

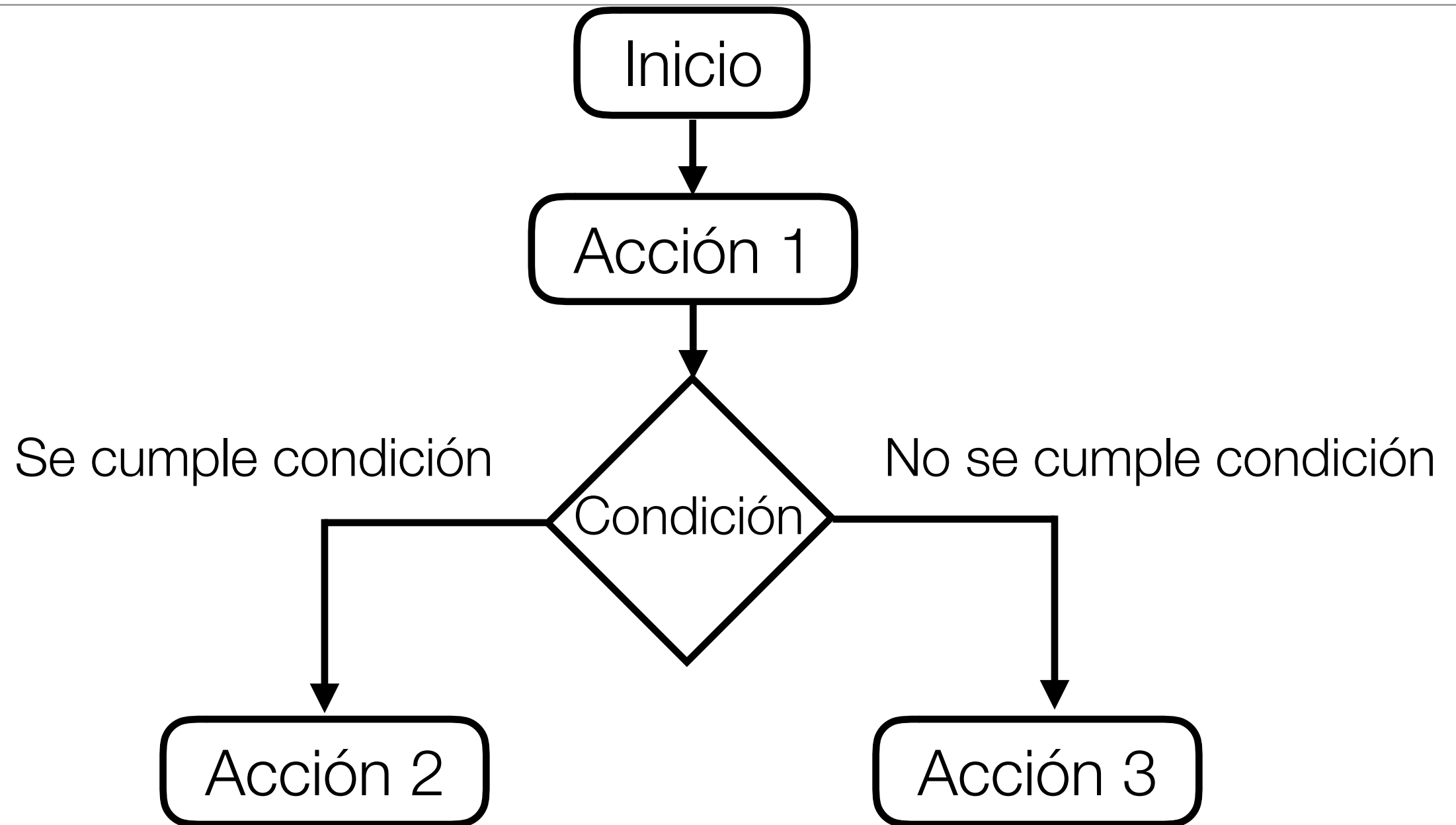
Algorítmica y Programación

Enero - Mayo 2020

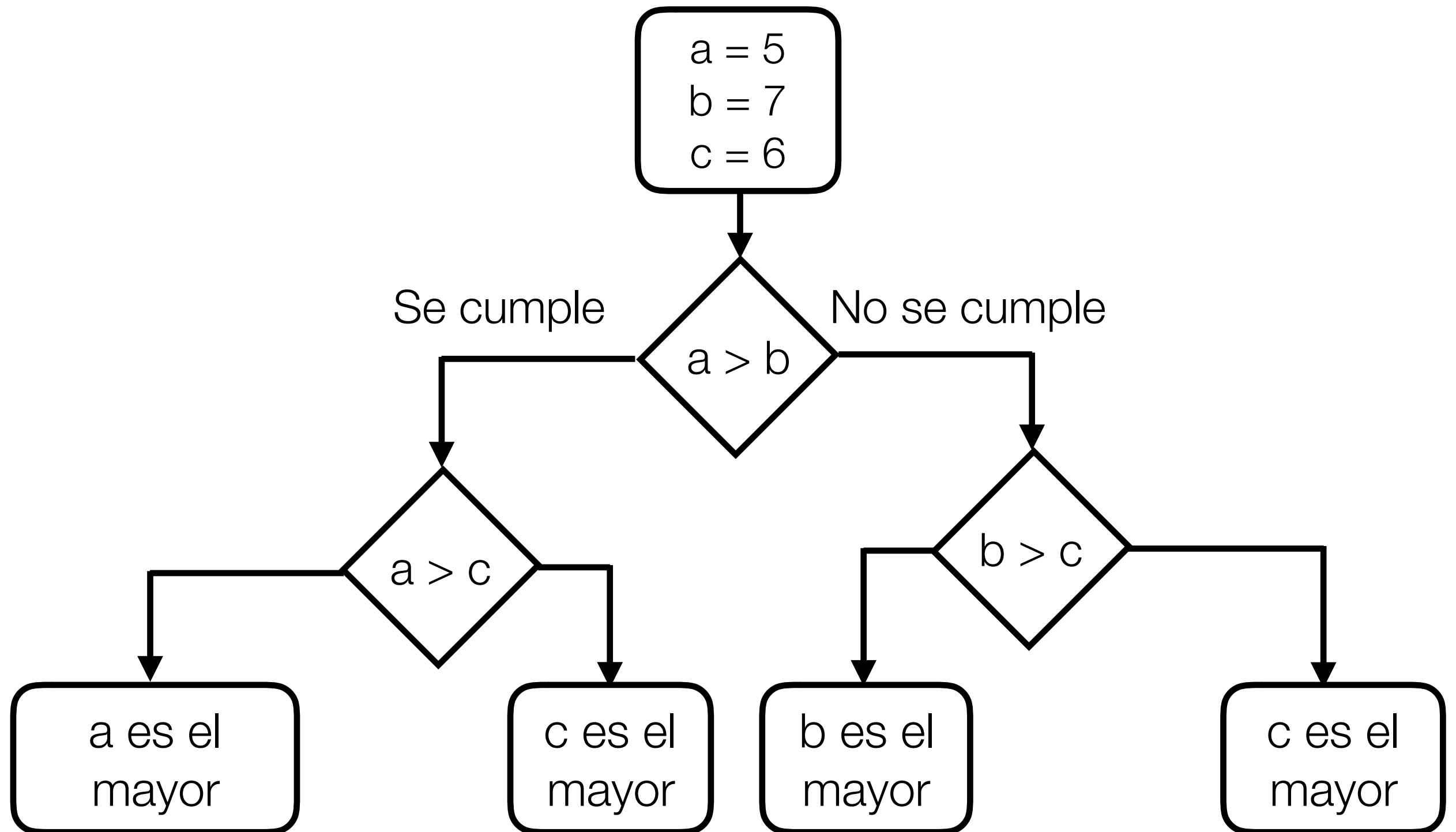
Dr. Iván S. Razo Zapata
(ivan.razo@itam.mx)

Python estructuras de control

If - else



If - else



If - else I Paso de parámetros

1

```
import sys
```

```
a = int(sys.argv[1])  
b = int(sys.argv[2])  
c = int(sys.argv[3])
```

```
if a > b:  
    if a > c:  
        print("a es el mayor de los tres")  
    else:  
        print("c es el mayor de los tres")  
else:  
    if b > c:  
        print("b es el mayor de los tres")  
    else:  
        print("c es el mayor de los tres")
```

2

Guardar archivo

Nombre: Ejemplo01

Ruta: Escritorio

Sobre sys

<https://docs.python.org/3.7/library/sys.html>

Python » English » 3.7.6 » Documentation » The Python Standard Library » Python Runtime Services »

Previous topic
Python Runtime Services

Next topic
sysconfig — Provide access to Python's configuration information

This Page
[Report a Bug](#)
[Show Source](#)

sys — System-specific parameters and functions

This module provides access to some variables used or maintained by the interpreter and to functions that interact strongly with the interpreter. It is always available.

sys.abiflags

On POSIX systems where Python was built with the standard `configure` script, this contains the ABI flags as specified by [PEP 3149](#).

New in version 3.2.

sys.argv

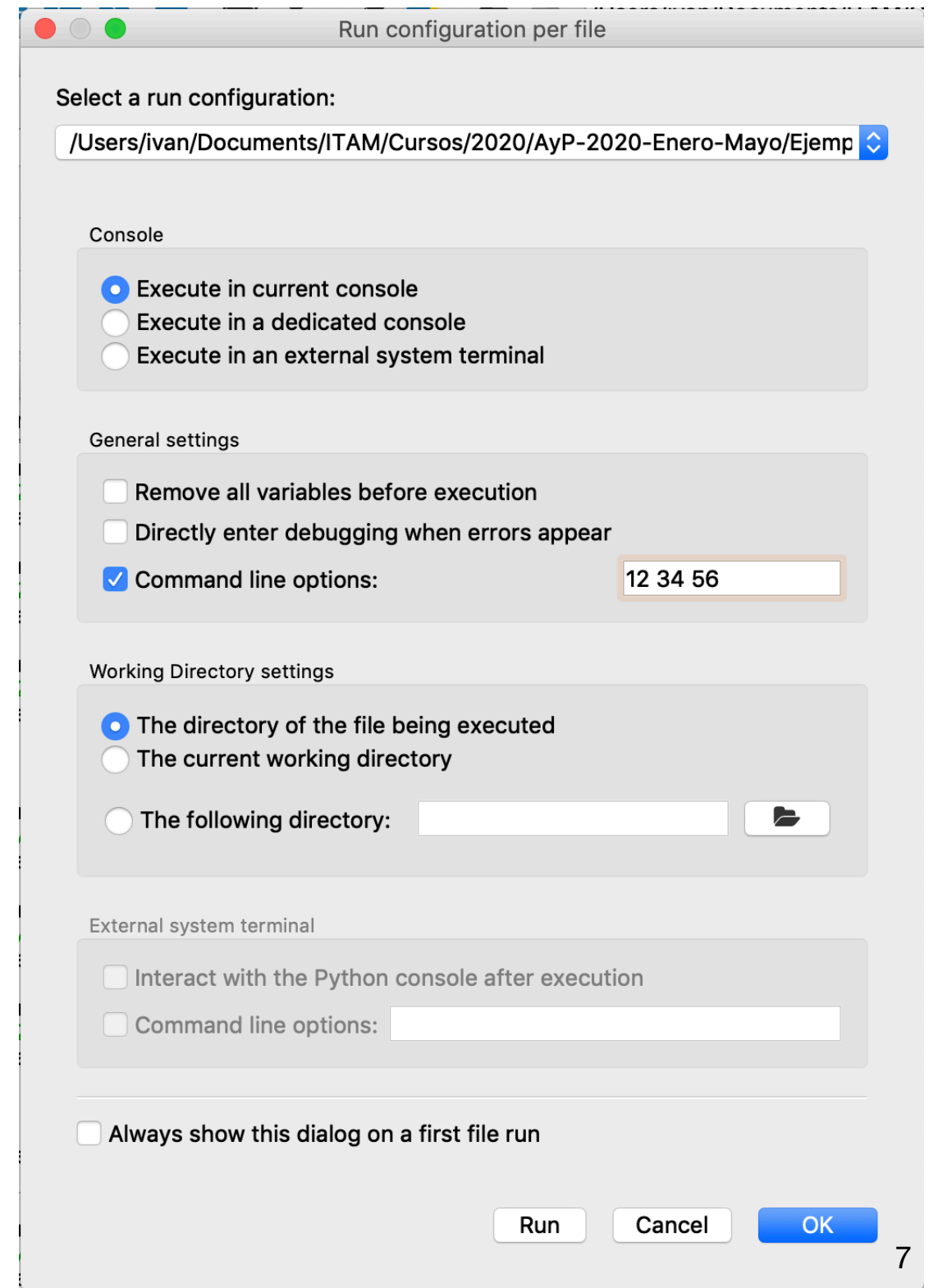
The list of command line arguments passed to a Python script. `argv[0]` is the script name (it is operating system dependent whether this is a full pathname or not). If the command was executed using the `-c` command line option to the interpreter, `argv[0]` is set to the string `'-c'`. If no script name was passed to the Python interpreter, `argv[0]` is the empty string.

