

# Estructuras de control

...

Programación  
2024

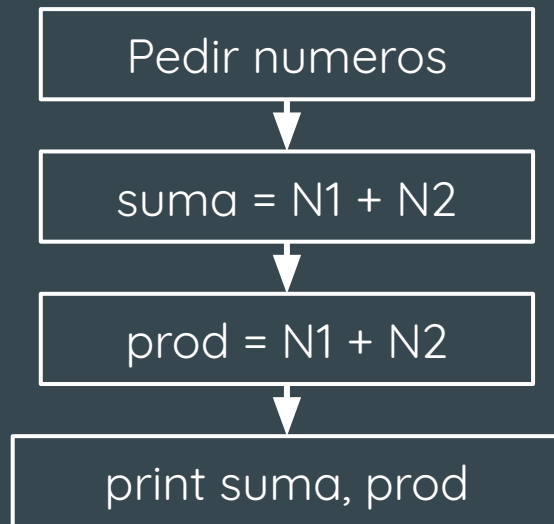
# Estructuras de control

- El flujo de control se refiere al orden en que se ejecutan las sentencias del programa.
- Cualquier algoritmo se puede construir utilizando combinaciones de tres estructuras de control:
  - Secuencial
  - *Selectiva* o *condicional*
  - Repetitiva o iterativa.

# Algoritmo Secuencial

- Se ejecutan las sentencias *en orden*
- Ejemplo: calcular la suma y el producto de dos números

- 1) Pedir números. Guardarlos como N1 y N2
- 2)  $\text{suma} = \text{N1} + \text{N2}$
- 3)  $\text{prod} = \text{N1} * \text{N2}$
- 4) Devolver suma, prod

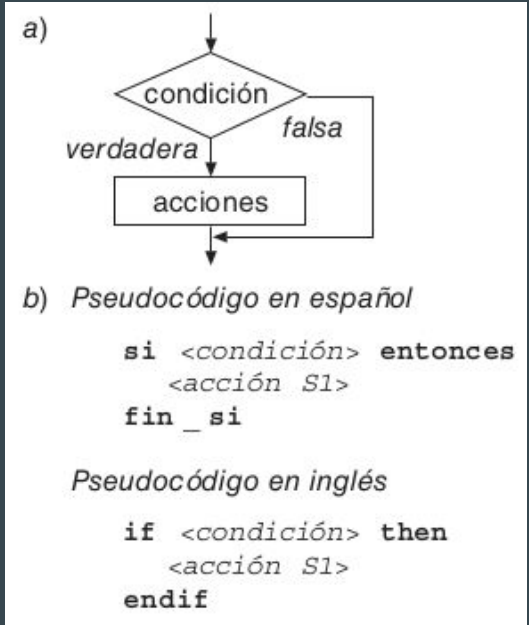


# Estructuras Selectivas

- Se utilizan para tomar decisiones lógicas.
- Se evalúa una **condición** y en función del resultado de la misma se realiza o no una opción.
- Para definir una condición se utiliza la palabra reservada **if**.
- Las estructuras selectivas pueden ser:
  - Simples
  - Dobles
  - Múltiples

# Estructura Simple

- Existe una sola condición.
  - Si la condición es verdadera, ejecutar la acción S1.
  - Si la condición es falsa, entonces no hacer nada.



# Comparaciones más comunes

Operación	Resultado
5 > 9	False
5 < 9	True
5 >= 9	False
5 <= 9	True
5 == 9	False
5 != 9	True

El resultado de evaluar una condición es siempre del tipo **booleano**!

# Ejemplo Estructura Simple

- Encuentre las raíces de una ecuación cuadrática del tipo:

$$ax^2 + bx + c.$$

- Resolución:

- La fórmula para encontrar las raíces es

$$-b \pm (b^2 - 4ac)^{1/2} / (2a)$$

- Las raíces existen si y sólo si  $b^2 \geq 4ac$

# Ejemplo Estructura Simple: Pseudocódigo

- Entrada:  
a, b, c
- Salida:  
raiz1, raiz2 (si existen)
- Pseudocódigo:  
Recibir “a”, “b” y “c” de usuario  
Si  $b^2 \geq 4ac$ :  
     $raiz1 = -b - (b^2 - 4ac)^{1/2} / (2*a)$   
     $raiz2 = -b + (b^2 - 4ac)^{1/2} / (2*a)$   
Devolver raiz1, raiz2



# Ejemplo Estructura Simple: Código

```
a = float(input("Ingrese el valor de a: "))
b = float(input("Ingrese el valor de b: "))
c = float(input("Ingrese el valor de c: "))

if b**2 >= 4*a*c:
    raiz1 = ( -b + (b**2 - 4 * a * c) **0.5) / (2 * a)
    raiz2 = ( -b - (b**2 - 4 * a * c) **0.5) / (2 * a)
    print("Raiz 1: ", raiz1, "Raiz 2:", raiz2)
```

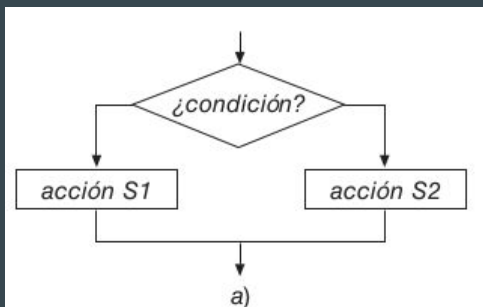
Caso 1:  $x^2 + 4x + 1$   
 $b^2 = 4^2 = 16$   
 $4ac = 4*1*1 = 4$   
 $16 > 4$

```
Ingrese valor de a: 1
Ingrese valor de b: 4
Ingrese valor de c: 1
Raiz 1: -7.0
Raiz 2: -1.0
```

Caso 2:  $x^2 + 2x + 4$   
 $b^2 = 2^2 = 4$   
 $4ac = 4*1*4 = 16$   
 $4 \nless 16$

```
Ingrese valor de a: 1
Ingrese valor de b: 2
Ingrese valor de c: 4
```

# Estructura Doble



*Pseudocódigo en español*

```
si <condicion> entonces  
    <accion S1>  
si_no  
    <accion S2>  
fin_si
```

*Pseudocódigo en inglés*

```
if <condicion> then  
    <accion S1>  
else  
    <accion S2>  
endif
```

- Nuevamente, existe una sola condición.
  - Si la condición es verdadera, ejecutar la acción S1.
  - Si la condición es falsa, ejecutar la acción S2.
- Las acciones alternativas se indican con la palabra **else**.

# Ejemplo Estructura Doble: pseudocódigo

- Entrada:  
a, b, c
- Salida:  
raiz1, raiz2 (si existen)
- Pseudocódigo:  
Recibir “a”, “b” y “c” de usuario  
Si  $b^2 \geq 4ac$ :  
     $raiz1 = -b - (b^2 - 4ac)^{1/2} / (2*a)$   
     $raiz2 = -b + (b^2 - 4ac)^{1/2} / (2*a)$   
    Devolver raiz1, raiz2  
Sino:  
    Devolver “No existen raices reales”

# Ejemplo Estructura Doble: Código

```
a = float(input("Ingrese el valor de a: "))
b = float(input("Ingrese el valor de b: "))
c = float(input("Ingrese el valor de c: "))

if b**2 >= 4*a*c:
    raiz1 = ( -b + (b**2 - 4 * a * c) **0.5) / (2 * a)
    raiz2 = ( -b - (b**2 - 4 * a * c) **0.5) / (2 * a)
    print("Raiz 1: ", raiz1, "Raiz 2:", raiz2)
else:
    print("No existen raices reales")
```

Caso 1:  $x^2 + 4x + 1$   
 $b^2 = 4^2 = 16$   
 $4ac = 4*1*1 = 4$   
 $16 > 4$

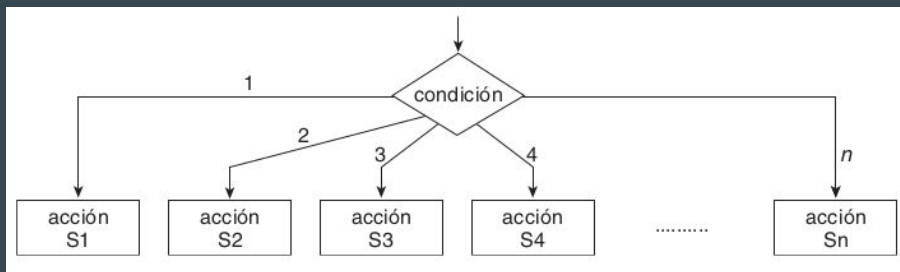
```
Ingrese valor de a: 1
Ingrese valor de b: 4
Ingrese valor de c: 1
Raiz 1:  -7.0
Raiz 2:  -1.0
```

Caso 2:  $x^2 + 2x + 4$   
 $b^2 = 2^2 = 4$   
 $4ac = 4*1*4 = 16$   
 $4 \nless 16$

```
Ingrese valor de a: 1
Ingrese valor de b: 2
Ingrese valor de c: 4
No existen raices reales
```

# Estructura Múltiple

- Hay casos en los que una misma condición puede dar múltiples resultados.
- Cada resultado tiene sus acciones asociadas.
- En Python, la forma de ejecutar este tipo de alternativas es usando la sentencia *elif*.
- Pueden concatenarse tantos elif como sean necesarios.



## Ejemplo Estructura Múltiple: Código

```
a = float(input("Ingrese el valor de a: "))
b = float(input("Ingrese el valor de b: "))
c = float(input("Ingrese el valor de c: "))

if b**2 > 4*a*c:
    raiz1 = ( -b + (b**2 - 4 * a * c) **0.5) / (2 * a)
    raiz2 = ( -b - (b**2 - 4 * a * c) **0.5) / (2 * a)
    print("Raiz 1: ", raiz1, "Raiz 2:", raiz2)
elif b**2 == 4*a*c:
    print("Raiz", -b)
else:
    print("No existen raices reales")
```

# Ejemplo Estructura Múltiple: Salidas

Caso 1:  $x^2 + 4x + 1$

$$b^2 = 4^2 = 16$$

$$4ac = 4 \cdot 1 \cdot 1 = 4$$

$$16 > 4$$

```
Ingrese valor de a: 1
Ingrese valor de b: 4
Ingrese valor de c: 1
Raiz 1:  -7.0
Raiz 2:  -1.0
```

Caso 2:  $x^2 + 2x + 1$

$$b^2 = 2^2 = 4$$

$$4ac = 4 \cdot 1 \cdot 1 = 4$$

$$4 = 4$$

```
Ingrese valor de a: 1
Ingrese valor de b: 2
Ingrese valor de c: 1
Raiz:  -2.0
```

Caso 3:  $x^2 + 2x + 4$

$$b^2 = 2^2 = 4$$

$$4ac = 4 \cdot 1 \cdot 4 = 16$$

$$4 \ngtr 16$$

```
Ingrese valor de a: 1
Ingrese valor de b: 2
Ingrese valor de c: 4
No existen raices reales
```

# Condiciones múltiples

- En una misma sentencia *if* puede indicarse más de una condición.
- Se puede requerir que todas las condiciones especificadas se tengan que cumplir:
  - si “aprobo parcial 1” **y** “aprobo parcial 2”:  
alumno = “regular”
- En otros casos, si se cumple una de las condiciones basta:
  - si “desaprobó parcial 1” **o** “desaprobó parcial 2”:  
recuperatorio = **True**



# Condiciones múltiples

Condicion1 and Condicion2		Condicion 2	
		True	False
Condicion 1	True	True	False
	False	False	False

Condicion1 or Condicion2		Condicion 2	
		True	False
Condicion 1	True	True	True
	False	True	False

# Condicionales Anidados

- Es posible incluir condicionales dentro de otro condicional.
- La práctica de incluir elementos de un tipo dentro de otros elementos del mismo tipo se denomina **anidar**.
- Los condicionales internos se evaluarán si y sólo si el condicional que los contiene se cumple.

