

DEBES ENTREGAR EN HOJAS DIFERENTES CADA PROBLEMA

### Problema 1 (5 puntos)

Una *matriz de producción de energía* es una matriz de  $N \times M$  valores flotantes que registra la energía producida por los  $N$  paneles solares de una planta fotovoltaica durante  $M$  días consecutivos. La celda  $[i][j]$  de esta matriz almacena la cantidad de kWh producida por el panel solar  $i$  durante el día  $j$ .

Implementa en Python:

- a) (1 PUNTO) Una función, **producción\_diaria**, que tenga como parámetro una matriz de producción de energía y devuelva como resultado una lista con la cantidad total de energía producida durante cada día.
- b) (1 PUNTO) Una función, **peor\_día**, que tenga como parámetro una matriz de producción de energía y devuelva como resultado el día en el que se produjo menos energía. En caso de empate entre varios días, la función debe elegir el último de ellos. Esta función debe llamar a la función anterior.
- c) (2 PUNTOS) Una función, **defectuoso**, que tenga como parámetros una matriz de producción de energía, un número de panel  $p$  y un número entero  $n$ . La función debe devolver **True** cuando el panel indicado no haya producido energía durante más de  $n$  días consecutivos y **False** en cualquier otro caso.
- d) (1 PUNTO) Una función, **panel\_más\_rentable**, que tenga como parámetro una matriz de producción de energía y devuelva como resultado el número del panel que ha producido la mayor cantidad total de energía.

## Problema 2 (5 puntos)

La Consejería de Sanidad Universal de la Generalitat Valenciana almacena en un fichero de texto los diagnósticos realizados durante cada día por su personal médico con respecto a tres enfermedades respiratorias contagiosas: *Gripe Común*, *Gripe A* y *Covid-19*. Cada línea de estos ficheros contiene, separados por una almohadilla, el número de SIP del paciente, el código del centro de salud correspondiente y el diagnóstico realizado. A continuación se muestran algunas líneas de ejemplo:

```
07701992#12001#Gripe común
12340005#46023#Gripe A
77771010#46007#Covid-19
75431234#12005#Covid-19
91911002#12001#Gripe A
89391234#12005#Covid-19
59931341#12001#Gripe común
```

Disponemos de la clase `CentroSalud`, que está definida como sigue:

```
class CentroSalud:
    def __init__(self, código):
        self.código = código # Código del centro de salud
        self.gripe_común = 0
        self.gripe_A = 0
        self.covid19 = 0
```

Como podemos ver, los objetos de la clase `CentroSalud` almacenan en el atributo `código` el código de un centro de salud y contienen además tres contadores (inicializados a cero cuando se crea el objeto) para cada una de las tres enfermedades que nos interesan.

Implementa en Python:

- a) (3 PUNTOS) Una función, `crear_lista`, que reciba como parámetro el nombre de un fichero de datos y que devuelva como resultado una lista de objetos. Cada objeto de la lista devuelta debe contener el código de un centro de salud y la cantidad total de pacientes diagnosticados con cada una de las tres enfermedades en dicho centro. Solo se puede recorrer el fichero dado una vez.

*Nota: Recuerda que `cadena.split(carácter)` devuelve una lista con las subcadenas de `cadena` usando `carácter` como separador.*

- b) (2 PUNTOS) Una función, `centro_con_másincidencia`, que reciba como parámetros una lista como la construida en el apartado anterior y el nombre de una enfermedad y que devuelva como resultado el código del centro de salud con una mayor cantidad de diagnósticos de la enfermedad dada. En caso de que varios centros de salud empaten en el valor máximo, la función puede devolver cualquiera de ellos. Si no hay ningún diagnóstico de la enfermedad dada en ningún centro de salud, la función debe devolver `None`.