To loop over the standard input, or the list of files given on the command line, see the [fileinput](#) module.

Note: On Unix, command line arguments are passed by bytes from OS. Python decodes them with filesystem encoding and “surrogateescape” error handler. When you need original bytes, you can get it by `[os.fsencode(arg) for arg in sys.argv]`.

If - else I Paso de parámetros

3.a Configuración para ejecutar



If - else I Paso de parámetros

3.b Consola IPython (Spyder)

```
runfile('nombre.py', wdir='ruta', args = '17 26 18')
```

Nombre asignado

Ruta donde se guardo el archivo

3.c Terminal Windows

Ir a la ruta donde se guardo el archivo

```
python Ejemplo01.py 22 45 78  
c es el mayor de los tres
```


If - else

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

- Ej1 - Indique el precio de un boleto de cine dependiendo de la edad del cliente. El programa debe aceptar como argumento la edad del cliente y calcular el precio de la siguiente manera: (a) Precio normal: 60 pesos; (b) Menores de 12 años: 40% de descuento; (c) Mayores de 60 años: 45% de descuento.
- Ej2 - Acepte tres parámetros enteros y los imprima de mayor a menor.
- Ej3 - Acepte un valor **x** y calcule **f(x)**
$$f(x) = \begin{cases} x + 3 & \text{if } x \leq 0 \\ x^2 + 2x & \text{if } x > 0 \end{cases}$$

elif

```
if y < 0:  
    print("El valor de y es negativo")  
elif y == 0:  
    print("El valor de y es cero")  
elif 0 < y < 10:  
    print("El valor de y es positivo pero menor a diez")  
else:  
    print("El valor de y es igual a diez o mayor")
```

If-else en una linea

```
resultado = "Mayor o igual a diez" if y >= 10 else "Menor a diez"  
print(resultado)
```

If - else

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

#Solución Ej1

```
import sys
edad = int(sys.argv[1])
precio = 60
if edad < 12:
    precio *= 0.6
elif edad > 60:
    precio *= 0.55

print("El precio es : ",precio)
```

If - else

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

Solucion Ej2

```
import sys
a = int(sys.argv[1])
b = int(sys.argv[2])
c = int(sys.argv[3])

if a > b:
    if a > c:
        print(a, b, c) if b > c else print(a, c, b)
    else:
        print(c, a, b) if a > b else print(c, b, a)
else:
    if b > c:
        print(b, a, c) if a > c else print(b, c, a)
    else:
        print(c, a, b) if a > b else print(c, b, a)
```

