

# Fundamentos de la Programación. Grado en Física - Curso 2022/2023

## Examen de teoría de la convocatoria C2 (11/01/2023)

### Instrucciones

1. Tienes 3 horas para realizar el examen. Lee tranquilamente el examen y decide por dónde empezar.
2. No puedes utilizar apuntes ni programas ya escritos por ti o por otros para hacer el examen. Tampoco puedes comunicarte con nadie (excepto con el profesor) durante el examen. Puedes consultar al profesor por instrucciones concretas de Python que no recuerdes y puedes consultar los materiales disponibles en MoodleUA.
3. Finalmente debes entregar tu examen con todos los ejercicios que hayas realizado en un archivo comprimido en zip cuyo nombre debe ser tu DNI (NUMERO\_DNI.zip).
4. Debes entregar dicho archivo comprimido a través de la tarea creada en MoodleUA. Dicha entrega se cerrará automáticamente a la hora fijada para la finalización del examen, de modo que debes tener cuidado en no retrasarte.

### Ejercicios

Disponemos de un archivo de texto que contiene tres columnas de números separados por uno o varios espacios en blanco. Los números de la primera columna son enteros, los de las otras dos son reales. El archivo por otra parte puede contener líneas en blanco. Un ejemplo del contenido de dicho archivo puede ser

```
1  2.1 3.2
-3  4.2 3e-2
3   4.2 -9.5
```

```
1  3.1 0
2  4.3 -2
```

Implementa un programa en Python llamado `examen.py` en el que se definan las funciones y realice las tareas que se describen a continuación:

- `leeDatos(nombre)` (2 puntos): función que lee los datos de un archivo del tipo anterior cuyo nombre se le pasa en el parámetro `nombre` y devuelve tres listas, cada una conteniendo los números de la columna correspondiente. Si el número de líneas con números del archivo no es al menos de 4 debe emitir el mensaje de error `Error: archivo incorrecto` y devolver tres `None`. Para el archivo del ejemplo el contenido de las listas devueltas sería

```
u1 -> [1, -3, 3, 1, 2]
u2 -> [2.1, 4.2, 4.2, 0., 4.3]
u3 -> [3.2, 0.03, -9.5, 0., -2.]
```

No es necesario que compruebes que el archivo está disponible y que tiene el formato correcto, únicamente si tiene el número mínimo exigido de líneas con números.

- `sumaDobleMinimo(v)` (1 punto): función que devuelve una lista obtenida sumando a cada elemento de la lista de números enteros `v`, que se pasa como parámetro, el doble del mínimo de los elementos de `v`. Por ejemplo, la llamada `sumaDobleMinimo(u1)` debe devolver la lista

```
[-5, -9, -3, -5, -4]
```

- `primoI(v)` (2 puntos): devuelve un vector en el que la componente en la posición `i` es el `i`-ésimo número primo si `vi` es mayor o igual que 0, o 0 en otro caso. Si `v = [0, 1, -2, 3, 4]` el vector devuelto será `[0, 2, 0, 5, 7]`.
- `mediaSinI(v)` (1 punto): devuelve un vector en el que la componente en la posición `i` es la media de todos los elementos de `v` excepto el que ocupa la posición `i`. Ejemplo: si `v = [0, 1, 2, 3, 4]`, el vector devuelto será `[2.5, 2.25, 2., 1.75, 1.5]`, (por ejemplo, en la segunda posición aparece la media de 0, 1, 3 y 4).

■ Finalmente en el programa principal se deben realizar las siguientes tareas:

1. Leer el archivo de nombre `datos.dat` con el formato descrito más arriba llamando a la función `leeDatos("datos.dat")`. Si los valores devueltos son `None` el programa debe terminar sin realizar ninguna otra acción.
2. (2 puntos) Si la lectura del archivo no es `None` el programa escribirá el valor devuelto por las otras tres funciones (a las que se le pasará como parámetro la lista con los valores de la primera columna) en el formato siguiente:

```
suma doble minimo: -5 -9 -3 -5 -4
i-esimo primo: 2 0 5 2 3
media sin i: 0.75 1.75 0.25 0.75 0.5
```

Si no has sido capaz de implementar alguna de las funciones debes imprimir 0's en la línea correspondiente. La salida será por pantalla si no se le pasa al programa ningún parámetro por línea de comandos y se escribirá en un archivo si el nombre de éste se pasa como parámetro por línea de comandos. Es decir, el programa podrá ser ejecutado desde consola de dos maneras diferentes:

```
$ python3 examen.py # salida por pantalla
$ python3 examen.py salida.txt # salida escrita en el archivo salida.txt
```

3. (2 puntos) Finalmente el programa creará una gráfica que contendrá dos tipos de líneas:
  - a) una con puntos cuya coordenada X serán los números de la segunda columna del archivo `datos.dat` y cuya coordenada Y serán los de la tercera.
  - b) la otra línea representará la solución de la ecuación diferencial

$$\frac{dy}{dx} = \frac{x^2 - 1}{y^2}$$

con la condición inicial  $y(0) = 8$  con 100 puntos en el intervalo  $[0, 6]$ .

Dicha gráfica se guardará con el nombre `grafica.png`. Para los datos del ejemplo la gráfica sería:

