Grado en análisis de datos en la empresa
PROGRAMACIÓN I
2º examen parcial - 29/05/2024

Universidad Autónoma de Madrid

INTRODUCCIÓN

- Duración del examen: 3 horas.
- El examen consta de 2 partes: teoría (ejercicios 1 y 2) y prácticas (ejercicios 3 y 4). Puedes organizar el tiempo como desees.
- Solo puedes acceder a la página del curso en Moodle. Si accedes a otras páginas, correo, chats, etc. se te expulsará del examen. Se comprobará el uso del chat a posteriori. Caso de que lo hayas usado, tu calificación será 0.

ENTREGAS

- Debes hacer tus entregas en la tarea en Moodle destinada a ello (en la cuestión 2).
- Debes entregar tres archivos .py
  1. **Apellido1Apellido2.py** --> Contendrá las respuestas a los ejercicios 1, 2 y 4.
  2. **Apellido1Apellido2_aux_gestion.py** --> modificación del auxiliar de gestión (Ejercicio 3).
  3. **Apellido1Apellido2_persona.py** --> modificación de la clase persona (Ejercicio 3)
- Cada archivo debe estar identificado con una cabecera con tu nombre y apellidos y fecha.
- No uses eñes, tildes, guiones, subrayados ni caracteres especiales.
- Si no sigues algunas de estas instrucciones se penalizará la entrega con 1 punto.

RÚBRICAS

La puntuación de los ejercicios 1 y 2 es la siguiente:

1) **Ejercicio 1: 5 puntos.**
   a) Si está bien: 5 puntos.
   b) Si está bien parcialmente (se calcula bien en algunos casos y en otros no): 2.5 puntos.
   c) Si no se calcula bien ningún valor, pero el programa no provoca error: 1 punto.
   d) Si el programa provoca un error: 0 puntos.

2) **Ejercicio 2: 5 puntos.**
   a) Si está bien: 5 puntos.
   b) Si está bien parcialmente (se muestra parte de la información de las líneas de metro y parte no): 2.5 puntos.
   c) Si no se muestra nada bien pero el programa provoca un error: 1 punto.
   d) Si el programa provoca un error: 0 puntos.

Sobre estas calificaciones se pueden introducir variaciones dependiendo de si se incluyen *print* o input indebidos, etc. Estas calificaciones contribuyen a la nota de teoría de la asignatura.

Grado en análisis de datos en la empresa
PROGRAMACIÓN I
2º examen parcial - 29/05/2024

Universidad Autónoma de Madrid

INSTRUCCIONES

- Descarga el archivo `examen_PROG1.py`
- **Renombra este archivo como Apellido1Apellido2.py, que es el que debes entregar.**
- Debes completar las funciones que se indican en los ejercicios 1, 2 y 4. Están identificadas con la palabra clave `pass`. Cuando las codifiques correctamente obtendrás las salidas que se te muestran en los ejemplos a continuación.

Ejercicio 1 (teoría) - 5 puntos

1. Descarga ahora el archivo `notas.csv`
2. Evalúa Apellido1Apellido2.py desde Spyder con la opción 1 (cuando te la pida).

   Verás la siguiente salida

```
In [1]: runfile('/Users/apple/Downloads/ejercicio1.py', wdir='/Users/apple/Downloads')

Indica el ejercicio que deseas ejecutar (1, 2 o 3): 1

######   E J E R C I C I O --- 1   ######

El mejor estudiante es Id0 con media 10.0
El peor estudiante es Id99 con media 0.0
```

3. Cuando esté bien resuelto el ejercicio verás una salida como la siguiente.

```
In [3]: runfile('/Users/apple/Downloads/examen_PROG1.py', wdir='/Users/apple/Downloads')

Indica el ejercicio que deseas ejecutar (1, 2 o 3): 1

######   E J E R C I C I O --- 1   ######

El mejor estudiante es Id17 con media 9.5
El peor estudiante es Id31 con media 3
```

¿Qué debes hacer?

Verás que el ejercicio 1 tiene TRES funciones.

```
def leeNotasClase(archivo):
def mejorEstudiante(lista):
def peorEstudiante(lista):
```

La primera se te entrega ya codificada. A partir del archivo con los datos (`notas.csv`), lee los datos y devuelve sobre la variable:

- `idsYnotas`: contiene una lista de listas con los IDs y las notas de cada estudiante

Tu misión es codificar las funciones `mejorEstudiante` y `peorEstudiante`, conforme a las descripciones de ambas en los *docstring*. El objetivo es calcular la calificación media del mejor y del peor estudiante, para producir una salida como la mostrada arriba.

Para calcular la media puedes usar la librería `statistics`, que se importa al inicio. Dentro de esa librería, la función `mean(lista)`, calcula y devuelve la media de los valores de la lista que se pasa como argumento.

