

## Práctico 1 - Repaso de Conceptos Básicos - Bucles y Condicionales

En este práctico, deberá resolver todos los ejercicios de modo artesanal, sin utilizar comandos que ya dé el propio lenguaje de Python, como `reverse`, `count`, `split`, `zip`, `shuffle`, `replace`, `sort`, o cualquier otro tipo parecido, etc.

Los comandos que SI puede utilizar son `range`, `len`, corchetes de posición de listas y strings.

Sigue siendo lo más importante y foco de la asignatura desarrollar y/o mejorar vuestra capacidad de razonamiento.

Este práctico hará fuerte hincapié en el uso de bucles.

Aquellos ejercicios marcados con una estrella, si se requieren. Servirán para un repaso, o para asegurar el nivel mínimo de razonamiento que se debe poseer antes de empezar esta asignatura como tal.

### Condicionales

1. Pedirle al alumno, la nota que se sacó en el último parcial. Determinar si el alumno aprobó o no ese parcial, teniendo en cuenta que debía sacar mayor o igual al 60 % para poder aprobar.
2. Pedirle al alumno tres notas, calcular el promedio. Determinar si aprobó solo para regularizar, o si la nota le sirve para promocionar. Regular es ente 60 % y menor a 75 %. Promocionado es un promedio mayor o igual a 75 %.
3. ★ Dada una lista de días de la semana, Lunes, Martes, Miércoles, etc. Pedir al usuario un número de día, 1, 2, 3, etc e indicarle entonces, en qué día se encuentra. Ingresa 1, el programa le dirá Lunes. Etc.
4. Pedirle dos números al usuario, *a* y *b*. Si son ambos iguales, mostrar un mensaje, son ambos iguales. Si *a* es positivo o menor a *b*, mostrar un mensaje que indique que o bien *a* era positivo o bien era menor a *b*.
5. Ingresar una letra. Indicar si es una vocal o no.
6. ★ Ingresar una letra, indicar si es una consonante o no.

### Bucles con Condicionales sencillos

7. Dados los siguientes valores: 4, 7, 1, 3, 9, 8, 2. Almacenarlos de algún modo apropiado en el código. Luego, mostrar uno por línea, siempre que sean menores a 7.
8. Dada la siguiente palabra *Hobbit* mostrar una letra por línea, siempre que no sea una vocal.
9. Dada una lista de 6 valores numéricos, ingresados por el usuario, mostrar en pantalla solo aquellos números que son positivos y distintos de 3.
10. ★ Dada una lista de valores enteros al azar entre 1 y 6, con al menos 10 valores en total, reemplazar los números que sean 1 por 0.

11. Pida una frase al usuario. ¿Cuántos signos de admiración ingresó en la frase el usuario? mostrar el resultado.

## Bucles

12. Pedir una frase al usuario y una posición, mostrar en pantalla la letra en dicha posición. Recordar que el usuario verá la primer letra, como la 1, y que el programa las inicia en realidad desde cero.
13. Dada la siguiente frase *Recórcholis Batman!*:
- a) Mostrar en pantalla la primera, segunda y quinta letra de la misma.
  - b) Mostrar una letra por línea.
  - c) Contar cuántas letras tiene la frase y mostrar el resultado.
  - d) Mostrar en pantalla la última letra.
14. Pedir una frase al usuario.
- a) Mostrar en pantalla la primera, segunda y quinta letra de la misma.
  - b) Mostrar una letra por línea.
  - c) Contar cuántas letras tiene la frase y mostrar el resultado.
  - d) Mostrar en pantalla la última letra.
15. Guardar 5 valores en una lista, inventados. Reemplazar cualquier 5 que hubiese por un 0.
16. Guardar una lista con los siguientes valores 5, 3, 1, 7, 8, 34, 3. Sumar todos los valores y mostrar el resultado. (*ojo! recorrer la lista a tales fines, no realizar la suma mentalmente nosotros*)
17. ★ Pedir una frase al usuario y una posición, y un símbolo. Reemplazar en dicha posición, la letra original de la frase, por el símbolo. Mostrar el resultado final.
18. Pedir una frase al usuario, y reemplazar todas sus vocales por el símbolo numeral #.
19. ★ Contar la cantidad de palabras en una frase ingresada por el usuario, sin utilizar *split*. ¿Qué diferencia una palabra de otra, en una frase cualquiera?
20. Pedir una frase al usuario y mostrar las letras de la primer palabra nada más. No utilizar *split*.
21. Indique si el siguiente código es correcto o no. Justifique.

---

```
for k in range(len(f)):  
    c = c + 1  
    k = 2 * c  
print(c)
```

---

22. Dada una lista de valores enteros, cualquiera, que obtenga por ingreso de parte del usuario, o por llenado de valores aleatorios con random, determine, muestre los resultados:
- a) ¿Cuántos números positivos tiene?
  - b) ¿Cuántos números mayores a 5 tiene?
  - c) ¿Cuántos números pares tiene?
  - d) ¿Cuántos números mayores o iguales a 6 tiene y menores a 7?
  - e) ¿Cuántos números menores a cero o negativos tiene?
  - f) La suma de todos sus valores.
  - g) El promedio de todos sus valores.
  - h) La suma de los valores de la primer mitad de la lista.
23. Pedir al usuario una palabra. Pedir una posición y mostrar la letra de dicha posición en pantalla. Continuar así, hasta que el usuario ingrese la letra *q* para salir del programa. ¿Qué tipo de bucle me veo forzado a utilizar en este caso?
24. Dada una lista de 100 valores aleatorios, determinar cuantos números fueron entre 10 y 45.

