

Examen de Programación Julio 2022 (Grados en Física y Matemáticas)

Primera parte (1.25 puntos por cuestión, 50% nota del examen)

- 1) Escribir una función que calcule el plan más económico que se ajuste a un servicio demandado por el cliente, para la inscripción a un gimnasio, de acuerdo a la siguiente tabla de planes y servicios.

		Plan		
		Básico	Confort	Premium
	Precio 4 semanas	14,99	19,99	29,99
Servicio	Asistente virtual	✗	✓	✓
	Clases virtuales	✗	✓	✓
	Invitar amig@	✗	✗	✓
	Descuento Sportswater	✗	✗	✓

La función recibe como parámetro un string con el servicio demandado (texto de tamaño nulo si no necesita ningún servicio, o uno de los cuatro servicios ofertados) y retorna el nombre del plan más económico que incluye dicho servicio (Básico, Confort o Premium).

Si el servicio demandado no es un texto de tamaño nulo o uno de los servicios ofrecidos, retornar el texto "Error".

Si el servicio demandado no es un texto de tamaño nulo o uno de los servicios ofrecidos, retornar el texto "Error".

Las comparaciones de strings deben hacerse de modo que no influyan mayúsculas y minúsculas.

Se valora la eficiencia y tamaño compacto del código.

- 2) Escribir una función a la que se le pasa una lista de números reales y retorna una segunda lista con los elementos de la lista original quitando los cinco mayores y los cinco menores. Si la lista original tiene 10 valores o menos, se retornará una lista vacía.
- 3) Se dispone de la función python $f(x)$ que tiene el parámetro real x y retorna un número real. Se desea calcular la integral definida de esta función entre los valores a y b por medio del método del trapecio compuesto, de acuerdo a la siguiente expresión:

$$\int_a^b f(x) dx \cong \left(\frac{b-a}{n} \right) \left(\frac{f(a)+f(b)}{2} + \sum_{k=1}^{n-1} f\left(a + k \frac{b-a}{n}\right) \right)$$

La función, que se encuentra en el mismo módulo que $f(x)$, recibe como parámetros los valores de a y b (reales) así como el número de intervalos de cálculo, n , y retorna el valor de la integral entre a y b . Si el número de intervalos n es demasiado pequeño, inferior a 10, se lanzará la excepción `PocosIntervalos`, ya importada.

- 4) Dentro del directorio del usuario se dispone de un directorio llamado `imagenes` que contiene entre otras cosas un par de directorios: por un lado el directorio `raw` que contiene imágenes originales en archivos con la terminación `".raw"`, y por otro lado el directorio

filtered, que contiene imágenes filtradas en archivos con la terminación ".jpg". Ambos directorios pueden tener también archivos de texto terminados en ".txt".

Por otro lado, se dispone de un pen drive situado en la ruta absoluta /mnt/drive.

Se desea escribir las órdenes linux/unix necesarias para hacer una copia selectiva de ficheros de la carpeta imagenes al pen drive. Se supone que el directorio de trabajo inicial es el del usuario. Se pide utilizar exclusivamente *rutas relativas* para las carpetas y ficheros contenidos en el directorio del usuario. Por el contrario, utilizar exclusivamente rutas absolutas para referirse a directorios y ficheros situados en el pen drive.

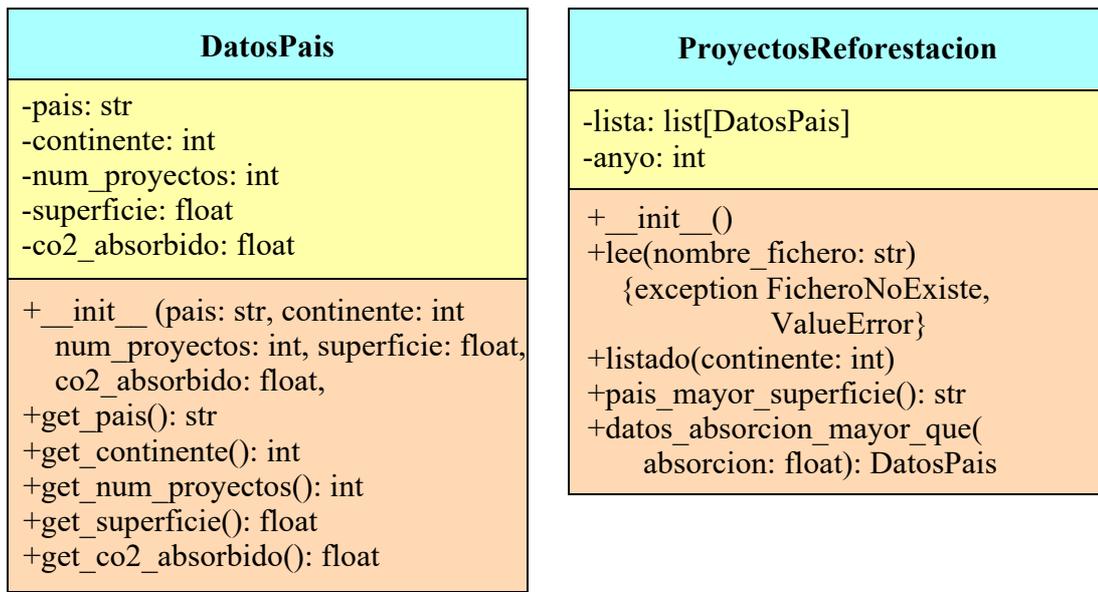
Se pide realizar los siguientes pasos:

- a) borrar del pen drive el directorio sel2021 con todos sus contenidos
- a) crear en el pen drive el directorio sel2022
- b) copiar todos los ficheros acabados en ".raw" del directorio raw a sel2022
- c) mover todos los ficheros acabados en ".txt" de raw a sel2022
- d) crear dentro de sel2022 un directorio llamado filt
- e) copiar todos los ficheros acabados en ".jpg" del directorio filtered al directorio filt
- f) borrar todos los ficheros que estén directamente en el directorio imagenes y que tengan en su nombre la secuencia "README"
- g) cambiar de nombre al directorio imagenes, poniéndolo a imagenes_procesadas

Examen de Programación Julio 2022 (Grados en Física y Matemáticas)

Segunda parte (5 puntos, 50% nota del examen)

Se dispone en el módulo `reduccion_co2.py` de una clase llamada `DatosPais` que permite guardar los datos de proyectos de reducción de CO2 en la atmósfera mediante reforestación, para un determinado país. La clase responde al siguiente diagrama de clases:



Los métodos de la clase `DatosPais` hacen lo siguiente:

- *constructor*: crea el objeto poniendo los datos del país a los valores indicados
- `get_pais()`: retorna el nombre del país
- `get_continente()`: retorna el número del continente. (0=Asia, 1=Europa, 2=África, 3= Norteamérica, 4= Sudamérica, 5=Oceanía)
- `get_num_proyectos()`: retorna el número de proyectos de reforestación
- `get_superficie()`: retorna la superficie total reforestada, en hectáreas (ha)
- `get_co2_absorbido()`: retorna las toneladas equivalentes de CO2 absorbido anual

Lo que se pide es escribir en el mismo módulo `reduccion_co2.py` una clase llamada `ProyectosReforestacion` que responda al diagrama de clases que se muestra arriba y que contenga como atributo una lista de objetos de la clase `DatosPais` con datos de proyectos de reforestación para diversos países. La clase tiene otro atributo que almacena el año al que se refieren los datos. Los métodos de la clase deben hacer lo siguiente:

- *constructor*: crea la lista vacía y pone a valor cero el atributo que almacena el año.
- `lee()`: lee los datos de proyectos de reforestación del fichero cuyo nombre se pasa como argumento y los añade al atributo `lista`. El fichero tiene en su primera línea la palabra "Año" y el año al que se refieren los proyectos, separados por una coma.

La segunda línea es un encabezamiento explicativo que se ignora.

A continuación el fichero tiene los datos de los países, un país por línea. En cada línea los datos se separan por comas y están en este orden: País, Continente, Num. proyectos, Superficie (ha), CO2 absorbido (t)

Se muestra un ejemplo de fichero:

```
Año,2022
País,Continente,Num proyectos,Superficie (ha),CO2 absorbido (t)
Italia,1,9,2508.0,6962.0
Senegal,2,1,1411.0,480.0
Uganda,2,1,12186.0,100.0
...
```

Si el fichero no existe se lanzará automáticamente IOError. Debemos tratar esta excepción poniendo un mensaje descriptivo del error que ha ocurrido y lanzando la excepción FicheroNoExiste, que hay que crear en el mismo módulo.

Si el fichero tiene errores se lanzará automáticamente ValueError. Debemos tratar esta excepción poniendo un mensaje descriptivo del error que ha ocurrido y volviendo a lanzar la misma excepción.

Pista: para separar una línea leída del fichero en sus datos usando un separador, puede usarse el método split() de los strings. Por ejemplo, si el string se llamase linea y el separador fuese "/":

```
linea.split("/")
```

- listado(): Hace un listado de los datos de reforestación de los países del continente indicado en el parámetro. Para este listado primero se ponen dos líneas de encabezamiento con:
 - Frase introductoria, continente (con letra) y año de los datos.
 - Explicación de las columnas con los datos

A continuación se ponen los datos de cada país en columnas, un país por línea, con su nombre, número de proyectos, superficie y CO2 absorbido anual. Poner los números reales con un decimal. Este es un ejemplo del formato:

```
Datos de reforestación del continente: África, en el año 2022
País                Núm. proyectos Superficie (ha) CO2 absorbido (t)
Senegal              1                1411.0           480.0
Uganda               1                12186.0          100.0
Tanzania             1                350.0            11167.0
...
```

Si no hubiese ningún dato para el continente indicado, se pondrá el texto "No hay datos para este continente" tras las dos líneas de encabezamiento.

- pais_mayor_superficie(): Calcula el país de mayor superficie reforestada. Retorna el nombre del país.
- datos_absorcion_mayor_que(): Busca y retorna el primer objeto de la lista que contiene un equivalente de absorción de CO2 mayor que el parámetro absorcion. Si no existe ninguno retorna None.

Se pide, además, escribir un programa principal que pruebe todos los métodos de la clase ProyectosReforestacion a partir de los datos del fichero de datos llamado "reforestacion.csv". Si se lanza FicheroNoExiste se deben abandonar todos los pasos restantes del programa y no hacer nada más, es decir poner un tratamiento de excepción vacío. No es preciso tratar la excepción ValueError. Con una sola llamada a cada método es suficiente.

Valoración:

- 1) Encabezamiento de la clase, excepción `FicheroNoExiste` y Constructor: (0,5 puntos)
- 2) `lee`: (1 punto)
- 3) `listado`: (1 punto)
- 4) `pais_mayor_superficie`: (0,75 puntos)
- 5) `datos_absorcion_mayor_que`: (0,75 puntos)
- 6) `main`: (1 punto)