  
elif condicion:  
    instruccion  
    ...  
else:  
    instruccion  
    ...
```

if

- Se ejecutan las instrucciones de ***if*** si el resultado de su condición es verdadero.

elif

- Sólo se ejecutan las instrucciones de ***elif*** si resultado de su condición es verdadero y el resultado de la condición de ***if*** era falsa.

else

- Sólo se ejecutan las instrucciones de ***else*** si las evaluaciones de las condiciones de ***if*** y ***elif*** eran falsas.

Ejemplo – ¿cuál es el mayor de tres números?

```
print("Ingresa tres números: ")
a = float(input())
b = float(input())
c = float(input())

if a > b and a > c:
    mayor = a
elif b > c:
    mayor = b
else:
    mayor = c

print("El número mayor es: " + str(mayor))
```


Ejercicio – año es bisiesto

```
'''
```

```
Un año es bisiesto si es divisible entre 4, a menos que sea  
divisible entre 100. Sin embargo, si un año es divisible entre 100  
y además es divisible entre 400, también resulta bisiesto.
```

```
'''
```

```
año = int(input("Ingrese un año cualquiera: "))
```

```
if ((año%4 == 0) and (año%100 != 0)) or (año%400 == 0):
```

```
    print(str(año) + " es un año bisiesto")
```

```
else:
```

```
    print(str(año) + " no es un año bisiesto")
```