

	<p align="center"> 2º EVALUACION 1º DAM CURSO 2020 - 2021 Programación 1º Examen 12/02/2021 – (8:30 a 11:30 h) </p>	
Nombre, apellidos y DNI:		Calificación

Puedes usar internet sin limitaciones, salvo para comunicarte con tus compañeros de clase durante el examen, estaremos vigilando, y la comunicación supondrá el suspenso. Solo puedes usar el ordenador, ni teléfonos ni smartwatch están permitidos. El examen se entregará a través de Edmodo en una entrega que se abrirá antes del final del examen. Si no te sale uno, pasa a hacer el siguiente, y luego vuelves a él: No pierdas el tiempo innecesariamente.

Realiza todo el examen en el mismo proyecto, pon TODOS los métodos en un archivo distinto al archivo donde guardes el main.

(-1,5 Puntos) Si algún método está en el mismo archivo que el main

(-0,2 Puntos) Por cada variable, método o clase que no se nombre con una notación correcta (UpperCamelCase o lowerCamelCase)

(-0,1 Puntos) Por cada variable que ocupe un tamaño mayor (o insuficiente) del necesario para almacenar los valores que puede tener dentro.

Vamos a construir un generador de mapas aleatorio. En nuestros mapas solo podrá haber dos cosas: posiciones con agua, y posiciones con terreno. Representaremos estos mapas en una matriz de char, donde el carácter '~' representará una zona con agua, y el carácter '#' representará una zona con tierra. Por ejemplo, un mapa generado de tamaño 10*15, podría representarse en una matriz así:

```

~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ # # ~ ~
~ ~ ~ ~ ~ ~ ~ ~ # # ~ ~
~ ~ ~ # # # ~ ~ ~ # # ~ ~
~ ~ ~ # # # ~ ~ ~ # # ~ ~
~ ~ ~ # # # ~ ~ ~ # # ~ ~
~ ~ ~ ~ # ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

```

Lo construiremos así:

1 - **(0,25 Puntos)** En el main, pide dos números por teclado. El primero será el número de filas del mapa, y el segundo el número de columnas. Asegúrate de que tanto filas como columnas tienen al menos longitud 10, y máximo 127. De no ser así, vuelve a pedir los números las veces que sea necesario, hasta que la condición se cumpla.

2 - **(2 Puntos)** Crea una función `matrizDeAgua`, que reciba por argumentos dos números (`numeroFilas` y `numeroColumnas`), y devuelva una matriz de char. En su interior, usará esos dos argumentos para crear una matriz de char con el número de filas y columnas indicado en los valores de los argumentos. Rellenará todas las posiciones de esa matriz con el carácter '~'.

3 - **(0,25 Puntos)** En el main, tras pedir los dos números, llama a la función `matrizDeAgua` usándolos como argumentos. Almacena el valor devuelto por la función en una variable llamada `mapa`. Tras lo anterior, en el main, pide un número por teclado (`nContinentes`), que representará el número de masas de tierra (continentes) que quieres que haya en el mapa. Asegúrate de que el número `nContinentes` es menor que el número total de posiciones de la matriz. De no ser así, vuelve a pedirlo las veces que sea necesario, hasta que la condición se cumpla.

4 - **(2 Puntos)** Crea una función llamada `ponerTerreno`. Recibirá como argumentos una matriz de char (`mapa`), y un entero (`nContinentes`). No devolverá nada. En su interior, elegirá tantas posiciones (fila y columna) aleatorias como `nContinentes` indique el argumento `nContinentes`. Cuando elija cada una de las posiciones, colocará en ella un '#'. Se asegurará de que en la posición escogida aleatoriamente haya un carácter '~' antes de poner el '#'. Si en esa posición ya había un carácter '#', volverá a elegir de forma aleatoria la posición.

Por ejemplo, si tenemos una matriz de 10*15, y `nContinentes` vale 5, una posible salida sería esta:

```
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ # # ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ # ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ # ~ ~ ~ ~ ~ ~
# ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
```

5 - **(0,25 Puntos)** En el main, con todo lo que habías pedido por teclado, llama a la función `ponerTerreno`, e imprime la matriz por consola. Después pide por teclado un número llamado `extensionMaxima`. Asegúrate que está entre 2 y 5. Si no es así, vuelve a pedirlo las veces que sea necesario.

6 - **(1,5 Puntos)** Haz que tu programa admita los argumentos de programa -filas, y -columnas. Ambos vendrán seguidos en la línea de comandos de un número. Estos argumentos podrán estar o no estar, o ponerse en cualquier orden, y el programa debería funcionar. Un ejemplo de la llamada desde consola sería:

```
java -jar miPrograma.jar -columnas 12 -filas 8
```

Si recibe el argumento -filas, ya no pedirá el número de filas por teclado en el main, como hacías en el ejercicio 1, y utilizarás el valor dado por argumentos. A no ser que ese valor no esté entre 10 y 127. En ese caso, pide el valor por teclado de todas formas, como en el ejercicio 1. el argumento -columnas hará lo mismo, pero con las columnas.