## Variado

25. Crear una función que determine si un número es par o no.
26. Crear una función que retorne la suma de todos los valores dentro de una lista de 8 números que usted puede inventar.
27. Dada una frase ingresada por el usuario, mostrar en pantalla, las letras de la misma, una por línea. Luego mostrar solo las letras en posiciones pares, una por línea. Y finalmente, mostrar las letras en posiciones impares, una por línea.
28. ★ Dada una frase ingresada por el usuario, determinar cuantas palabras de longitud par contiene. (*La frase “hola Frodo, y Bilbo?” contiene 1 sola palabra de longitud par*)
29. ★ Obtener un cuento corto, desde un archivo de texto llamado cuento.txt (*Busque algún párrafo o cuento corto clásico y guardelo en un archivo de texto con dicho nombre*). Realizar el siguiente análisis y mostrar en pantalla un informe de:
- a) Cuantas palabras contiene en total.
  - b) Cuantas frases contiene en total. (*¿Cómo deberá distinguir una frase de otra?*)
  - c) Cuantos nombres propios o nombres de lugares hay en total.
  - d) El total de vocales que contiene ese cuento.
  - e) Cuantas palabras de longitud par existen.
  - f) Cuantas palabras de longitud impar contiene.

30. ★ Pida al usuario una frase e inviertala. Implemente al menos 3 modos diferentes de lograrlo, sin utilizar el comando *reverse*.
31. Implementar una función que reciba dos vectores y retorne el producto escalar de ambos.
32. ★ Pedir números al usuario y almacenarlos en una lista, hasta que el usuario ingrese la letra *q* para salir.
33. ★ Implementar una función llamada *signos* que reciba un vector de números enteros, y retorne un vector de 1 y -1, según si el número en cada posición es positivo o negativo.
34. ★ ¿Qué diferencia hay entre listas y arreglos? Explique.
35. Haga un listado de comandos básicos de la librería Numpy, relacionados a crear vectores y matrices. El resto de las operaciones de Numpy que no sean específicos para la creación de arrays, no serán utilizados. Se deberán utilizar bucles para manipular los valores de las matrices.
36. ★ Dada una matriz de 10 x 10 retornar la suma de todos sus valores. Utilizar Numpy.
37. Dada una matriz de 6 x 6 valores numéricos, implementada de dos modos, con listas anidadas, y con arrays de Numpy, realice un programa que sume los elementos de la diagonal y muestre el resultado. Los valores iniciales deben ser puestos aleatoriamente. (Repase cómo utilizar random)
38. ★ Generar la siguiente matriz e imprimirla en pantalla, utilizando Numpy. Además se deberá almacenar en un archivo de texto llamado *matriz.txt* sin utilizar métodos de Numpy, es decir sin utilizar comandos como *savetofile* y del estilo, deberá resolverlo artesanalmente, solo con lo esencial, condicionales, bucles, variables:

---

```
1 0 0 0 0 0
0 2 0 0 0 0
0 0 3 0 0 0
0 0 0 4 0 0
0 0 0 0 5 0
0 0 0 0 0 6
```

---

## Resumen - Tarea a entregar aparte

39. Realice manuscrito en su cuaderno, una síntesis de los tipos de bucles que existen en Python, formas de usarlos (for, while) y las diferencias entre cada uno. Debe haber 3 formas de usar los bucles al menos: dos modos en que se usa el bucle for, un while y sus diferencias con el bucle for, y cómo haría un while equivalente a un bucle for. Deberá subirlo al campus. Si son varias páginas, arme un archivo PDF y suba un link.
40. ¿Para qué sirve un condicional o estructura de decisión if? Invente un pequeño ejemplo de un mini programa utilizando un condicional sencillo.

41. ¿Puedo combinar condicionales con bucles? Por ejemplo, es posible tener un condicional dentro de un bucle? Justifique y ejemplifique. ¿Y un bucle, dentro de un condicional?
42. Dé ejemplos de las condiciones válidas para los condicionales. ¿Qué son los operadores lógicos, cómo se usan para armar condiciones?
43. ¿Cómo distingo cuando debo usar un bucle del tipo while, y no podré usar uno del tipo for?
44. ¿Qué tipo de for debo usar, por posiciones o por valores, si quiero modificar los valores de una lista, o arreglo, o frase? Dé un ejemplo, para una lista de 4 valores numéricos o para una palabra.
45. ¿Qué es un acumulador? ¿Qué es un contador? ¿En qué difieren? Dé un ejemplo pequeño de código de cada uno, de cómo los usaría usted.
46. ¿Qué diferencia hay entre una estructura de decisión simple de una estructura de decisión múltiple? justifique. Dé un ejemplo.
47. ¿Cuál es la diferencia entre usar solo if, if seguido de elif, e if seguido de else? Proporcione ejemplos para cada caso.
48. ¿Qué significa *anidamiento de sentencias if* y cuál es su utilidad? Proporcione un ejemplo mostrando su utilización.