Objetivos

- Resolver problemas en Python a través de algoritmos con ciclos que contemplen una cantidad predefinida de iteraciones. Los ciclos se implementan con la instrucción while y un contador.

- Resolver problemas en Python a través de algoritmos con ciclos basados en condiciones generales (uso de flags, en caso de ser necesario). Los ciclos se implementan con la instrucción while.

- Reconocer la necesidad de uso e implementar patrones conocidos de ciclos: buscar mayor/menor, buscar algo, acumular (+,\*), contar

- Escribir programas que repiten un proceso que ya es repetitivo, creando entonces una estructura de ciclo anidado. Todos los ciclos se implementan con la instrucción while y pueden ser con cantidad de iteraciones definidas o controlados por condiciones generales.

- Diseña algoritmos que utilizan ciclos anidados para resolver problemas que, por su naturaleza, requieren de esa estructura de control.

- Ser capaz de dar seguimiento (ruteo) a un programa o algoritmo que comprenda ciclos.

Clase 1:

1. ¿Cuántas veces se ejecuta cada uno de los siguientes? *Destacar la importancia de actualizar la variable iteradora y las condiciones de borde.*

|  |  |  |
| --- | --- | --- |
| i=1  n=4  while i<n:  print(i) | i=0  n=4  while i<n:  print(i)  i+=1 | i=1  n=4  while i<n:  print(i)  i+=1 |

1. ¿Cuántas veces se ejecuta este código, y qué hace? *Mostrar que un ciclo podría no ejecutarse ninguna sola vez*

|  |
| --- |
| n = int(input('n: '))  while n<=0:  n = int(input('n: ')) |

1. Se quiere obtener la suma de los 5 números ingresados, pero el código a continuación tiene un error. ¿Cuál es? *Variable suma debe incializarse fuera del ciclo*

|  |
| --- |
| i = 0  while i <= 5:  suma = 0  num = int(input("Ingrese un numero: "))  suma += num  i+= 1  print(suma) |

1. Corregir y modificar el programa anterior para que muestre la suma de los números ingresados por la persona hasta escribir 0.

*Cuidar que el resultado sume el primer número ingresado, es decir, que no lo sobrescriba con el input interior del ciclo. También fijarse que la condición de término sea la correcta, pues “hasta escribir 0” podría ser entenderse como while nro == 0.*

1. Considerando el ejemplo de la Conjetura de Collatz, complete la secuencia generada a partir del número 13.

|  |
| --- |
| t = 13  while t != 1:  print(t)  if t%2 == 0:  t = t // 2  else:  t = 3\*t+1  print(1)  13\_\_ \_\_ 20 \_\_ \_\_ \_\_ 8 4 2 1 |

1. Ruteo

[Ruteo clase 8](https://docs.google.com/spreadsheets/d/1Tkzx8m3b1uKisrIzPU05qLVUU3yO4xRfQPN1Hhl4zZY/edit?usp=sharing)

1. Para repasar patrón de encontrar máximo, mínimo y promedio de valores dados (por Gabriela Montenegro): [Control 4](https://docs.google.com/document/d/1XmeVwXZ5r8ZPDlIquoeBIsv5zy_SVBtDn60fllQ0Fn0/edit)

Clases 2 en adelante

1. Se desean sumar los números hasta que la persona ingresa 0. Este código tiene un error, ¿cuál? *Error en la condición del Flag, debe ser != 0*

|  |
| --- |
| continuar = True  suma = 0  while continuar:  opcion = int(input("Ingrese nro a sumar o 0 para detener: "))  if opcion == 0:  suma += opcion  else:  continuar = False |

1. Se desean sumar los números hasta que la persona ingresa 0 o la suma obtenida es 100 o más. Este código tiene dos errores, ¿cuáles? *Error en condición (debe ser and) y en opción (debe estar en else, para no pedir un número de más luego de lograr la meta)*

|  |
| --- |
| opcion = int(input("Ingrese nro a sumar o 0 para detener: "))  meta = False  suma = 0  while opcion != 0 or meta == False:  suma += opcion  if suma >= 100:  meta = True  opcion = int(input("Ingrese nro a sumar o 0 para detener: ")) |

1. Hacer un programa en python que imprima la tabla de multiplicar. Se puede comenzar haciéndola con uno de los valores fijos (ej: tabla de multiplicar del 5 x n, con n ingresado por consola), y luego modificar el programa para que ambos números sean ingresados. Se sugiere entregar este código de andamiaje

|  |
| --- |
| a = 5  b = 10  inicio = 1  fin = 1  while inicio <= \_\_\_:  while fin <= \_\_\_:  print(inicio,"x",fin,"=",a\*b)  fin+=\_\_\_  fin = \_\_\_  inicio += \_\_\_ |

R:

|  |
| --- |
| a = 5  b = 10  inicio = 1  fin = 1  while inicio <= a:  while fin <= b:  print(inicio,"x",fin,"=",a\*b)  fin+=1  fin = 1  inicio += 1 |

1. Factorial con andamiaje (por Luis Ramirez): [Propuestas de Ciclos](https://docs.google.com/document/d/1Dqvw1eZObNKapr0bUS5JxxQCykwBLpzQBySHW5XGOxc/edit?usp=sharing)

Problemas

1. Escriba un programa que lea dos números enteros y muestre la multiplicación entre ellos sin usar el operador \*
2. Haga un programa que calcule los divisores de un número dado, los sume, y muestre el resultado en pantalla.
3. Escriba un programa que lea un número n e imprima los primeros n términos de la siguiente serie: 1,2, 4, 7, 11, … que comienza en 1 y se forma sumando 1, luego 2, luego 3, luego 4, y así sucesivamente.

