 PONTIFICIA UNIVERSIDAD CATÓLICA ARGENTINA Facultad de Ingeniería y Ciencias Agrarias	EXÁMEN PRÁCTICO	01 / 12 / 2020	FINAL	
		INFORMÁTICA GENERAL		
	En máquina, virtual presencial y a distancia			TEMA 1

INTRODUCCIÓN

Exam.net nos ofrece, el siguiente menú:

- 1. **Área de escritura:** Es un editor de texto que lo podés usar como borrador, anotador, anotar comentarios del enunciado, y registrar las aclaraciones que realicen los docentes de manera oral durante el examen, etc.
- 2. **>_ Programación:** Es una herramienta de programación. Podés escribir el código en lenguaje de programación y probarlo. Debes seleccionar el lenguaje Python.
- 3. **EXAMEN ...:** Aquí vas a encontrar el texto con la consigna del examen.
- 4. **Chat de profesores:** Podrás realizar preguntas a tus profesores utilizando este chat.
- 5. **Enviar examen:** Es el link que te lleva al envío del examen, donde estarías realizando la entrega de tu examen. *En síntesis..terminaste el examen !!!.*

El examen práctico contiene **DOS ejercicios a desarrollar**. **IMPORTANTE LEER:** El estudiante deberá guardar **un** archivo por **cada** ejercicio. Grabar con el nombre del ejercicio (**EJ01** o **EJ02**) según corresponda.

Además, para identificar el ejercicio es obligatorio escribir dentro del archivo codificado (como comentario en los dos primera línea), el número de ejercicio que están resolviendo y el apellido y nombre del estudiante que lo resuelve. Por ejemplo para el ejercicio 1 (**EJ01**), para la estudiante **Suarez Alejandra** deberá quedar :

```
### EJ01 ###
### SUAREZ ALEJANDRA ###
```

Por favor, identificar todos lo ejercicios de esta manera

Es importante que si grabas varias versiones de un mismo ejercicio, antes de enviar el examen, elimina las versiones que están de más, y deja una sólo por ejercicio (la última). De esta manera recibiremos un archivo por ejercicio.
No deben usar input ni tampoco archivos la resolución de ninguno de los ejercicios.

IMPORTANTE: Sólo podemos usar los siguientes elementos del lenguaje para resolver los problemas en los **FINALES o evaluaciones en la cátedra. Y sujeto a indicaciones del profesor.**

```
- DELIMITADORES:      # ' " , : . \ ( ) [ ] { }
- OPERADORES:        + - * ** / // < > <= >= == != = += -=
- PALABRAS RESERVADAS: and def elif else False for if import in None
not or return True while
- FUNCIONES INTEGRADAS: bool() chr() dict() float() id() input() int() len()
list() max() min() open() ord() print() range() round() str() tuple()
type()
- FUNCIONES LIBRERÍA math: math.pow() math.sqrt() - FUNCIONES LIBRERÍA random:
random.randint()
- MÉTODOS DE STRING: .format() .split()
- MÉTODOS DE LIST: .append() .insert() .pop() .remove()
- MÉTODOS DE DICT: .get() .keys() .pop() .values()
- MÉTODOS DE FILE: .close() .readline() .readlines() .write()
```

DESCRIPCIÓN DE LOS DATOS DEL PROBLEMA A RESOLVER

Se cuenta con la información de ciudades y de cantidad de residuos (“basura”) generada por cada ciudad, para un día determinados. A continuación más detalle.

