

Programación – Certamen Fase I - Jueves 9 de Enero de 2020

Sugerencia: Se recomienda el uso de diccionarios. La solución es más sencilla si se utilizan.

Contexto

Un ornitólogo registra las observaciones de aves que realiza en un archivo de texto como el que se muestra a continuación:

observaciones.txt

```
2019/12/17:Valparaiso:Chercan,Pelicano
2019/12/15:Santiago:Zorzal,Chercan,Paloma,Tenca
2019/12/19:Valparaiso:Gaviota,Colibri
2019/12/20:Santiago:Chercan,Loica,Paloma
2019/12/14:Curico:Loica,Queltehue
2019/12/17:Valdivia:Cisne,Colibri,Cormoran
2019/12/15:Curico:Chercan,Loica,Colibri
...
```

Cada línea contiene la fecha de la observación, el lugar donde ocurrió y las especies avistadas, siguiendo la siguiente estructura: **fecha:lugar:especie, especie, ..., especie**. La fecha se expresa en el formato **aaaa/mm/dd**, en donde el año siempre tiene 4 dígitos y el mes y el día siempre tienen dos dígitos.

Observe que una misma ciudad puede aparecer varias veces en el archivo en distintas fechas, pero nunca en la misma fecha.

1. [50 %] Escriba la función **mas_vistos(nombre_archivo)**, que recibe como parámetro un **string** con el nombre de un archivo que sigue el formato descrito. La función debe construir el archivo **mas_vistos.txt** que contendrá las 3 especies que aparecen más veces en las observaciones, ordenadas de mayor a menor según el número de apariciones. En caso de empates puede mostrar cualquiera de las especies que empatan. Puede suponer que siempre existirán al menos 3 especies diferentes.

La función debe retornar el número total de especies distintas que se observaron.

Para comprender el formato que debe tener el archivo, guíese por el ejemplo.

Ejemplo:

```
>>> print(mas_vistos('observaciones.txt'))
11
```

mas_vistos.txt

```
Se observaron 4 especímenes de Chercan
Se observaron 3 especímenes de Loica
Se observaron 3 especímenes de Colibri
```

Programación – Certamen Fase I - Jueves 9 de Enero de 2020

2. [50 %] Escriba la función `vistos_por_ciudad(nombre_archivo)`, que recibe como parámetro un `string` con el nombre de un archivo que sigue el formato descrito. La función debe construir el archivo `vistos_por_ciudad.txt` que agrupará por ciudad las observaciones efectuadas, una por línea, incluyendo la fecha y la especie avistada. Las observaciones dentro de una ciudad deben aparecer ordenadas por fecha, desde la más antigua hasta la más reciente. Por otra parte, las ciudades no siguen ningún orden particular en el archivo.

La función debe retornar el número total de ciudades distintas donde hubo avistamientos.

Para comprender el formato que debe tener el archivo, guíese por el ejemplo.

Ejemplo:

```
>>> print(vistos_por_ciudad('observaciones.txt'))
4
```

`vistos_por_ciudad.txt`

```
Valparaiso:
    2019/12/17:Chercan
    2019/12/17:Pelicano
    2019/12/19:Colibri
    2019/12/19:Gaviota
Santiago:
    2019/12/15:Chercan
    2019/12/15:Paloma
    2019/12/15:Tenca
    2019/12/15:Zorzal
    2019/12/20:Chercan
    2019/12/20:Loica
    2019/12/20:Paloma
Curico:
    2019/12/14:Loica
    2019/12/14:Queltehue
    2019/12/15:Chercan
    2019/12/15:Colibri
    2019/12/15:Loica
Valdivia:
    2019/12/17:Cisne
    2019/12/17:Colibri
    2019/12/17:Cormoran
```