If - else

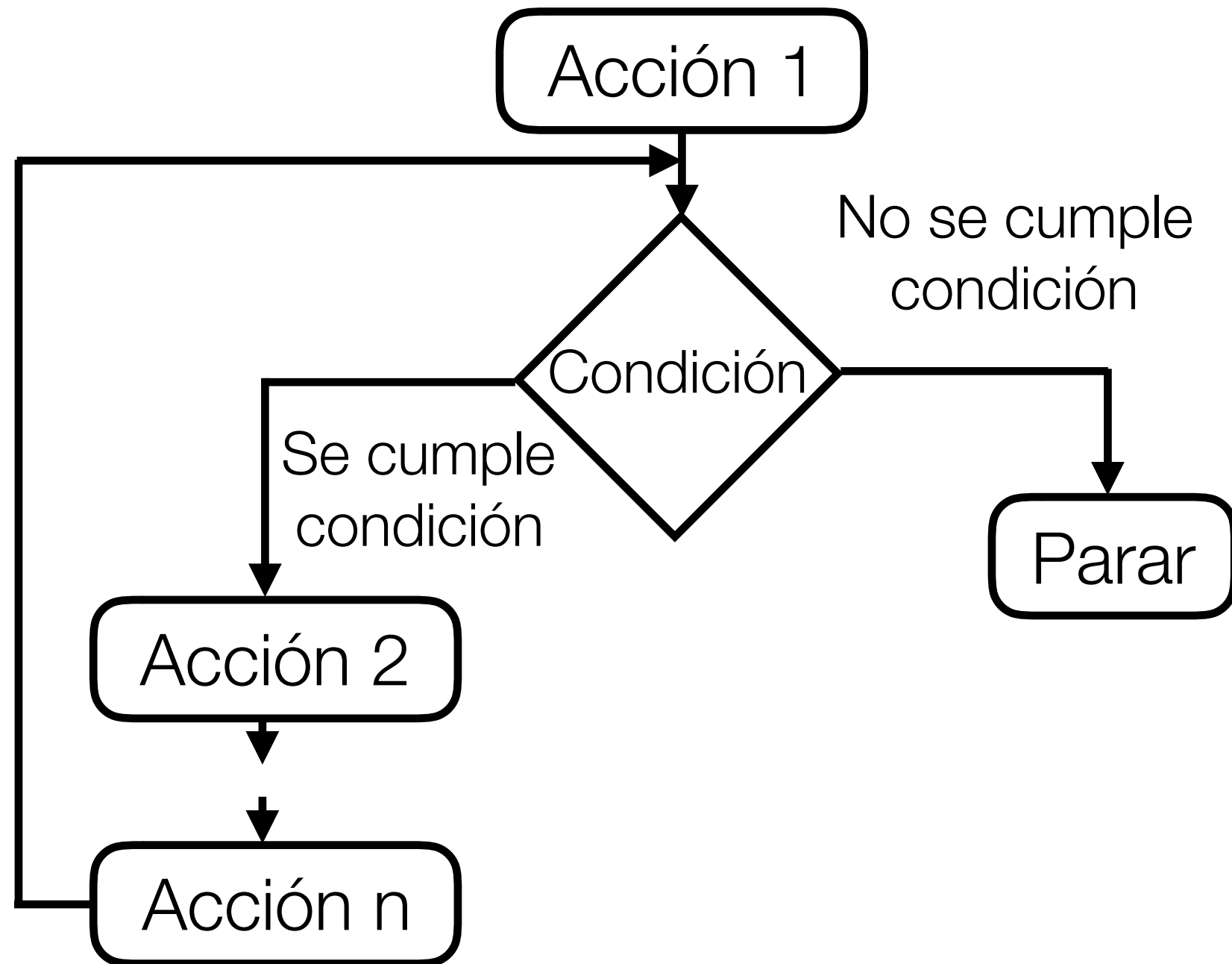
Ejercicios: Diseñar algoritmos y escribir programas correspondientes