Se cuenta con dos variables del tipo lista `lstCiudad` y `lstResiduos`. Cada lista tiene el contenido proveniente de la lectura completa del archivo `.csv` correspondiente, utilizando el método `.readlines()`. Los campos del archivo están separados por coma (,) y no está ordenado por ningún campo. El contenido del archivo y su estructura de datos es:

	<code>ciudad.csv</code>	<code>residuos.csv</code>
Estructura de datos	<code>ID_ciudad</code> [int] <code>nombre</code> [string]	<code>cantidad</code> [int] <code>ID ciudad</code> [int] <code>fecha</code> [int] (***)
Ejemplo de Contenido	223,Parana 114,Merlo 218,Guaymallen 132,C. Rivadavia 341,Adolfo Alsina 404,Jose C. Paz	33,114,200518 31,223,200519 27,218,200319 26,132,200616 74,341,200319 62,404,200606 46,218,200709 55,132,200630 55,341,200612 54,404,200701 23,114,200315 55,223,200519 34,218,200319 33,132,200425 27,341,200422 21,404,200501 31,114,200503 44,114,200513 44,218,200519

(***) Las fechas son números de 6 dígitos que tiene un formato AMD.

Formato AMD (Año Mes Día). Dentro de un **número entero** de seis dígitos hay una fecha representada según se indica: **AAMMDD**. Ejemplo: `120318`. El año (AA) es 12, el mes (MM) es 03, el día (DD) es 18.

COMENTARIO SOBRE LOS ARCHIVOS

- `ciudad.csv`: El archivo contiene la información de ciudades. Contiene dos campos: - `ID_ciudad`: Es el número de ciudad. El campo `ID_ciudad` es clave, es decir no se repite su valor en el archivo. - `nombre`: Es el nombre de la ciudad.


- `residuos.csv`: El archivo contiene la información de la cantidad de residuos recolectados en cada ciudad, por día.. Contiene tres campos:
`cantidad`: Es la cantidad de residuo que se recolectó en una ciudad, en un día determinado. `ID_ciudad` es el número identificador de la ciudad. En este archivo se puede repetir muchas veces. `fecha` es la fecha en que se recolectó el residuo en la ciudad.
Es clave en este archivo `ID_ciudad` y `fecha`, ambos a la vez. Quiere decir que no se repiten en el archivo las mismas combinaciones de `ID_ciudad` y `fecha`.

EJEMPLO DE LISTAS - Obtenidas a partir de la lectura de cada archivo
Resulta de un `readlines()` a `ciudad.csv`

```
lstCiudad = ['223,Parana\n', '114,Merlo\n', '218,Guaymallen\n', '132,C. Rivadavia\n', '341,Adolfo Alsina\n', '404,Jose C. Paz\n']
```

Resulta de un `readlines()` a `residuos.csv`

```
lstResiduos = ['33,114,200518\n', '31,223,200519\n', '27,218,200319\n', '26,132,200616\n', '74,341,200319\n', '62,404,200606\n', '46,218,200709\n', '55,132,200630\n', '55,341,200612\n', '54,404,200701\n', '23,114,200315\n', '55,223,200519\n', '34,218,200319\n', '33,132,200425\n', '27,341,200422\n', '21,404,200501\n', '31,114,200503\n', '44,114,200513\n', '44,218,200519\n']
```

 PONTIFICIA UNIVERSIDAD CATÓLICA ARGENTINA Facultad de Ingeniería y Ciencias Agrarias	EXÁMEN PRÁCTICO	01 / 12 / 2020	FINAL	
		INFORMÁTICA GENERAL		
	En máquina, virtual presencial y a distancia			TEMA 1

Desarrollar la función `media_01 (n,lstCiudad ,lstResiduos)` que recibe por parámetro un número entero `n`. y las listas `lstCiudad` y `lstResiduos` que contiene la información de lectura del archivo correspondiente con readlines (cómo se indica arriba).