# Ejemplo Condicionales Anidados

```
nota = float(input("Ingrese una nota: "))
regular = input("Aprobo la materia correlativa (Si/No): ")

if nota >= 8:
    if regular == "Si":
        print("Promociono")
    else:
        print("Guardar nota")

elif nota >= 6:
    print("Regular")

else:
    print("Libre")
```

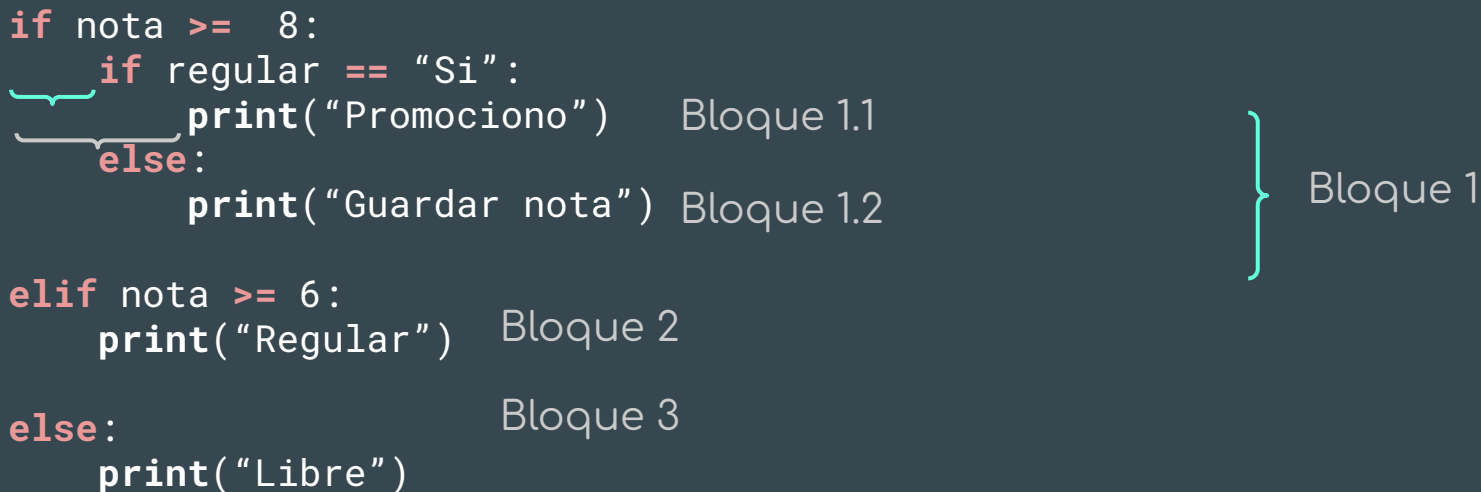
# Indentación

- Todas las líneas de código que se ejecutan como parte de un condicional se denominan **bloque de código**.
- Muchos lenguajes separan los bloques de código con llaves ( { y } ).
- Python, por su parte, utiliza **espacios en blanco** o **tabulaciones**.
- Todas las líneas que estén a un mismo nivel formarán parte de un mismo bloque.

# Indentación

```
nota = float(input("Ingrese una nota: "))  
regular = input("Aprobo la materia correlativa (Si/No): ")
```

```
if nota >= 8:  
    if regular == "Si":  
        print("Promociono")    Bloque 1.1  
    else:  
        print("Guardar nota")  Bloque 1.2  
  
elif nota >= 6:  
    print("Regular")    Bloque 2  
  
else:  
    print("Libre")      Bloque 3
```



Se puede usar cualquier tipo y cantidad de espacios en blanco, pero lo más común es usar 4 espacios en blanco.

# Ejemplo 1

- Determinar si un número par o impar.

# Ejemplo 1: Pseudocódigo

- Determinar si un número par o impar.
  - Entrada:  
numero
  - Salida:  
“par” o “impar”
  - Pseudocódigo:  
Recibir “numero” de usuario  
resto = numero%2  
Si resto == 0:  
    Devolver “par”  
sino:  
    Devolver “impar”

# Ejemplo 1: Código

- Determinar si un número par o impar.

```
numero = int(input("Ingrese una nota: "))
resto = numero % 2

if resto == 0:
    print("Par")
else:
    print("Impar")
```



## Ejemplo 2

- Se desea calcular el sueldo neto semanal de un trabajador que cobra por hora.
  - Si la persona registra más de 40 horas en esa semana, las horas extras se pagan un 50% más.
  - Los impuestos varían según el sueldo total:
    - sueldo menor a 300.000 no paga impuesto
    - sueldo entre 300.000 y 400.000 paga un 10% de impuesto
    - sueldo mayor a 400.000 paga un 18% de impuestos.
- Se pide como entrada al usuario la cantidad total de horas y el sueldo por hora del empleado.

