

Práctica 4:

1. Imprimir los primeros 20 números naturales.
2. Imprimir la tabla de multiplicar del número 5
3. Imprima los números de -10 a -1.
4. Dada la siguiente secuencia de números:

```
numeros = [12, 75, 150, 180, 145, 525, 50]
```

imprimir los números divisibles por 5 menores a 150.

5. Imprimir los primeros 10 valores de la secuencia de Fibonacci. La secuencia de Fibonacci es una serie de números en la cual los dos primeros números son 0 y 1, y el siguiente número se corresponde a la suma de los dos números anteriores. Resultado esperado:

```
0 1 1 2 3 5 8 13 21 34
```

6. Imprimir el valor factorial del número 5 usando un bucle. El valor factorial (símbolo: !) se obtiene de multiplicar todos los números enteros desde el número elegido hasta 1. Resultado esperado: 120
7. Explique el resultado de los siguientes programas

```
a. lista = []  
for j in lista:  
    print(j)
```

```
b. i = 1  
for i in [1, 2, 3]:  
    print(i)  
print(i)
```

8. Resuelva los siguientes ejercicios
 - a. Calcular el cuadrado de los primeros 10 números enteros.
 - b. Calcular la suma de los números enteros entre 0 y 100 inclusive.
 - c. Calcular la suma de los números pares menores a 100. ¿Cuántos números pares hay menores a 100?
 - d. Calcular la suma de los números impares menores a 100. ¿Cuántos números impares hay menores a 100?
9. ¿Cuál es el problema asociado al siguiente programa? No hace falta ejecutarlo para responder esta pregunta.

```
x = 0  
while x < 5:  
    print(x)
```

10. Escriba un programa que dada una secuencia de números y un valor de umbral vaya sumando los números de la secuencia hasta que la suma alcance el umbral. Utilice **break** para terminar la ejecución del bucle cuando la suma alcance el umbral.

Resultados esperados:

```

umbral = 21
valores = [3, 5, 4, 4, 5, 5, 3, 5, 2, 7]
# suma -> 21

umbral = 34
valores = [3, 5, 4, 4, 5, 5, 3, 5, 2, 7]
# suma -> 34

umbral = 100
valores = [3, 5, 4, 4, 5, 5, 3, 5, 2, 7]
# suma -> 43

```

11. Escriba un programa que dada una secuencia numérica compute la suma de los números pares. Utilice un bucle `while` y la sentencia `continue` para saltar los casos donde el número no sea par.

Resultados esperados:

```

valores = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
# suma -> 30

valores = [1, 4, 7, 10]
# suma -> 14

```

12. Escriba un programa que solicite un número entero al usuario y compute la suma de todos los números naturales menores a él. Este programa debe ser interactivo. Es decir, el programa solicita un número al usuario, devuelve la suma, y solicita un nuevo número. Esto continúa así hasta que el usuario ingresa "salir", determinando que el programa debe terminar. Utilice un bucle `while` para resolver este problema. Tenga en cuenta la sentencia `break` para determinar la interrupción del bucle. Ejemplos:

```

Ingrese un numero o 'salir' para terminar: 10
La suma es 45
Ingrese un numero o 'salir' para terminar: 50
La suma es 1225
Ingrese un numero o 'salir' para terminar: salir

```

13. Una mañana ponés un billete en la vereda al lado del Monumento a la Bandera. A partir de ahí, cada día vas y duplicás la cantidad de billetes, apilándolos prolijamente. ¿Cuánto tiempo pasa antes de que la pila de billetes sea más alta que la del Monumento? Considere los siguientes valores para comenzar a resolver el problema:

```

billete_grosor = 0.11 * 0.001 # grosor de un billete en metros
altura_monumento = 70        # altura en metros
billetes_n = 1
dia = 1

```

Utilice un bucle `while` para resolver el problema.

14. Escriba un programa reciba dos números como parámetros, y compute cuántos múltiplos del primero hay, que sean menores que el segundo.
- Implementarla utilizando un ciclo `for`, desde el primer número hasta el segundo.
 - Implementarla utilizando un ciclo `while`, que multiplique el primer número hasta que sea mayor que el segundo.
 - Comparar ambas implementaciones: ¿Cuál es más clara? ¿Cuál realiza menos operaciones?
-