7 - **(2,25 Puntos)** crea una función llamada extenderTerreno, que reciba por argumentos una matriz de char (mapa). En su interior, hará lo siguiente: Buscará todas las posiciones que contengan el carácter '#'. Cada vez que encuentre una, tendrá un 40% de probabilidad de colocar un carácter '%' en todas las casillas adyacentes que no contengan ya el carácter #. Las casillas adyacentes son las que limitan con ella. Por ejemplo, en esta matriz:

```
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ # # ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ # ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ # ~ ~ ~ ~ ~ ~
# ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
```

La primera vez que encuentra un #, es en la posición (2,3). Sus adyacentes son las posiciones (1,2), (1,3), (1,4), (2,2), (2,4), (3,2), (3,3) y (3,4). Es decir, las de alrededor:

```
~ ~ ~
~ # #
~ ~ ~
```

En este caso, colocará el carácter '%' con un 40% de probabilidad en cada una de esas 8 posiciones que tiene alrededor (adyacentes), y que no contienen ya un carácter #. Podría quedar algo como:

```
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ % % ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ # # ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ % ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ # ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ # ~ ~ ~ ~ ~ ~
# ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~
```

Ten en cuenta a los valores extremo, por ejemplo, para el carácter # que está en la posición (8,0), no tiene ninguna casilla adyacente a su derecha, y solo tiene 5 casillas adyacentes. En este caso, harás el sorteo para

ver si se pone el carácter % solo en esas 5 casillas adyacentes. Es decir, evita que esta función te de un `IndexOutOfBoundsException` :)

Una vez haya colocado los % correspondientes a todos los caracteres # que había en el mapa, la función volverá a recorrer toda la matriz sustituyendo todos los carácter '%' que encuentre, por el carácter '#'.

8 - **(0,25 Puntos)** En el main, tras pedir por teclado el número `extensionMaxima`, úsalo para llamar a la función `extenderTerreno` tantas veces como se haya indicado en ese número `extensionMaxima`. Después, imprime la matriz resultante por pantalla.

9 - **(1,25 Puntos)** Documenta correctamente todo tu código fuente. No tienes por qué tener todo el examen terminado para hacer este ejercicio. Si lo que tienes está bien documentado, este ejercicio tendrá la máxima nota.

CRITERIOS DE CALIFICACIÓN

Los ejercicios que los alumnos no hayan hecho, o estén comentados, o tan errados que no tienen nada que ver con lo pedido, no contarán para la nota. Cada ejercicio que hayan hecho, o al menos intentado los alumnos, cuenta su nota completa, y se van restando esta cantidad de puntos según los fallos cometidos:

- **-0,5 puntos** Si hay algún error que impide compilar o que hace reventar la ejecución del programa, o lo mete en una situación de bloqueo permanente, o bucle infinito.
- **-0,25 puntos** Si hay algún error que puede ocasionar fallos en ciertas situaciones muy específicas, o hacen que la salida del programa sea errónea aunque este termine sin reventar.
- **-0,1 puntos** Si en el ejercicio 3, no se calcula bien el número de continentes máximo posible.
- **-0,1 puntos** En el ejercicio 1, por cada uno de los límites que no se comprueban bien.
- **-0,15 puntos** En el ejercicio 1, si las comprobaciones se hacen mal
- **-0,15 puntos** Si en el ejercicio 5, lo único que se hace es llamar a la función.
- **-0,15 puntos** Si en el ejercicio 5, no se comprueba que esté entre 2 y 5
- **-0,25 puntos** Si en el ejercicio 4 no se comprueba bien que no se ponga una # sobre otra #
- **-0,75 puntos** Si en el ejercicio 4 no se tiene en cuenta el nº de continentes, y se ponen más de los pedidos.
- **-0,5 puntos** Si en el ejercicio 7, se intentan mirar los valores extremo, pero da `IndexOutOfBoundsException` en algunas situaciones.
- **-0,1 puntos** Si en el ejercicio 8, no se imprime la matriz.
- **-0,15 puntos** En el ejercicio 9 por cada función no documentada
- **-0,1 puntos** En el ejercicio 9 por cada sección no documentada
- **-0,5 puntos** El ejercicio 2 se hace fuera de una función
- **-0,15 puntos** En el ejercicio 3 no se pide `nContinentes`
- **-0,15 puntos** En el ejercicio 8 no llamas a `extenderTerreno`
- **-0,15 puntos** Si se modifica la cabecera de una de las funciones