

# </ Ayudantía 4: Ciclos 2

/>

} /> [

Rolando Rojas

# </ Temas a tratar

{01}

Repaso [Ciclo while]

{02}

Ciclos while anidados

{03}

Ejercicio 1: Sumatoria

{04}

Ejercicio 2: Fibonacci

{05}

Ejercicio 3: Taylor

{06}

Tarea

## </ Ciclo while (repaso)

El ciclo **while** se utiliza cuando deseas que el bloque de código se ejecute mientras una condición sea **verdadera**. En otras palabras, el ciclo continuará ejecutándose mientras la condición sea cierta.

```
while (Condición):  
    ...código a repetir
```

# </ Ciclos while anidados

## Ciclo Anidado

¿Qué es?



Es un ciclo que depende de otro ciclo

¿Para que  
sirve?

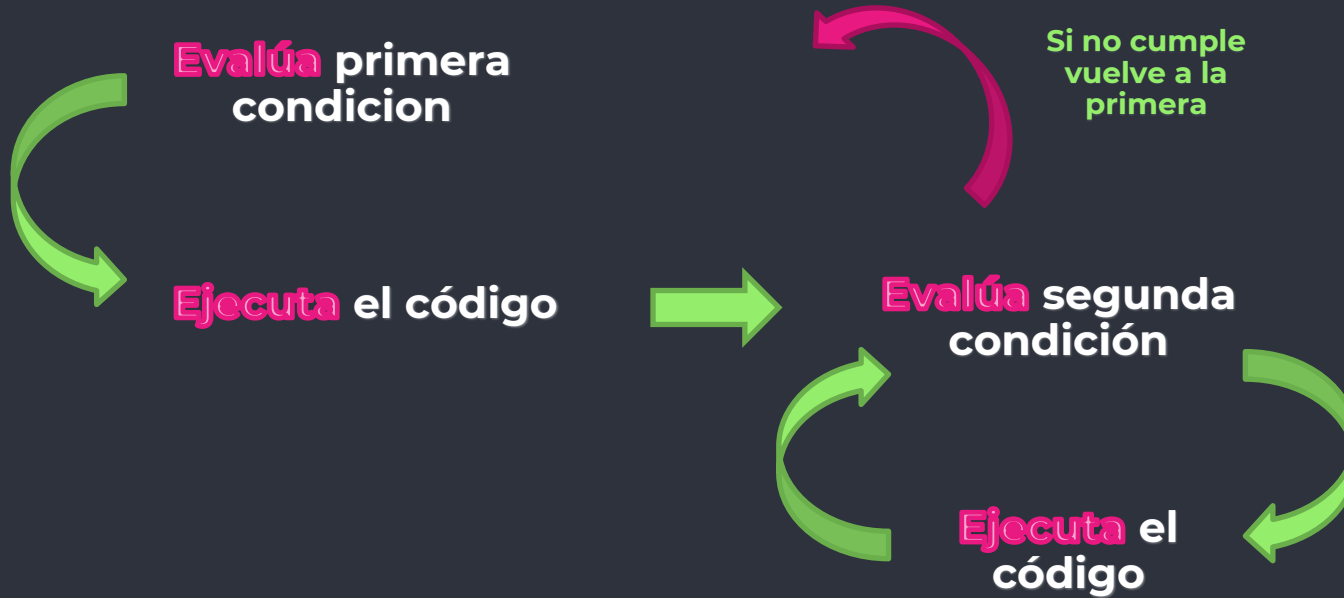


Sirve para repetir procesos ya iterativos

**"Anidado"** en programación es la práctica de  
incorporar procedimientos dentro de otros

1 0 1 1   0 1 1   0 1   1 0 1 1 0 0 1   1 0   1 1 0 1 1   0 1 1   0 1   1 1 0 1 1 0   1 1 0 1 1 1   1 1 0 1

# </ Ciclos while anidados



1 0 1 1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 1 1 1 0 1

## </ Ejercicio 1: Sumatoria

*Cree un programa que pida un número y calcule la **sumatoria** de todos los números desde el cero hasta el número que se ingresó anteriormente.*

$$\sum_{x=0}^{\infty} x = 1 + 2 + 3 \dots$$

## Ejercicio:

Cree un programa que pida un número y calcule la **sumatoria** de todos los números desde el cero hasta el número que se ingresó anteriormente.

```
#Sumatoria de todos los números desde el cero hasta el número que se ingrese
limite_superior=int(input("Ingrese límite superior sumatoria:"))
i=0
suma=0
while i<=limite_superior:
    suma+=i
    i+=1
print(suma)
```

```
Ingrese límite superior sumatoria:5
15
```