Ejercicio 2 (teoría) - 5 puntos

- Evalúa `Apellido1Apellido2.py` desde Spyder con la opción 2 (cuando te la pida).
- El programa devolverá un error. El objetivo es que logres ver una salida como la siguiente (se muestra el resultado del ejercicio 2).

```
In [5]: runfile('/Users/apple/Downloads/examen_PROG1.py', wdir='/Users/apple/Downloads')

Indica el ejercicio que deseas ejecutar (1, 2 o 3): 2

######   E J E R C I C I O --- 2   ######
Ciudad: Madrid
Número de líneas: 1

Línea 5:
- Gran Vía (correspondencias: [1])
- Chueca (correspondencias: [])
- Alonso Martinez (correspondencias: [4, 10])
```

¿Qué debes hacer?

Verás que el ejercicio 2 tiene 3 clases, con sus respectivas descripciones de los atributos y los métodos que las componen (la clase `RedMetro` carece de métodos, solo tiene atributos).

```
class EstacionMetro:
class LineaMetro:
class RedMetro:
```

Debes codificar los constructores (con los atributos indicados en los *docstring* respectivos a cada clase), y los métodos, para que el programa funcione y produzca la salida que se muestra más arriba. También se muestra una imagen de las estaciones y líneas descritas en el ejemplo (Gran Vía, Chueca y Alonso Martínez, de la línea 5, la verde).

Ejercicio 3 (prácticas) - 8 puntos

1. Descarga los siguientes archivos correspondientes a la práctica 2:
   a. `gestion.py`
   b. `Ap1Ap2_aux_gestion`
   c. `Ap1Ap2_persona.py`
   d. `ciudadanos.txt`
   e. `contratar.txt`
2. Debes completar el método `eliminaCiudadano` en `Ap1Ap2_aux_gestion.py` para eliminar un ciudadano del diccionario de personas, de modo que, si se listan con las opciones 2 o 3, ya no se muestre más, independientemente de si tenía fecha de contrato o no. La búsqueda del ciudadano que hay que eliminar se realiza por DNI.
3. Debes modificar la clase persona para que se lean los nuevos datos que figuran en el archivo `ciudadanos.txt`. Los nuevos datos son, como verás en las salidas, el salario (en miles de euros) y el departamento al que se adscribe la persona.

A continuación, se muestran imágenes de cómo debe ser la salida del programa.

```
In [1]: runfile('/Users/FDR/Downloads/practica2/gestion.py', wdir='/Users/FDR/Downloads/
practica2')
============================================================
=                                                          =
=       SELECCIONE UNA OPCIÓN DEL MENÚ                     =
=       ------------------------------                     =
=                                                          =
=       1. CONTRATAR CIUDADANOS                            =
=       2. IMPRIMIR CIUDADANOS CONTRATADOS                 =
=       3. IMPRIMIR CIUDADANOS SIN CONTRATO                =
=       4. BUSCAR CIUDADANO                                =
=       5. ELIMINAR CIUDADANO (nueva opción)               =
=       9. TERMINAR                                        =
=                                                          =
============================================================
TECLEE EL NÚMERO DE LA OPCIÓN DESEADA: 1
Se ha leído el fichero de ciudadanos.txt
Se ha contratado con fecha 29/05/2024 a los ciudadanos
cuyo DNI figura en el fichero de contratar.txt

Pulse ENTER para continuar
```

La opción 1 lee el archivo de ciudadanos, genera el diccionario de personas y contrata (pone fecha de contrato) a los ciudadanos cuyo DNI figura en el archivo de `contratar.txt`

La opción 2 lista los ciudadanos contratados. En esta imagen solo se muestran dos de los cinco.

```
Pulse ENTER para continuar
============================================================
=                                                          =
=       SELECCIONE UNA OPCIÓN DEL MENÚ                     =
=       ------------------------------                     =
=                                                          =
=       1. CONTRATAR CIUDADANOS                            =
=       2. IMPRIMIR CIUDADANOS CONTRATADOS                 =
=       3. IMPRIMIR CIUDADANOS SIN CONTRATO                =
=       4. BUSCAR CIUDADANO                                =
=       5. ELIMINAR CIUDADANO (nueva opción)               =
=       9. TERMINAR                                        =
=                                                          =
============================================================
TECLEE EL NÚMERO DE LA OPCIÓN DESEADA: 2

Nombre:  Laura
Apellido:  Garcia
NIF: 52736428P
Teléfono:  638233312
CCC:  ES71 1234 5678
Contrato: 29/05/2024
Salario:  1.85 euros (en miles)
Departamento:  Recursos Humanos

Nombre:  Paula
Apellido:  Lopez
NIF: 50123425H
Teléfono:  639675513
CCC:  ES31 9182 3546
Contrato: 29/05/2024
Salario:  1.85 euros (en miles)
Departamento:  Gestion Economica
```

```
============================================================
=                                                          =
=       SELECCIONE UNA OPCIÓN DEL MENÚ                     =
=       ------------------------------                     =
=                                                          =
=       1. CONTRATAR CIUDADANOS                            =
=       2. IMPRIMIR CIUDADANOS CONTRATADOS                 =
=       3. IMPRIMIR CIUDADANOS SIN CONTRATO                =
=       4. BUSCAR CIUDADANO                                =
=       5. ELIMINAR CIUDADANO (nueva opción)               =
=       9. TERMINAR                                        =
=                                                          =
============================================================
TECLEE EL NÚMERO DE LA OPCIÓN DESEADA: 5
Teclea el DNI a buscar: 52736428P
Pulse ENTER para continuar
```

