



M13: FASE 2 Exercicis introducció Python

1. Calcula la multiplicació i la suma de dos números. Donats dos nombres de tipus integer, retorna el seu producte només si el producte és igual o inferior a 1000, si no, retorna la seva suma.

Multiplicacion:

```
In [3]: x = int(input("introduce primer numero: "))
        y = int(input("introduce segundo numero: "))

        suma = x + y
        mult = x * y

        if suma <= 1000 :
            print(suma)
        else :
            print(mult)

introduce primer numero: 1000
introduce segundo numero: 1000
1000000
```

Suma:

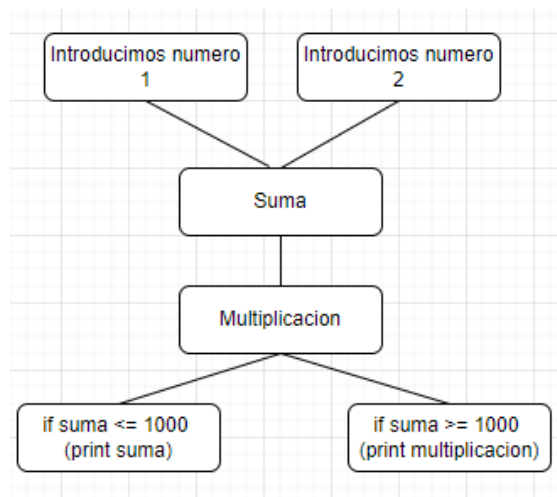
```
In [26]: x = int(input("introduce primer numero: "))
        y = int(input("introduce segundo numero: "))

        suma = x + y
        mult = x * y

        if suma <= 1000 :
            print(suma)
        else :
            print(mult)

introduce primer numero: 100
introduce segundo numero: 100
200
```

Diagrama de flujo:



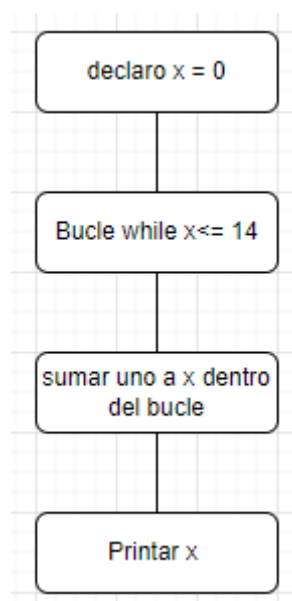
2. Imprimeix per pantalla els primers 15 números naturals fent servir un bucle while.

Bucle while:

```
In [4]: x = 0  
  
while(x <= 14):  
    x += 1  
    print(x, end=" ")
```

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Diagrama de flujo:



3. Imprimeix per pantalla els primers 15 números naturals fent servir un bucle for.

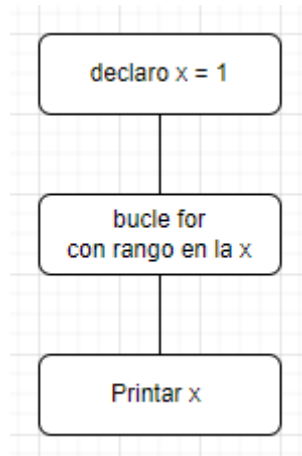
Bucle for:

```
In [5]: x = 1
```

```
for x in range(1,16):  
    print(x, end=" ")
```

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
```

Diagrama de flujo:



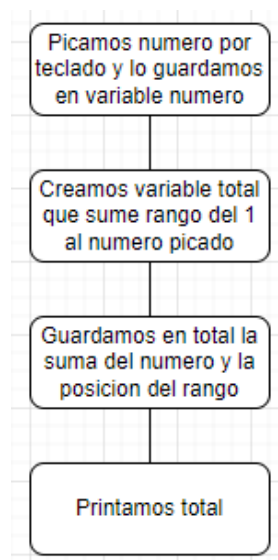
4. Calcula la suma d'1 a un número donat que l'usuari pot introduir per pantalla. Per exemple, si l'usuari ha introduït 10 la sortida ha de ser 55 (1+2+3+4+5+6+7+8+9+10)

Suma:

```
In [6]: numero = int(input("introduce primer numero: "))  
  
total = sum(range(1, numero))|  
total += numero  
print(total)
```

```
introduce primer numero: 10  
55
```

Diagrama de flujo:



5. Escriu un programa que calcula la taula de multiplicar d'un número que l'usuari pot introduir per pantalla.

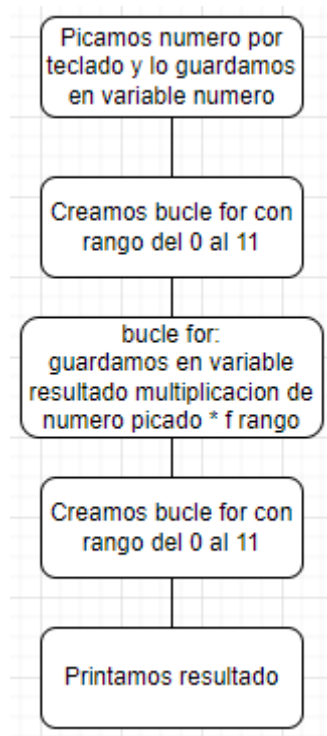
Tabla de multiplicar:

```
In [10]: numero = int(input("introduce primer numero: "))

for f in range(0,11):
    resultado = numero * f
    print(numero , "x", f, "=", resultado)

introduce primer numero: 5
5 x 0 = 0
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
```

Diagrama de flujo:

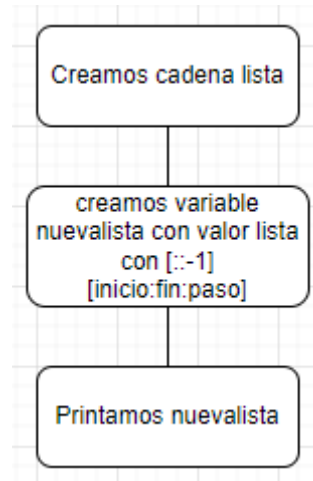


6. Donada la llista `list = [20, 60, 80, 100]`, imprimeix-la per pantalla de manera invertida.

Lista invertida:

```
In [18]: lista = [20, 60, 80, 100]
nuevalista = lista[::-1]
print(nuevalista)
[100, 80, 60, 20]
```

Diagrama de flujo:

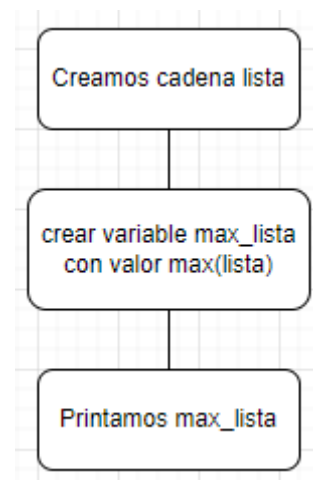


7. Troba (imprimint per pantalla) l'element més gran d'una llista de noms creada per tu.

Valor más grande:

```
In [24]: lista = ["Fernando", "Lucia", "Carlos", "Aitor", "Luis Miguel"]
max_lista = max(lista)
print(max_lista)
Luis Miguel
```

Diagrama de flujo:



8. Comptabilitza la quantitat de lletres, dígit i símbols especials de la llista donada.

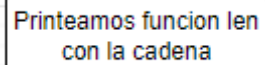
→ string = [&#Pr@j3ctE#D4M#&]

Contabilizar:

```
In [25]: print(len("&#Pr@j3ctE#D4M#&"))
```

16

Diagrama de flujo:



```
graph TD; A[Printeamos funcion len con la cadena]
```

9. Ara exporta aquests codis, des del Jupyter i fes un versionat amb git/github, que com a mínim inclogui 3 branques del framework d'Scrum (GitFlow). (per ex: main, features i release). Afegeix captures que mostrin el versionat.