

Guía de ejercicios

Con el fin de iniciar el desarrollo de nuestras habilidades en programación con el lenguaje **Python**, se deja en este documento varios ejercicios que deben realizarse con dicho lenguaje.

- Ejercicio 1:

Crear un programa que le pida al usuario cualquier palabra, secuencia de números u oración para que luego el programa imprima lo mismo pero con una “a” al principio y una “a” al final.

- Ejercicio 2:

Elabore un programa que le pida al usuario 4 números reales (use float()) y luego el programa debe imprimir el promedio de ellos.

- Ejercicio 3:

Realice un programa que le pida una cadena de caracteres (cualquier cosa) al usuario y que después muestre lo que ingresó el usuario pero al revés.

- Ejercicio 4:

Crear un programa que reciba una sucesión de 0s y 1s y luego determine si la cantidad 0s y 1s es par o impar.

- **Ejercicio 5:**

Elaborar un programa que le pida un número entero n, luego pida n veces al usuario que ingrese texto (por ejemplo un nombre), luego de terminar de pedir esos datos, imprima los datos recogidos pero en el orden inverso a como se recibieron.

- **Ejercicio 6:**

Realice un programa que pida al usuario un número real (flotante) para luego imprimir su raíz cuadrada. Si el resultado tiene decimales, limite la cantidad de los mismos a 2.

- **Ejercicio 7:**

Hacer un programa que reciba una string (cadena de caracteres) y detecte si es palindromo. (que se pueda leer igual de izquierda a derecha y viceversa).

- **Ejercicio 8:**

Crear una calculadora de operaciones aritméticas, el usuario debe ingresar dos números y debe elegir cuál operación efectuar para que luego se muestra en la salida el resultado. La interfaz se deja a su elección al igual que el orden de los datos de entrada.

- **Ejercicio 9:**

Realizar un programa en el cual el usuario luego de ingresar un número en sistema binario, se le muestre el mismo en sistema decimal.

- **Ejercicio 10:**

El usuario ingresa una cadena de caracteres (string) donde es seguro que dentro de ella habrá un número en sistema binario, si resulta ser que

el número es par, entonces el programa deberá imprimirlo en sistema decimal, en caso contrario se imprime el string ingresado sin el número binario.

El string que el usuario va a ingresar, podrá tener letras y cualquier símbolo de puntuación, pero en cuanto a números sólo tendrá los 0s y los 1s correspondiente al número binario.

- **Ejercicio 11:**

Dada una lista de 5 palabras dadas por el usuario, crea otra lista que contenga sólo aquellas palabras que tienen más de 5 letras. Imprime la nueva lista.

- **Ejercicio 12:**

Dado un diccionario de nombres y direcciones (creelo en el código, al menos 5 pares), encuentra la dirección de una persona específica utilizando la clave correspondiente. Pide al usuario que ingrese el nombre y luego imprime la dirección

- **Ejercicio 13:**

Dada una lista de 5 palabras dadas por el usuario, crea otra lista que contenga sólo aquellas palabras que tengan más de 3 consonantes. Imprime la nueva lista.

- **Ejercicio 14:**

Un número se considera "tímido" si sus dígitos suman exactamente 10. Dado un entero positivo n , devuelve el n -ésimo número "tímido". Por ejemplo, dado 1, deberías devolver 19. Dado 2, deberías devolver 28.

- **Ejercicio 15:**

Dada una lista de números y un número k , determine si hay tres entradas en la matriz que suman el número k especificado. Por ejemplo, dado $[20, 303, 3, 4, 25]$ y $k = 49$, devuelve true porque $20 + 4 + 25 = 49$.

Instructor: - **Fidel Serpa**

- **Ejercicio 16:**

Dados n números, encuentre el máximo común denominador entre ellos. Por ejemplo, dados los números $[42, 56, 14]$, retorna 14

- **Ejercicio 17:**

Dada una colección de intervalos, encuentre la cantidad mínima de intervalos que necesita eliminar para que el resto de los intervalos no se superpongan.

Los intervalos pueden "tocarse", como $[0, 1]$ y $[1, 2]$, pero no se considerarán superpuestos.

Por ejemplo, dados los intervalos $(7, 9)$, $(2, 4)$, $(5, 8)$, devuelve 1 ya que el último intervalo se puede eliminar y los dos primeros no se superpondrán.

Los n intervalos que ingresará el usuario no están necesariamente ordenados en ningún orden.

- **Ejercicio 18:**

Dada una cadena, devuelve el primer carácter recurrente que contiene, o nulo si no hay ningún carácter recurrente. Por ejemplo, dada la cadena "acbbac", devuelve "b". Dada la cadena "abcdef", devuelve nulo.

- **Ejercicio 19:**

Dada una matriz de números N por M , imprima la matriz en una espiral en el sentido de las agujas del reloj. Por ejemplo, dada la siguiente matriz:

```
[[1, 2, 3, 4, 5],  
 [6, 7, 8, 9, 10],  
 [11, 12, 13, 14, 15],  
 [16, 17, 18, 19, 20]]
```

Debes imprimir lo siguiente:

1 2 3 4 5 10 15 20 19 18 17 16 11 6 7 8 9 14 13 12

- **Ejercicio 20:**

El usuario ingresa 3 números de 3 dígitos cada uno pegados, en una sola línea. por ejemplo: 139496233. El programa debe desglosarlos (el 1ero 139, 2do 496 y tercero 233) e identificar si cada uno es primo, perfecto o perteneciente a la sucesión de fibonacci.

- **Ejercicio 21:**

La raíz digital de por ejemplo 32 es 5, porque $3 + 2 = 5$ y 5 es menor que 10. La raíz digital de 763 es 3, puesto que $7+6+3=21$, pero al ser 21 mayor o igual a 10 se repite el proceso con 21, así que $2+1=3$ y 3 es menor que 10, por tanto 3 es la raíz digital de 763. Cree un programa en python que imprima la raíz digital de un número natural que ingrese el usuario.

- **Ejercicio 22:**

Cree un juego de piedra, papel y tijeras, claramente debe usar una librería para la generación de números aleatorios. Realice la interfaz como lo considere conveniente, el usuario debe ingresar ya sea "tijera", "piedra" o "papel", mostrar el resultado de la computadora e indicarse quién ganó la partida.

- **Ejercicio 23:**

Dado un número natural, crea un programa en python que imprima todos los números resultantes en la [Conjetura de Collatz](#) hasta llegar a 1.

- **Ejercicio 24:**

Dado un número natural ingresado por el usuario, compruebe si dicho número es un número de armstrong.

Ejemplo: 153 es un número de amstrong y de hecho es el menor de ellos.

La cantidad de dígitos de 153 es 3. Si elevemos todos sus dígitos a 3 y los sumamos tendríamos como resultado el número original 153.