

<https://github.com/TheInventorist/Material-Programacion>

Guia de ejercicios en Python

Listas

Javier Aguilera
Vincent Depassier
Roberto León

June 1, 2020

1. Escriba un programa que solicite números enteros y que los almacene en una lista. El programa debe solicitar números hasta que se ingrese 0, luego debe imprimir todos los números ingresados (Excepto el 0) en orden, de **menor** a **mayor**, cada valor debe mostrarse por pantalla en una sola línea.

```
Ingrese numero: 4
Ingrese numero: 8
Ingrese numero: 3
Ingrese numero: 2
Ingrese numero: 7
Ingrese numero: 0
2
3
4
7
8
```

2. Escriba un programa que solicite números enteros y que los almacene en una lista. El programa debe solicitar números hasta que se ingrese 0, luego debe imprimir todos los números ingresados (Excepto el 0) en orden, de **mayor** a **menor**, cada valor debe mostrarse por pantalla en una sola línea.

```
Ingrese numero: 4
Ingrese numero: 8
Ingrese numero: 3
Ingrese numero: 2
Ingrese numero: 7
Ingrese numero: 0
8
7
4
3
2
```

3. Al analizar datos recolectados en un experimento de ciencia puede ser deseado remover la mayor cantidad de valores extremos (Outliers) antes de realizar cálculos. Escriba un programa que solicite una lista de números al usuario y la cantidad de valores extremos a remover (n). Para eliminar los valores extremos, se deben quitar de la lista los n valores mas pequeños y los n valores mas grandes de la lista. Después de quitar los valores extremos, se debe mostrar por pantalla los nuevos datos ordenados de menor a mayor.

```
cantidad de datos experimento: 10
cantidad de valores extremos (n): 2
Ingrese numero: 8
Ingrese numero: 3
Ingrese numero: 2
Ingrese numero: 7
Ingrese numero: 10
Ingrese numero: 9
Ingrese numero: 1
Ingrese numero: 5
Ingrese numero: 6
Ingrese numero: 4

[3,4,5,6,7,8]
```

4. Escriba un programa que lea palabras ingresadas por el usuario hasta que este ingrese una línea vacía, luego de ingresar la línea vacía se deben imprimir, línea por línea, todas las palabras ingresadas, sin repetirse, en el mismo orden que fueron agregadas.

```
palabra 1: hola
palabra 2: que
palabra 3: que
palabra 4: tal
palabra 5: hola
palabra 6: que

hola
que
tal
```

5. Escriba un programa que lea números ingresados por el usuario hasta que este ingrese una línea vacía. Imprima todos los números ingresados ordenados de menor a mayor, se deben imprimir primero los números negativos en una línea y como lista, luego los 0 en otra y al final, los números positivos en una línea y como lista también, no se permite imprimir línea por línea.

```
ingrese numero: -3
ingrese numero: 2
ingrese numero: 10
ingrese numero: 0
ingrese numero: -1
ingrese numero: 5
ingrese numero: 0
ingrese numero: -4
ingrese numero: 8
```

```
[-4,-3,-1]
[0,0]
[2,5,8,10]
```

6. El divisor propio de un entero positivo n , es un número entero positivo menor a n , el cual divide uniformemente a n . Escriba un programa que calcule todos los divisores propios de un entero positivo n ingresado por el usuario y los muestre en una sola línea como lista.

```
ingrese numero: 16
```

```
[1,2,4,8]
```

```
ingrese numero: 20
```

```
[1,2,4,5,10]
```

7. Un entero, n , dice ser *perfecto* cuando la suma de todos sus divisores propios son igual a n . Por ejemplo, los divisores propios de 28 son;

$$28 : [1, 2, 4, 7, 14]$$

Y la suma:

$$1 + 2 + 4 + 7 + 14 = 28$$

por lo tanto, 28 es un numero perfecto según sus divisores propios. Escriba un programa que imprima todos los números perfectos entre el 1 y el 10000.

8. Realice un programa que solicite un texto al usuario y muestre por pantalla cada palabra del texto como un ítem de una lista.

```
ingrese texto: Ayer tenia mucha hambre, lo que hice fue;  
primero, comprar pan. Segundo, comerlo.  
  
[ 'Ayer', 'tenia', 'mucha', 'hambre', 'lo', 'que', 'hice',  
  'fue', 'primero', 'comprar', 'pan', 'segundo', 'comerlo' ]
```

9. Escriba un programa que lea números ingresados por el usuario hasta que este ingrese una linea vacía. Debe imprimir el promedio de los números, luego debe mostrar todos los números ingresados bajo el promedio en una lista ordenada y en otra lista ordenada, los que estén sobre el promedio.

```
Ingrese numero: 6.3  
Ingrese numero: 2.0  
Ingrese numero: 3.3  
Ingrese numero: 5.8  
Ingrese numero: 9.8  
Ingrese numero: 1.6  
Ingrese numero: 9.3  
Ingrese numero: 9.0  
Ingrese numero: 4.4  
Ingrese numero: 3.7  
Ingrese numero:  
  
Promedio = 5.5  
  
numeros bajo el promedio: [1.6, 2.0, 3.3, 3.7, 4.4]  
  
numeros sobre el promedio: [5.8, 6.3, 9.0, 9.3, 9.8]
```

10. Escriba un programa para simular un campeonato de tenis.

Primero, debe pedir al usuario que ingrese los nombres de ocho tenistas. A continuación, debe pedir los resultados de los partidos juntando los jugadores de dos en dos. El ganador de cada partido avanza a la ronda siguiente.