## Ejemplo 2: Pseudocódigo

- Entrada:

sueldo hora

cantidad\_horas

- Salida:

sueldo neto

- Pseudocódigo:

recibir sueldo\_hora y cantidad\_horas

sueldo\_bruto = 0

si cantidad\_horas <= 40:

    sueldo\_bruto = sueldo\_hora \* cantidad\_horas

sino:

    horas\_extras = cantidad\_horas - 40

    sueldo\_bruto = sueldo\_hora \* 40 + horas\_extras \* sueldo\_hora \* 1.5

si 300000 < sueldo\_bruto <= 400000:

    sueldo\_netto = sueldo\_bruto - sueldo\_bruto \* 0.10

sino, si sueldo\_bruto > 400000:

    sueldo\_netto = sueldo\_bruto - sueldo\_bruto \* 0.18

sino:

    sueldo\_netto = sueldo\_bruto

Devolver sueldo\_netto

## Ejemplo 2: Código

```
sueldo_hora = float(input("Ingrese el sueldo por hora: "))
cantidad_horas = int(input("Ingrese la cantidad de horas trabajadas: "))

# Calculo del sueldo bruto
if cantidad_horas <= 40:
    sueldo_bruto = sueldo_hora * cantidad_horas
else:
    extras = cantidad_horas - 40
    sueldo_bruto = sueldo_hora * cantidad_horas + extras * sueldo_hora * 1.5

# Calculo impuestos
if sueldo_bruto > 300000 and sueldo_bruto <= 400000:
    sueldo_netto = sueldo_bruto - sueldo_bruto * 0.10 # sueldo_bruto*=0.90
elif sueldo_bruto > 400000:
    sueldo_netto = sueldo_bruto - sueldo_bruto * 0.18 # sueldo_bruto*=0.82
else:
    sueldo_netto = sueldo_bruto

print("El sueldo neto es de: ", sueldo_netto)
```

## Ejemplo 3

- Dadas las condiciones indicadas en la convocatoria, chequear si un estudiante puede inscribirse o no a las becas Manuel Belgrano

### ¿Quiénes pueden aplicar?

- Argentinos nativos o naturalizados con DNI.
- Estudiantes regulares de universidades nacionales y provinciales de gestión pública que ingresen en el primer semestre de 2024 o que estén cursando al momento de la inscripción alguna de las carreras universitarias de pregrado o grado definidas por el Programa. [Listado de carreras alcanzadas](#).
- Los estudiantes ingresantes deben tener entre 18 y 30 años de edad; los cursantes, hasta 35 años.
- Quedan exceptuadas del requisito de edad las personas con algún tipo de discapacidad, las personas con hijos menores de edad en hogares monoparentales y quienes pertenezcan a pueblos originarios.

<https://www.argentina.gob.ar/educacion/becas/becas-manuel-belgrano/1era-convocatoria-2022-programa-nacional-de-becas-estrategicas-manuel-belgrano>

## Ejemplo 3: Pseudocódigo

- Entrada:
  - edad
  - condicion
  - excepcion
- Salida:
  - SI / NO

◦ Pseudocódigo:  
recibir edad, condicion, excepcion

si excepcion es "Si":

    Devolver "Califica"

si edad < 18:

    Devolver "No califica"

si edad < 30 y condicion == "ingresante":

    Devolver "Califica"

si edad < 35 y condicion == "cursante":

    Devolver "Califica"

else:

    Devolver "No califica"

## Ejemplo 3: Código

```
edad = int(input("Ingrese la edad: "))
condicion = input("Ingrese si es ingresante o cursante: ")
excepcion = input("Ingrese Si si es una persona con discapacidad o de pueblo originario")

if excepcion == "Si":
    print("Califica")
elif edad < 18:
    print("No califica")
elif edad < 30 and condicion == "ingresante":
    print("Califica")
elif edad < 35 and condicion == "cursante":
    print("Califica")
else:
    print("No califica")
```