

Programación—Certamen 3 - 5 de Julio de 2017

Nombre

Rol

Paralelo

2. [40 %] Un concesionario de una autopista mantiene información relevante de sus clientes en 3 archivos: Un registro diario de patentes que circulan por ella, la información de las patentes que contienen o no contienen TAG (indicado con un 1 o un 0 respectivamente) y por último, la información de los dueños y de sus patentes asociadas.

A modo de ejemplo se presenta cada uno de los archivos en el cuadro a continuación:

registro_diario.txt

```
abmn32
crtj12
df1p11
hb5101
```

info_duenos.txt

```
Alex Perez;crtj12,hb5101,kcf136,emda16
Aguiles Castro ;abmn32,t1jg99,avrv33
Maria Gana;ab7677
Fede Santos;utfs90,df1p11
```

De la información de ejemplo se desprende que Alex Perez tiene cuatro vehículos registrados en la autopista con patentes crtj12, hb5101, kcf136 y emda16. De estos solo crtj12 cuenta con dispositivo TAG y los otros tres no. Además, los vehículos con patente crtj12 y hb5101 pasaron durante el día por la autopista.

Siguiendo con el ejemplo, Maria Gana tiene solamente un vehículo y éste no circuló durante el día por la autopista.

Considere que en el archivo info_duenos cada persona aparece solamente una vez y que puede tener un número variable de patentes.

patentes.txt

```
crtj12,1
abmn32,0
hb5101,0
df1p11,1
t1jg99,0
jfzo10,0
kcf136,0
utfs90,0
ab7677,0
avrv33,0
emda16,0
```

- a) Se necesita saber cuantos vehículos tienen cada usuario. Para esto contruya una función `patentes_por_dueno(archivo_duenos)` que recibe el nombre del archivo que contiene la información de los dueños y retorna un diccionario. Este diccionario debe tener como llave el nombre y apellido de la persona y como valor el número de vehículos que posee.

```
>>> patentes_por_dueno('info_duenos.txt')
{'Alex Perez': 4, 'Aguiles Castro ': 3, 'Fede Santos': 2,
'Maria Gana': 1}
```

- b) Se requiere listar a aquellas patentes que circularon durante el día por la autopista y que no disponen de TAG. Para esto, construya la función `patentes_multadas(archivo_registro, archivo_patentes)` que recibe los nombres de los archivos que contienen el registro diario de patentes y el registro general de patentes respectivamente. Esta función debe retornar una lista con las patentes que satisfacen el criterio ya mencionado.

```
>>> patentes_multadas('registro_diario.txt', 'patentes.txt')
['abmn32', 'hb5101']
```

- c) Se necesita listar a su(s) respectivo(s) dueño(s). Para esto construya la función `personas_multadas(archivo_registro, archivo_patentes, archivo_duenos)` que recibe los nombres de los archivos que contienen el registro diario de patentes, el registro general de patentes y la información de los dueños de automóviles respectivamente. Esta función debe retornar un listado con los nombres de los dueños de las patentes multadas. Este listado no debe contener nombres repetidos.

```
>>> personas_multadas('registro_diario.txt', 'patentes.txt', '
info_duenos.txt')
['Aguiles Castro ', 'Alex Perez']
```