El programa debe continuar preguntando ganadores de partidos hasta que quede un único jugador, que es el campeón del torneo.

El programa en ejecución debe verse así:

```
Jugador 1: Nadal
Jugador 2: Melzer
Jugador 3: Murray
Jugador 4: Soderling
Jugador 5: Djokovic
Jugador 6: Berdych
Jugador 7: Federer
    Jugador 8: Ferrer

Ronda 1
a.Nadal - b.Melzer: a
a.Murray - b.Soderling: b
a.Djokovic - b.Berdych: a
a.Federer - b.Ferrer: a

Ronda 2
a.Nadal - b.Soderling: a
a.Djokovic - b.Federer: a

Ronda 3
a.Nadal b.Djokovic: b

Campeon: Djokovic
```

11. Un grupo de bio-tecnologos tiene demasiado trabajo analizando cadenas de ADN. Todo el trabajo lo realizan a mano ya que no tuvieron un buen ramo de Programacion. Por esto, le piden a los estudiantes que les ayuden. Las cadenas de ADN estan compuestas por 4 bases nitrogenadas (A: Adenina, C: Citosina, G: Guanina y T: Timina) agrupadas en bloques de 4. Ahora usted debe ayudar en las siguientes operaciones: a) Escriba un programa que reciba un string con una cadena de ADN e imprima True si la cadena es valida o False si no lo es. Una cadena no es valida cuando aparecen bases nitrogenadas distintas a las antes descritas.

```
cadena: CTGA CTGA AATT GGGC CTGG CCCC
True
```

```
cadena: CTGA XCGA CGAT GGTA ACCC CCPC TTAA
False
```

- b) Escriba un programa que reciba un string con una cadena de ADN y una base nitrogenada. El programa debe imprimir la cantidad de apariciones de la base en la cadena. Asuma que la cadena siempre es valida.

```
cadena: CTGA CTGA AATT GGGC CTGG CCCC
base: A
4
```

```
cadena: CTGA CCGA CGAT GGTA ACCC CCPC TTAA
base: C
10
```

- c) Escriba un programa que reciba un string con una cadena de ADN e indique la base nitrogenada que mas apariciones tiene en la cadena. Asuma que la cadena siempre es valida.

```
cadena: CTGA CTGA AATT GGGC CTGG CCCC
C
```

```
cadena: CTGA CCGA CGAT GTTA ACTC TTAA
T
```

12. Un numero entero se dice n-multiplo-m si la cantidad de digitos que son multiplos de n es igual a la cantidad de digitos que son multiplos de m. Por ejemplo, el numero 4569 es 2-multiplo-3 ya que tiene dos digitos multiplos de 2 (4 y 6) y dos digitos multiplos de 3 (6 y 9). Escriba un programa que recibe tres numeros enteros (numero, n y m) e imprima True si numero es n-multiplo-m, en caso contrario, debe imprimir False.

```
numero: 4569
n: 2
m: 3
True
```

```
numero: 489238
n: 3
m: 4
False
```


13. La ciudad de Pitonia tiene una alta congestión de vehículos circulando por sus calles. Las autoridades han decidido aplicar restricción vehicular para descongestionar la ciudad en base a las patentes de los vehículos. La patente de un vehículo es un string de cuatro letras y dos dígitos, y la restricción depende del penúltimo dígito. Por ejemplo, para la patente GEEA78, el dígito 7 es el utilizado para evaluar la restricción. La restricción vehicular de los días hábiles de la semana se encuentra ingresada en el string restricción, el cual indica el día de la semana mediante la letra inicial del día, junto con los 4 dígitos que están en restricción. Cada día está separado por un espacio.

a) Escriba un programa que muestre los dígitos que tienen restricción para un día de la semana en particular.

```
restriccion: L3456 M7890 X1234 J5678 V9012
dia: jueves
5 6 7 8
```

```
restriccion: L3456 M7890 X1234 J5678 V9012
dia: martes
7 8 9 0
```

b) Escriba un programa que indique si un vehículo está o no con restricción para un día de la semana en particular.

```
restriccion: L3456 M7890 X1234 J5678 V9012
dia: lunes
patente: BBdT35
True
```

```
restriccion: L3456 M7890 X1234 J5678 V9012
dia: miercoles
patente: HTCT99
False
```