



ITERACIÓN: `while`

10145 - FUNDAMENTOS DE PROGRAMACIÓN PARA INGENIERÍA

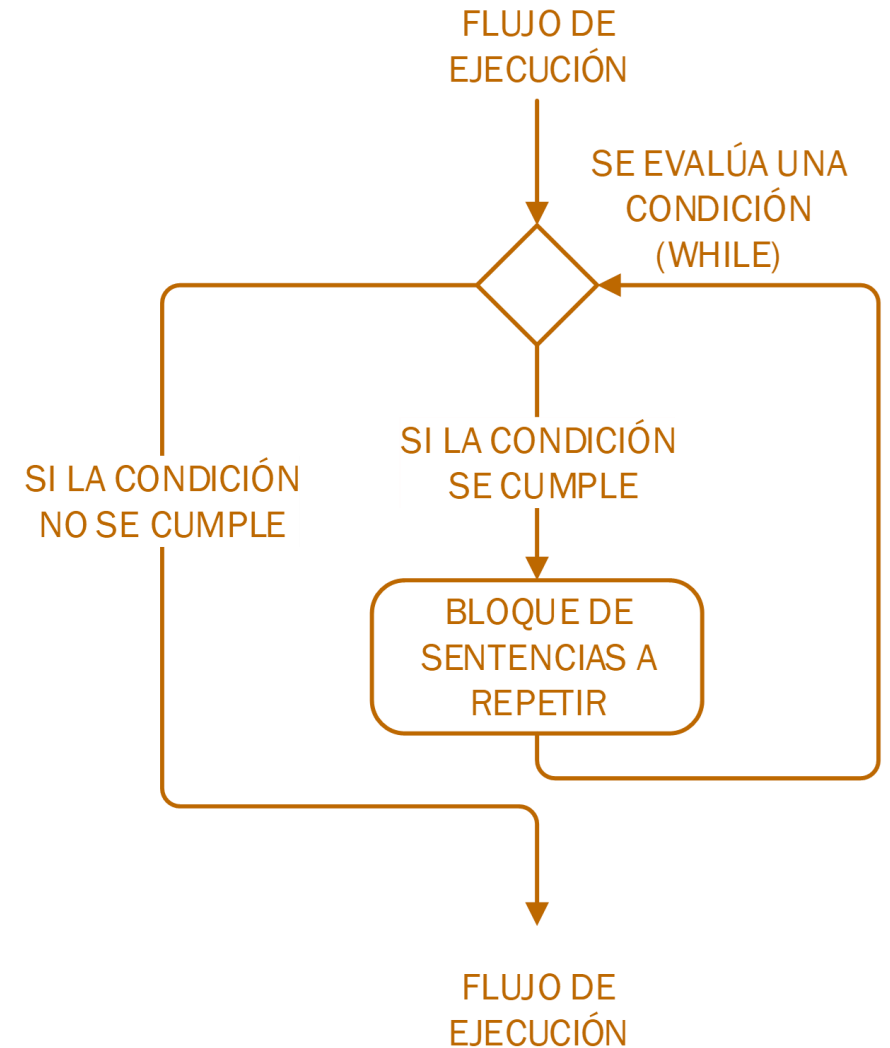
10110 – FUNDAMENTOS DE COMPUTACIÓN Y PROGRAMACIÓN



RESUMEN DE CONTENIDOS

ITERACIÓN

- Flujo de una sentencia iterativa (**while**)
 - A diferencia de un **if**, cuando un bloque de sentencias condicionado por un **while** alcanza su fin **la condición se vuelve a evaluar**
 - Esto implica que el código puede **repetirse**
 - Pero en algún momento se debe **encontrar una forma de salir** del **while**





SENTENCIA `while`

- Estructura de la sentencia `while`

```
<Sentencias previas>
```

```
while <condición>:
```

```
    # Se ejecuta si la condición se cumple
```

```
    <Bloque de sentencias a repetir>
```

```
<Sentencias después del ciclo>
```

TRAZAS

- Analicemos el programa con **numero = 5**

```
numero = 5
```

```
i = 0
```

```
suma = 0
```

```
while i <= numero :
```

```
    suma = suma + i
```

```
    i = i + 1
```

```
# SALIDA
```

```
print ("La suma de los primeros", numero)
```

```
print ("números, es: ", suma)
```

Variable\Valor	Iteración						
	Inicio	1	2	3	4	5	6
numero	5	5	5	5	5	5	5
suma	0	0	1	3	6	10	15
i	0	1	2	3	4	5	6

TRAZAS

- Analicemos el programa con **numero = 5**

```
numero = 5
```

```
i = 0
```

```
suma = 0
```

```
while i <= numero :
```

```
    suma = suma + i
```

```
    i = i + 1
```

Variable\Valor	Iteración						
	Inicio	1	2	3	4	5	6
numero	5	5	5	5	5	5	5
suma	0	0	1	3	6	10	15
i	0	1	2	3	4	5	6

Al proceso de revisar manualmente el comportamiento del código en base a las variables involucradas, se le conoce como realizar una **traza del programa**

Y es una herramienta vital para entender que es lo que hace un código



TAUTOLOGÍAS

- ¿Qué pasa si colocamos una tautología como condición?

```
i = 0
```

```
while i < 10 :
```

```
    print('Debo practicar programación')
```

- El mensaje 'Debo practicar programación' se repetirá por siempre
- El programa seguirá iterando hasta que **forcemos una detención**
 - En estos casos, es posible detener la ejecución de un programa con el comando **ctrl + c**
- Si tenemos una sentencia **while** de la cuál es imposible salir durante una ejecución normal del programa significa que hemos creado **ciclo infinito**



EJERCICIOS



EJERCICIOS PROPUESTOS

- Construya un programa en Python que calcule la multiplicación de dos números enteros positivos usando solamente operadores de suma y resta. Por ejemplo:

$$- 5 * 4 = 5 + 5 + 5 + 5 = 4 + 4 + 4 + 4 + 4 = 20$$



EJERCICIOS PROPUESTOS

1. Modifique el código anterior para que calcule la división entera y el resto de números enteros positivos
2. Modifique el código anterior para que calcule la multiplicación utilizando cualquier par de números enteros (positivos o negativos)
3. Modifique el código anterior para que, en caso de que alguno de los números sea flotante, el programa lo informe y pida nuevamente el número



TAREAS PARA TRABAJO AUTÓNOMO

1. Revisar el apunte:
 - **for-in** en Google Colab (Disponible en: https://github.com/PROGRA-FING-USACH/2023-1/blob/main/Lecturas/05_iteracion_-_for-in.ipynb)
2. Revisar la guía 3, disponible a partir del próximo lunes en Replit



¿CONSULTAS?