#Solución Ej3

```
import sys
x = int(sys.argv[1])
fx = (x + 3) if x <= 0 else (x**2 + 2*x)
print(fx)
```

while

- Ejecutar acciones **mientras** que una **condición** se cumpla (i.e. mientras sea verdadera)



while - Ejemplo

```
edad = 0
```

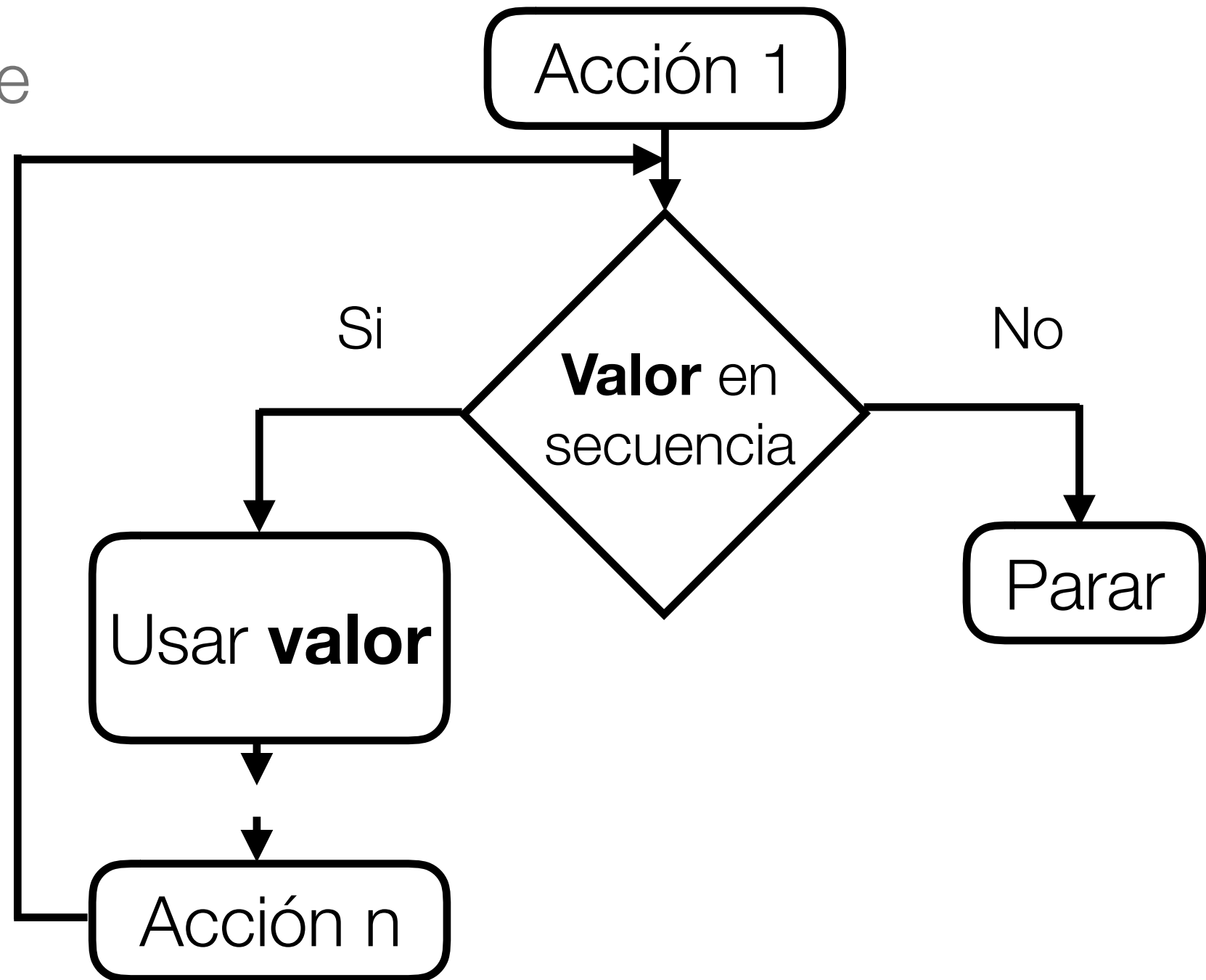
```
while edad < 18:
```

```
    edad = edad + 1
```

```
    print("Por ahora tu edad es ", edad)
```


