



**UAP**

The logo consists of the letters 'UAP' in a bold, sans-serif font. The letter 'A' is stylized with three horizontal wavy lines passing through its center. The entire logo is centered within a white square frame, which is itself centered on a solid dark blue background.

# ALGORITMOS Y ESTRUCTURAS DE DATOS

RECURSIVIDAD II – BÚSQUEDA LINEAL

# CONCEPTO

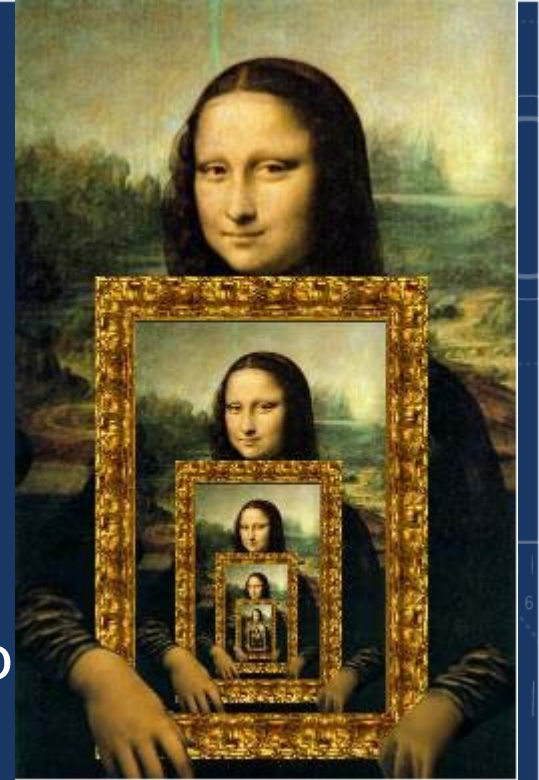
## RECURSIVIDAD (otra forma de verlo)

Forma de iterar en donde una función se llama a sí misma.

También hay que tener cuidado de no llegar a un bucle infinito

Una función recursiva siempre tendrá:

- Al menos una llamada a sí misma, sino no es recursiva
- Al menos un if con el caso base que corta la recursividad
- Python tiene un límite de 1000 llamados recursivos de profundidad



```

1 # Recursividad II
2 def fact_1():
3     print("Entrando en 1")
4     r = 1
5     print("Saliendo de 1 => " + str(r))
6     return r
7
8 def fact_2():
9     print("Entrando en 2")
10    r = 2 * fact_1()
11    print("Saliendo de 2 => " + str(r))
12    return r
13
14
15 def fact_3():
16     print("Entrando en 3")
17     r = 3 * fact_2()
18     print("Saliendo de 3 => " + str(r))
19     return r
20
21 def fact_4():
22     print("Entrando en 4")
23     r = 4 * fact_3()
24     print("Saliendo de 4 => " + str(r))
25     return r
26
27 def fact_5():
28     print("Entrando en 5")
29     r = 5 * fact_4()
30     print("Saliendo de 5 => " + str(r))
31     return r
32
33 resultado = fact_5()
34 print("El factorial de 5 es " + str(resultado))
35

```

```
Entrando en 5
Entrando en 4
Entrando en 3
Entrando en 2
Entrando en 1
Saliendo de 1 => 1
Saliendo de 2 => 2
Saliendo de 3 => 6
Saliendo de 4 => 24
Saliendo de 5 => 120
El factorial de 5 es 120
```

-----

```
36 print("-----")
37 def factorial(x):
38     if x < 1:
39         return 1 #caso base
40     else:
41         r = x * factorial(x-1)
42         return r
43
44 print("El factorial de 5 es " + str(factorial(5)))
```

# CONCEPTO

## Búsqueda lineal

Algoritmo que se aplica a un array con el fin de buscar un elemento en particular.

Se recorren cada uno de los elementos y se va comprando si es el buscado.

```
1 curso = ["Ana", "Juan", "Valeria", "Maria", "Pedro"]
2
3 for pos, elemento in enumerate(curso):
4     print("Chequeando el elemento: " + str(pos) + " valor: " + elemento)
5     if elemento == "Maria":
6         print("    Encontrado en posición: " + str(pos) + "!!!!!!")
7         #break
```



```
Chequeando el elemento: 0 valor: Ana  
Chequeando el elemento: 1 valor: Juan  
Chequeando el elemento: 2 valor: Valeria  
Chequeando el elemento: 3 valor: Maria  
    Encontrado en posición: 3!!!!  
Chequeando el elemento: 4 valor: Pedro
```

```
1 curso = ["Ana", "Juan", "Valeria", "Maria", "Pedro"]
2
3 esta = False
4 for pos, elemento in enumerate(curso):
5     if elemento == "Maria":
6         esta = True
7         break
8
9 print("Maria está?: " + str(esta) + " => posición: " + str(pos))
10 #Maria está?: True => posición: 3
```

```
1 paises = {
2     1: 'Argentina',
3     2: 'Uruguay',
4     3: 'Chile',
5     4: 'Bolivia',
6     5: 'Paraguay'
7 }
8
9 for codigo, pais in paises.items():
10     if codigo == 3:
11         print("País encontrado: " + pais)
12     if pais == "Paraguay":
13         print("País encontrado con codigo: " + str(codigo))
14
15 #País encontrado: Chile
16 #País encontrado con codigo: 5
```

```
1 curso = ["Ana", "Juan", "Valeria", "Maria", "Pedro"]
2
3 indice = 0
4 while(indice < len(curso)):
5     print("Procesando elemento " + curso[indice] + " en posición " + str(indice))
6     if curso[indice] == "Maria":
7         print("    Está")
8     indice = indice + 1
9 ""
10 Procesando elemento Ana en posición 0
11 Procesando elemento Juan en posición 1
12 Procesando elemento Valeria en posición 2
13 Procesando elemento Maria en posición 3
14     Está
15 Procesando elemento Pedro en posición 4
16 ""
```

```
1 matriz = [  
2     [3, -2, 1, 12],  
3     [1, 3, 1, -4],  
4     [2, 2, -4, 6]  
5 ]  
6  
7 #Primer for para recorrer filas  
8 for fila in matriz:  
9     #segundo for para recorrer columnas por cada fila  
10    for elemento in fila:  
11        print("Elemento: " + str(elemento))  
12
```

```
1 matriz = [  
2     [3, -2, 1, 12],  
3     [1, 3, 1, -4],  
4     [2, 2, -4, 6]  
5 ]  
6 ""  
7 Ejercicios:  
8 1) Sumatoria de cada fila  
9 2) Sumatoria de cada columna  
10 3) Calcular los valores de x, y y z del siguiente sistema de ecuaciones lineales  
11 con 3 incógnitas:  
12 (https://www.matesfacil.com/BAC/ejercicios-resueltos-CRAMER.html)  
13  
14  $3x - 2y + z = 12$   
15  $x + 3y + z = -4$   
16  $2x + 2y - 4z = 6$   
17 La solución es  $(x, y, z) = (3, -2, -1)$ .  
18 ""
```



**UAP**.EDU.AR | @**UAP**ARGENTINA