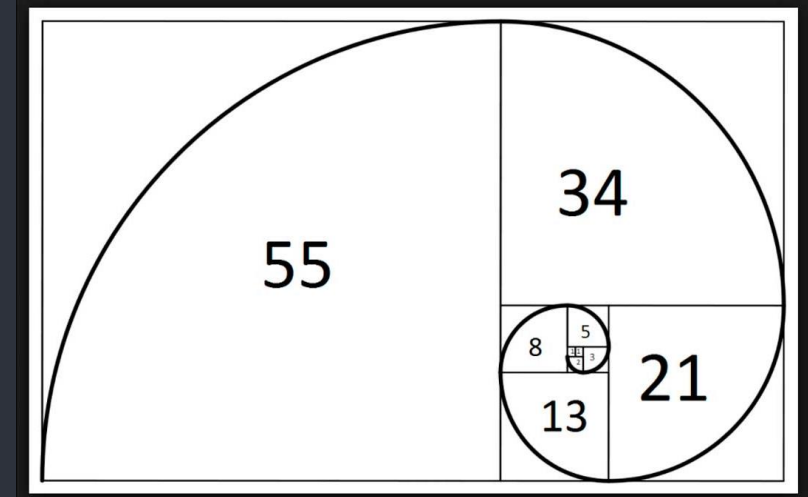
Sumatoria		
Límite Superior	i	suma
5	0	0
	1	
		1
	2	
		3
	3	
		6
	4	
		10
	5	
		15
	6	
Output:	15	

## </ Ejercicio 2: Fibonacci

Cree un programa que pida un número (cantidad de términos a calcular) y en base a eso calcule la **Serie de Fibonacci**

(Cada número corresponde a la suma de los **dos números anteriores**)

**Ejemplo:** 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144...





**Ejercicio:**  
Cree un programa que pida un número (cantidad de términos a calcular) y en base a eso calcule la **Serie de Fibonacci**

```
#Serie de Fibonacci:
n=int(input("Ingrese cantidad de términos:"))
suma_anterior = 1
suma_que_sigue = 1
serie_total = '1'
i = 1
while i < n :
    nueva_suma = suma_anterior + suma_que_sigue
    serie_total = serie_total + ',' + str(nueva_suma)
    suma_anterior = suma_que_sigue
    suma_que_sigue = nueva_suma
    i += 1
print(serie_total)
```

Ingrese cantidad de términos:5  
1,2,3,5,8

Serie de Fibonacci					
n	suma_anterior	suma_que_sigue	i	nueva_suma	serie_total
5	1	1	1		"1"
				2	
					"1,2"
	1				
		2			
			2		
				3	
					"1,2,3"
	2				
		3			
			3		
				5	
					"1,2,3,5"
	3				
		5			
			4		
				8	
					"1,2,3,5,8"
			5		
Output:	"1,2,3,5,8"				

## </ Ejercicio 3: Taylor

Cree un programa que pida un número (cantidad de términos a calcular  $\rightarrow n$ ) y el valor de  $x$ . En base a eso calcule la **Serie de Taylor** para  $e^x$

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} \dots$$

*\*Tip: ¡recordar ciclos anidados!*

**Ejercicio:**  
 Cree un programa que pida un número (cantidad de términos a calcular) y en base a eso calcule la **Serie de Taylor** para  $e^x$

```
x=int(input('Ingrese el valor de x:'))
n=int(input('Ingrese cantidad de términos:'))
i=0
j=0
total=0

while i<=n:
    factorial=1
    j=1
    #haremos un while para buscar el factorial
    while j<=i:
        factorial*=j
        j+=1
    #Ahora que tenemos el factorial (divisor) podemos sumar la fracción al total
    total+=(x**i/factorial)
    i+=1

print(total)
```

Ingrese el valor de x:2  
 Ingrese cantidad de términos:1000  
 7.389056098930649

Serie de Taylor $e^x$					
x	n	i	j	total	factorial
2					
	2	0		0	
			1		1
					1
			2		
				1	
		1			
			1		1
					1
			2		
				3	
		2			
			1		1
					1
			2		
					2
			3		
				5	
		3			
Output:	5				

1 0 1 1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1 1 0 1 1

1 0 1

Otra vez no  
hay tarea



1 0 1 1 0 1 1 0 1 1 0 1 1 0 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 1 0 1 1 0 1 1 0 1 1 1 0 1

# </ Ayudantía 4: Ciclos 2

¿Dudas?

rolando.rojass@usm.cl

/>

} /> [

Rolando Rojas