

Programación – Certamen 2 - Miércoles 16 de Diciembre de 2020

1. [30 %] La Feria de Software es un gran evento educativo en donde los estudiantes de 5to. año de Ingeniería Civil en Informática exponen los productos que han desarrollado a lo largo de todo un año. En esta feria encontrarás innovación, emprendimiento y mucha tecnología, por lo que te invitamos a visitarla este jueves 17 de diciembre, desde las 12:00 hrs. Sólo debes inscribirte en el enlace:

<http://www.virtual.feriadesoftware.cl>

Todos los años, quienes participan de la Feria eligen a un o una representante para dar un discurso de bienvenida, y necesitan de su ayuda para completar esta elección. Para ello, se le solicitó a cada estudiante que votara por 3 de sus compañeros o compañeras. Quien obtenga la mayor votación será quien haga el discurso. Los datos de la votación se registraron en un archivo cuyas líneas tienen el siguiente formato:

estudiante1;estudiante2;estudiante3

Cada línea del archivo corresponde a la votación de uno o una estudiante de la Feria. A modo de ejemplo, a continuación se muestra una parte del archivo descrito:

votos.txt

```
Andrea Campos;Andrea Campos;Pedro Vasquez
pedro vasquez;mabel lazo;javier Godoy
Andrea Campos;andrea campos;Andrea Campos
FEDERICO ROBLEDO;PEDRO VASQUEZ;JAVIER GODOY
Mabel Lazo;federico Robledo;Andrea Campos
ANDREA CAMPOS;Federico Robledo;Sergio Meza
...
```

Observe que los nombres pueden estar escritos en mayúsculas o minúsculas o una combinación de ambos, y que algunas personas quisieron votar por una misma persona más de una vez, aunque esto no está permitido.

Escriba la función **obtener_representante(nombre_archivo)**, que recibe como parámetro un *string* con el nombre del archivo que contiene los datos de la votación, siguiendo el formato descrito. La función debe retornar una tupla con el nombre en mayúscula y la cantidad de votos de quien resultó ganador. En caso de empate, puede retornar cualquiera de los que empatan.

Si en una línea hay más de un voto para una misma persona, no se deben considerar los repetidos. Por ejemplo, la tercera línea del archivo mostrado contiene:

```
Andrea Campos;andrea campos;Andrea Campos
```

En este caso, sólo se contará un voto para ANDREA CAMPOS.

Por otra parte, considere la primera línea del archivo:

```
Andrea Campos;Andrea Campos;Pedro Vasquez
```

En este caso, se debe contar un voto para ANDREA CAMPOS y un voto para PEDRO VASQUEZ.

Ejemplo:

```
>>> print(obtener_representante('votos.txt'))
('ANDREA CAMPOS', 4)
```

Programación – Certamen 2 - Miércoles 16 de Diciembre de 2020

2. [70 %] Necesitamos de su ayuda para clasificar y ordenar los locales de comida que se encuentran dentro de un rango de distancia particular. Los datos de los locales se encuentran en un archivo cuyas líneas tienen el siguiente formato:

```
nombre;tipo;x,y;calificacion;precio
```

Las coordenadas x, y permiten saber la ubicación del local respecto nuestra ubicación, que puede considerarse como $0, 0$. Observe que esos valores están separados entre sí con una coma, mientras que todos los demás valores se separan con punto y coma. El campo de precio está compuesto por una serie de signos de peso: \$. Entre mayor es el número de signos de peso, más caro resulta comer en el local.

A modo de ejemplo, a continuación se muestra una parte del archivo descrito:

locales.txt

```
Carl's Jr.;hamburguesas;10,10;4.8;$$$  
La Cevichería;mariscos;4,3;4.5;$$$$  
Lasagna Factory;italiano;8,4;2.5;$$  
Burger King;hamburguesas;8,1;3.4;$  
Nueva China;comida china;2,2;4.9;$$$$  
...
```

Escriba la función **filtrar_locales(nombre_archivo, rango)**, que recibe como parámetro un *string* con el nombre del archivo que contiene los datos de los locales, y un número que representa el rango de distancia en el que queremos buscar. La función debe considerar únicamente locales que se encuentren a esa distancia o a una menor. Recuerde que la distancia de un punto (x, y) al origen se calcula con la fórmula: $\sqrt{x^2 + y^2}$.

Se debe crear un archivo por cada tipo distinto de comida, cuyo nombre debe ser el tipo seguido del sufijo .txt. Por ejemplo, mariscos.txt. Cada archivo debe contener los locales que cumplan con el criterio del rango de distancia, ordenados del más barato al más caro, y siguiendo el formato que se muestra en el ejemplo. Si ningún local cumple con el criterio, se debe crear de todas maneras el archivo pero no debe contener ninguna línea.

Además de crear los archivos señalados, la función debe retornar el número total de locales distintos que se consideraron, es decir, que cumplieron con el criterio de distancia.

A continuación se muestra el comportamiento que debe tener la función, y el contenido de los archivos resultantes.

Ejemplo:

```
>>> print(filtrar_locales('locales.txt', 10))  
18
```

comida china.txt

```
Locales de tipo comida china que están a distancia 10 o menos:  
China Express, calificación 2.9, precio: $.  
Restaurant Ming Feng, calificación 3.9, precio: $$.  
P.F. Chang's, calificación 4.9, precio: $$$$.  
Nueva China, calificación 4.9, precio: $$$$.
```

hamburguesas.txt

```
Locales de tipo hamburguesas que están a distancia 10 o menos:  
Burger King, calificación 3.4, precio: $.  
McDonald's, calificación 3.2, precio: $.  
Ruby Tuesday, calificación 4.5, precio: $$$$.
```

Programación – Certamen 2 - Miércoles 16 de Diciembre de 2020

italiano.txt

Locales de tipo italiano que están a distancia 10 o menos:
Lasagna Factory, calificación 2.5, precio: \$\$.
Il Mascarpone, calificación 4.5, precio: \$\$\$\$.

mariscos.txt

Locales de tipo mariscos que están a distancia 10 o menos:
La Cevichería, calificación 4.5, precio: \$\$\$\$\$.

pizza.txt

Locales de tipo pizza que están a distancia 10 o menos:
Little Caesars Pizza, calificación 4.0, precio: \$.
Telepizza, calificación 2.9, precio: \$.
Domino's Pizza, calificación 3.9, precio: \$\$.
Melt Pizzas, calificación 3.5, precio: \$\$.
Pizza Hut, calificación 4.2, precio: \$\$\$.
Papa John's, calificación 4.1, precio: \$\$\$\$.

sushi.txt

Locales de tipo sushi que están a distancia 10 o menos:
NIU Sushi, calificación 4.5, precio: \$\$\$.
Sakura, calificación 4.3, precio: \$\$\$\$.