for

- Iterar / navegar sobre una secuencia de valores (lista o tupla)



for - Ejemplo

```
secuencia = [1, 2, 3, 4, 5]
```

```
total = 0
```

```
for valor in secuencia:
```

```
    total += valor
```

```
print("El total es ", total)
```

while

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

- Ej1. Imprima números enteros de dos en dos del 50 al 0
- Ej2. Acepte un valor entero (N) como parámetro y calcule el factorial de dicho entero
 - $N! = 1 * 2 * 3 * \dots * (N-2) * (N-1) * N$
- Ej3. Acepte un valor entero (N) como parámetro e imprima los N números de Fibonacci
 - 0, 1, 1, 2, 3, 5, 8, 13, ...,

while

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

#Solucion Ej1 - Imprima números enteros de dos en dos del 50 al 0

```
n = 50
while n >= 0:
    print(n)
    n -= 2
```

while

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

#Solucion Ej2 (factorial de N) - Opción 1

```
import sys
N = int(sys.argv[1])

Factorial = 1
Contador = 1

while Contador <= N:
    Factorial = Factorial * Contador
    Contador += 1

resultado = 'El Factorial de ' + str(N) + ' es ' + str(Factorial)
print(resultado)
```

while

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

#Solucion Ej2 (factorial de N) - Opción 2

```
Factorial = 1
while N > 1:
    Factorial = Factorial * N
    N -= 1
print(Factorial)
```

while

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

#Solucion Ej3 - Imprimir N números de Fibonacci

```
import sys
N = int(sys.argv[1])
print("Numeros de Fibonacci")

primero = 0
segundo = 1
print(primero)
print(segundo)
Contador = 2
while Contador <= N:
    Fibonnaci_num = primero + segundo
    print(Fibonnaci_num)
    primero = segundo
    segundo = Fibonnaci_num
    Contador += 1
```

for

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

- Ej1. Dada una secuencia con 10 números, indique
 - 1. Promedio de los números
 - 2. Número mayor y posición en la secuencia
 - 3. Número menor y posición en la secuencia
- Ej2. Tome un entero como parámetro y produzca su tabla de multiplicar del 1 al 10

for

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

#Solución Ej1

```
secuencia = [7, 2, 3, 4, 1, 61, 7, 88, 0, 10]
Mayor      = secuencia[0]
posMayor   = 0
Menor      = secuencia[0]
posMenor   = 0
contador   = 0
promedio   = 0
```

```
for valor in secuencia:
    if valor > Mayor:
        Mayor = valor
        posMayor = contador

    if valor < Menor:
        Menor = valor
        posMenor = contador
    promedio += (valor / 10)
    contador += 1
```

```
sPromedio = "El promedio en la secuencia es " + str(promedio)
sMayor     = "El mayor es " + str(Mayor) + " su posicion es " + str(posMayor)
sMenor     = "El menor es " + str(Menor) + " su posicion es " + str(posMenor)
```

```
print(sPromedio)
print(sMayor)
print(sMenor)
```

for

Ejercicios: Diseñar algoritmos y escribir programas correspondientes

#Solucion Ej2 - Tabla de multiplicar para un numero N

```
import sys
N = int(sys.argv[1])
secuencia = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
for i in secuencia:
    print(N, 'x', i, '=', N*i)
```