La opción 5, nueva, permite hacer búsquedas por la clave DNI en el diccionario y, caso de que dicha clave figure en el diccionario, se elimina del mismo dicha clave y su correspondiente valor (la persona asociada con ese DNI). En el caso del ejemplo eliminamos el DNI 52736428P correspondiente a Laura García. Dicha persona ya no aparecerá en ningún listado.

Finalmente, volvemos a listar los ciudadanos con fecha de contrato y observamos que ya no figura Laura García, dado que ha sido eliminada en el paso anterior.

```
Teclea el DNI a buscar: 52736428P
Pulse ENTER para continuar
============================================================
=                                                          =
=       SELECCIONE UNA OPCIÓN DEL MENÚ                     =
=       ------------------------------                     =
=                                                          =
=       1. CONTRATAR CIUDADANOS                            =
=       2. IMPRIMIR CIUDADANOS CONTRATADOS                 =
=       3. IMPRIMIR CIUDADANOS SIN CONTRATO                =
=       4. BUSCAR CIUDADANO                                =
=       5. ELIMINAR CIUDADANO (nueva opción)               =
=       9. TERMINAR                                        =
=                                                          =
============================================================
TECLEE EL NÚMERO DE LA OPCIÓN DESEADA: 2

Nombre:  Paula
Apellido:  Lopez
NIF: 50123425H
Teléfono:  639675513
CCC:  ES31 9182 3546
Contrato: 29/05/2024
Salario:  1.85 euros (en miles)
Departamento:  Gestion Economica

Nombre:  Nerea
Apellido:  Lupia
NIF: 07937734T
Teléfono:  666998877
CCC:  ES54 1234 9876
Contrato: 29/05/2024
Salario:  2.6 euros (en miles)
Departamento:  Ciencia de Datos
```

Para eliminar elementos de un diccionario puedes emplear el método `del(clave)` de la clase diccionario.

Ejercicio 4 (prácticas) - 2 puntos

Evalúa `Apellido1Apellido2.py` con la opción 3. Verás que produce una salida de 25 unos.

En combinatoria, los números de Catalan[1] forman una secuencia de números naturales que aparece en varios problemas de conteo que habitualmente son recursivos. Su nombre hace referencia al matemático belga Eugène Charles Catalan (1814-1894). En este ejercicio se trata de que calcules el enésimo número de Catalan.

El n-ésimo número de Catalan se obtiene a partir de alguna de las siguientes fórmulas:

$$C_n = \frac{(2n)!}{(n+1)!\,n!} = \prod_{k=2}^{n} \frac{(n+k)}{k} \; para\; n \geq 0$$

Debes codificar, en el archivo `Apellido1Apellido2.py`, la función de prototipo que allí se indica y que calcula e imprime hasta el n-ésimo número de Catalan.

Puedes escoger para codificar la fórmula que desees, con el factorial o con el *productorio*. Si deseas implementar la fórmula con la función factorial, puedes hacer uso del método `factorial` de la librería `math`.

A continuación se muestra la salida que produce la ejecución del módulo `Apellido1Apellido2.py` si se selecciona la opción 3. Corresponde a los números de Catalan del 0 al 25 una vez que esté bien codificada la función.

[1] https://en.wikipedia.org/wiki/Catalan_number

Grado en análisis de datos en la empresa
PROGRAMACIÓN I
2º examen parcial - 29/05/2024

Universidad Autónoma de Madrid

```
In [1]: runfile('/Users/apple/Downloads/catalan.py', wdir='/Users/apple/Downloads')
1
1
2
5
14
42
132
429
1430
4862
16796
58786
208012
742900
2674440
9694845
35357670
129644790
477638700
1767263190
6564120420
24466267020
91482563640
343059613650
1289904147324
4861946401452
```