Ingrese n:10

1 2 4 7 11 16 22 29 37 46

1. La famosa serie FizzBuzz para un número natural N es una sucesión desde 1 hasta N donde:

Los números que sean múltiplos de 3 se cambian por Fizz.

Los números que sean múltiplos de 5 se cambian por Buzz.

Los números que sean múltiplos de 3 y 5 se cambian por FizzBuzz.

Escriba un programa que lea N e imprima la serie FizzBuzz.

Ingrese n:15

1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11 Fizz 13 14 FizzBuzz

1. Un robot ha sido diseñado para moverse a lo largo de una cuadrícula, recibiendo como entrada alguna de las letras N, S, E, O, que le ordenan moverse un metro hacia el norte, sur, este, oeste, respectivamente. La letra F le pone fin al movimiento del robot. Escriba un programa que simule el movimiento el robot, leyendo letras ingresadas una por una. Al finalizar el movimiento, debe imprimir la distancia recorrida y la distancia de la ruta óptima (camino más corto posible para llegar al mismo destino).

Movimiento:N

Movimiento:E

Movimiento:E

Movimiento:E

Movimiento:S

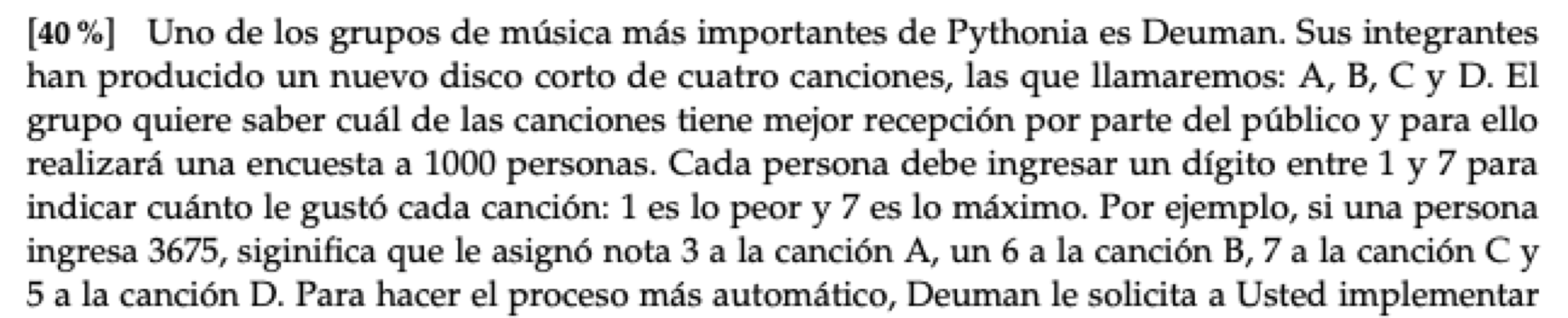
Movimiento:S

Movimiento:O

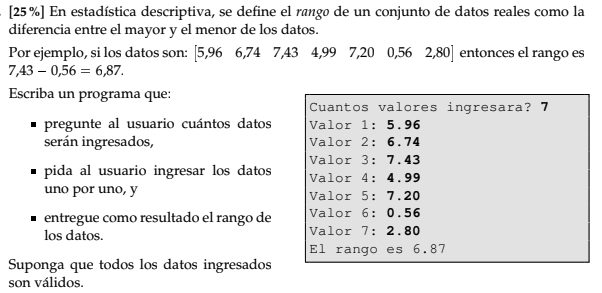
Movimiento:O

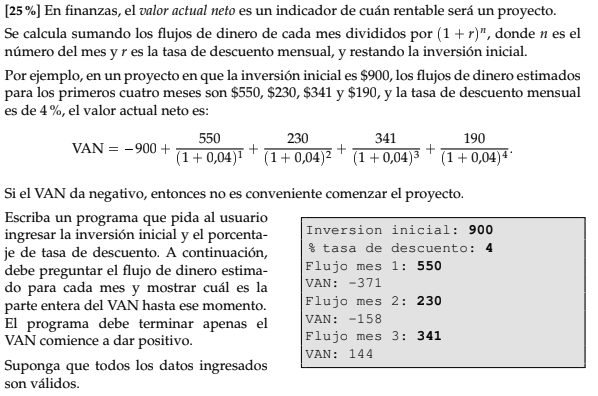
Movimiento:F

Distancia recorrida: 8 mts Distancia óptima: 2 mts



1. Haga un programa que pida a la persona su voto, y muestre por pantalla la canción con mejor nota
2. Luego, modifique el programa para encuestar a 1000 personas y mostrar la canción con mejor nota y cuántas votos obtuvo en total.





1. Haga un programa que sume las cifras de un número de n dígitos.

También haga un programa que invierta un número de n cifras y diga si es palíndromo o no (que se lee igual de izquierda a derecha).