La función deberá calcular la media (promedio) de residuos para cada una de la ciudades y la media (promedio) del total de residuo para todas las ciudades. La función deberá retornar una tupla de dos elementos, de acuerdo a la siguiente especificación:

- En la posición `0(cero)` de la tupla se debe encontrar un número “float” que representa la media (promedio) del total de residuo para todas las ciudades.
- En la posición `1(unos)` de la tupla se debe encontrar una lista con sólo las `n` primeras ciudades que más residuos producen.
 - Cada elemento de dicha lista debe ser [`NombreCiudad` , `CantidadPormedioResiduo`]
 - `NombreCiudad` es un string
 - `CantidadPormedioResiduo` es un float

Este es un `main` de ejemplo el cual sugerimos utilizar para ejecutar pruebas con la función del ejercicio:

```
def main():
    print("Prueba para el EJ01")
    lstCiudad = ['223,Parana\n', '114,Merlo\n', '218,Guaymallen\n', '132,C. Rivadavia\n', '341,Adolfo Alsina\n', '404,Jose C. Paz\n']
    lstResiduos = ['33,114,200518\n', '31,223,200519\n', '27,218,200319\n', '26,132,200616\n', '74,341,200319\n', '62,404,200606\n', '46,218,200709\n', '55,132,200630\n', '55,341,200612\n', '54,404,200701\n', '23,114,200315\n', '55,223,200519\n', '34,218,200319\n', '33,132,200425\n', '27,341,200422\n', '21,404,200501\n', '31,114,200503\n', '44,114,200513\n', '44,218,200519\n']

    print(media_01(3,lstCiudad ,lstResiduos ))
```

Al ejecutar el `main` planteado arriba, la **SALIDA ESPERADA** es:

```
>>> Prueba para el EJ01

(40.78, [['Adolfo Alsina', 60.8], ['Parana', 55.0], ['Jose C. Paz', 45.6666]])

FE DE ERRATAS: (40.78, [['Adolfo Alsina', 52.0], ['Jose C. Paz', 45.6], ['Parana', 43.0] ])
```


EJ02

Desarrollar la función `media_02 (lstResiduos)` que recibe por parámetro la lista `lstResiduos` que contiene la información de lectura del archivo correspondiente con readlines (cómo se indica arriba).

La función deberá calcular la media (promedio) de residuos para cada mes (*) que figure en `lstResiduos`. y retornar un diccionario con el promedio mensual de residuos generados. El diccionario debe cumplir la siguiente especificación:

- Clave: debe ser un número entero que representa el número de mes. El rango válido es de 1 a 12 inclusives.
- Valor: Es un número float que representa el promedio mensual.

(*) Importante: sólo *DEBE* contener las claves de aquellos meses que existan en `lstResiduos`. Si no se encuentra el mes en dicha lista, no se deberá consignar la información en el diccionario. Además, debe considerar para calcular la media como “mismo mes” a aquellas fechas que tengan el mes igual sin importar su año. Por ejemplo, para el mes de Mayo, se debe calcular la media de todos los meses de Mayo que aparezcan en el archivo, “sean del año que sean”.

 PONTIFICIA UNIVERSIDAD CATÓLICA ARGENTINA Facultad de Ingeniería y Ciencias Agrarias	EXÁMEN PRÁCTICO	01 / 12 / 2020	FINAL	
		INFORMÁTICA GENERAL		
	En máquina, virtual presencial y a distancia		TEMA 1	

Este es un `main` de ejemplo el cual sugerimos utilizar para ejecutar pruebas con la función del ejercicio:

```
def main():
    print("Prueba para el EJ02")
    lstResiduos = ['33,114,200518\n', '31,223,200519\n', '27,218,200319\n', '26,132,200616\n',
'74,341,200319\n', '62,404,200606\n', '46,218,200709\n', '55,132,200630\n', '55,341,200612\n',
'54,404,200701\n', '23,114,200315\n', '55,223,200519\n', '34,218,200319\n', '33,132,200425\n',
'27,341,200422\n', '21,404,200501\n', '31,114,200503\n', '44,114,200513\n', '44,218,200519\n']

    print(media_02(lstResiduos ))
```

Al ejecutar el `main` planteado arriba, la **SALIDA ESPERADA** es:

```
>>> Prueba para el EJ02
{ 3: 39.5, 4: 30.0, 5: 37.0, 6: 49.5, 7: 50.0 }
```

FIN