

Complejidad Algorítmica

Prof. Robert Zubieta

INTRODUCCION A PYTHON

Declarar Variables

```
In [ ]: i = 5  
f = 3.7  
s = "Robert"
```

Imprimir Variables

```
In [ ]: print("Integer = ", i)  
print("Flotante = ", f)  
print("String = ", s)
```

```
Integer = 5  
Flotante = 3.7  
String = Robert
```

Crear una secuencia de numeros del 0 al 5 e imprimirlos

```
In [ ]: x = range(6) #Numeros consecutivos: incrementan en 1  
for n in x:  
    print(n)
```

```
0  
1  
2  
3  
4  
5
```

Crear una secuencia de numeros del 0 al 19, incrementados en 2 e imprimirlos

```
In [ ]: x = range(3, 20, 2) #Numeros incrementan en 2  
for n in x:  
    print(n)
```

```
3  
5  
7  
9  
11  
13  
15  
17  
19
```

Crear un ciclo Repetitivo con FOR

```
In [ ]: for i in range(1,7):  
    print("Linea ", i)
```

Linea 1
Linea 2
Linea 3
Linea 4
Linea 5
Linea 6

Crear un ciclo Repetitivo con WHILE

```
In [ ]: i=0
while i<7:
    print("Linea ", i)
    i = i + 1
```

Linea 0
Linea 1
Linea 2
Linea 3
Linea 4
Linea 5
Linea 6

Crear una LISTA

```
In [ ]: lista = []
for i in range(10):
    lista.append(i)
print(lista)
```

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

Crear una FUNCION

```
In [ ]: # Funcion para hallar el Doble de un numero
def funcion(n):
    return 2*n

print("Resultado de Funcion: ", funcion(5))
```

Resultado de Funcion: 10

Analisis de Tiempo de dos Funciones

```
In [ ]: import time

# Funcion Iterativa
def factorial(n):
    respuesta = 1
    while n>1:
        respuesta = respuesta * n
        n = n-1
    return respuesta

# Funcion Recursiva
def factorial_r(n):
    if n==1:
        return 1
    return n*factorial_r(n-1)
```

```
if __name__ == '__main__':  
  
    n = 2000  
    #n = 2000 #Limite, >=20000 => Kernel Dead  
    #n = 200000      => Kernel Dead  
  
    comienzo = time.time()  
    factorial(n)  
    final = time.time()  
    print("factorial: ", final - comienzo)  
  
    print()  
  
    comienzo = time.time()  
    factorial_r(n)  
    final = time.time()  
    print("factorial_r: ", final - comienzo)
```

factorial: 0.0019986629486083984

factorial_r: 0.008997201919555664