

Laboratorio 22/10/2018

Uso de Tuplas y Diccionarios

Nombre: _____

En este laboratorio deberás ingresar a la Plataforma de Laboratorios en <https://ing1103.iccuandes.org>, utilizando la cuenta de usuario que te fue enviada por correo electrónico. **Esta plataforma evalúa tus soluciones a los problemas de laboratorio en forma automatizada, es decir, no corrigen los ayudantes sino un software.**

Problema 1

Una recta en el plano cartesiano se puede representar como $y = mx + b$, siendo m el valor de la pendiente y b nos indica en dónde corta al eje y.

Dados dos puntos en el plano cartesiano $p_1 = (x_1, y_1)$ y $p_2 = (x_2, y_2)$ los valores de m y b que representan una recta que pasa por ambos puntos vienen dado por:

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$
$$b = y_1 + \frac{x_1(y_1 - y_2)}{x_2 - x_1}$$

Tarea a realizar

Se le entregará el siguiente código base:

Listing 1: Código base para el problema 1

```
1 a = float(input())
2 b = float(input())
3 c = float(input())
4 d = float(input())
5 m, b = ecuacion_recta((a,b),(c,d))
6 print("y="+str(round(m,2))+ "x"+str(round(b,2)))
```

Debe implementar las funciones `ecuacion_recta(punto1,punto2)`, la cual recibe como parámetros dos puntos, los cuales son tuplas de tamaño dos. La función retorna una tupla de tamaño dos de la forma (m, b) , en donde m y b representan la pendiente y la intersección de la recta con el eje y respectivamente.

Si queremos probar el programa anterior para los puntos $p_1 = (3, 4)$ e $p_2 = (1, 3)$ entonces la entrada será:

Ejemplo de Entrada

3
4
1
3

Para la cual se obtiene como salida:

Ejemplo de Salida

$$y=0.5x+2.5$$

Si queremos probar el programa para los puntos $p_1 = (-3, 4)$ e $p_2 = (1.5, -3.2)$, la entrada sería:

Ejemplo de Entrada

-3

4

1.5

-3.2

Para la cual se obtiene como salida:

Ejemplo de Salida

$$y=-1.6x+-0.8$$

A continuación más ejemplos:

Ejemplo de Entrada

0

-1.2

-100

2

Ejemplo de Salida

$$y=-0.03x+-1.2$$

Ejemplo de Entrada

-3.3

3.3

0

0

Ejemplo de Salida

$y = -1.0x + 0.0$

Ejemplo de Entrada

1000

1000

-2000

100

Ejemplo de Salida

$y = 0.3x + 700.0$

Consejos

Si quiere que su laboratorio obtenga todos los puntos sin problemas, solo debe definir la función solicitada al inicio del código base, sin alterarlo.

Puntaje

El puntaje total (100) se asignará probando el programa realice correctamente cada una de las acciones. Se ejecutarán los casos de prueba mostrados como ejemplo en el enunciado y ninguno más.

Problema 2

Se realiza una prueba beta del voto digital para las próximas elecciones presidenciales. Se desea implementar una aplicación en python que permita obtener un ranking ordenado por la cantidad de votos de forma descendente. Para ello se le entregará un código base:

Listing 2: Código base para el problema 2

```
1 n = int(input())
2 diccionario = leer_votos(n)
3 lista = list(diccionario.items())
4 lista.sort(key=lambda tup:tup[1],reverse=True)
5 for elemento in lista:
6     print(elemento[0],elemento[1])
```

Tarea a realizar

Debe implementar una sola función llamada `leer_votos(n)` que recibe como parámetro la cantidad de votos por leer. Esta función utiliza `input` para solicitar al usuario ingresar uno a uno los nombres de los candidatos y retorna un diccionario cuya clave es el nombre del candidato y valor (o llave) un número entero que señala la cantidad de votos recibidos por dicho candidato.

Notar que la primera línea recibida por este programa (por el código base) es un entero que señala la cantidad de votaciones que debe leer, luego la función pide dicha cantidad de votaciones una por una, y si la implementa correctamente, el código base será el encargado de imprimir la salida (usted no tiene que hacer nada con `print`).

Veamos el siguiente ejemplo:

Ejemplo de Entrada

```
10
fernando lagos
joaquin pineda
joaquin pineda
masiel bescherelle
joaquin pineda
fernando lagos
joaquin pineda
masiel bescherelle
masiel bescherelle
joaquin pineda
```

Para la cual se obtiene como salida:

Ejemplo de Salida

```
joaquin pineda 5
masiel bescherelle 3
fernando lagos 2
```

Consejos

Si quiere que su laboratorio obtenga todos los puntos sin problemas, solo debe definir solo la función solicitada al inicio del código base, sin alterarlo.

Puntaje

El puntaje total (100) se asignará probando el programa realice correctamente cada una de las acciones. Se ejecutarán varios casos de prueba cuya primera línea será siempre un numero entero mayor que cero y las siguientes líneas serán strings con el nombre de candidatos reales o imaginarios. Siempre el número entero será consistente con la cantidad de votaciones ingresadas (si dice 10, entonces luego hay 10 votaciones, ni más ni menos). La salida está ordenada de mayor a menor por la cantidad de votos y nunca habrá empates en la cantidad de votos entre candidatos.