

Ejercicio Tetris

Crea una clase llamada Tetris.java que implemente la siguiente interfaz

```
public interface Arcade {  
    public ArrayList<Integer> lineasTetris(int m[][]);  
    public int[][] tableroToMatriz(String fichero);  
    public int insertaPuntuacion(Jugador a);  
    public int mejorRacha(ArrayList<Partida> listaPartidas);  
    public int mediaPais(String pais, ArrayList<Jugador> jugadores);  
}
```

Ejercicio 1: (2 puntos): Implementa el método lineasTetris que recibe como parámetro una matriz que representa el tablero de un juego de Tetris (de cualquier número de filas y columnas) y retorna como resultado un ArrayList con las filas donde se haya producido una “línea”. Cada elemento de la matriz tablero tendrá un número entero que representa si está ocupada por un bloque o no. Cada número entero podrá ser:

0- casilla vacía

1- casilla ocupada por un bloque rojo

2- casilla ocupada por un bloque verde

3- casilla ocupada por un bloque azul

En el juego del Tetris, se considera que se ha conseguido una “línea” cuando todas las casillas de la misma fila de la matriz están ocupadas por bloques.

Por ejemplo, dada la matriz A:

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
1	0	3	0	0
1	3	2	1	2
1	2	2	3	1
0	2	1	0	1
2	3	1	1	2

→ Línea en fila 4!

→ Línea en fila 5!

→ Línea en fila 7!

El método debe retornar el ArrayList

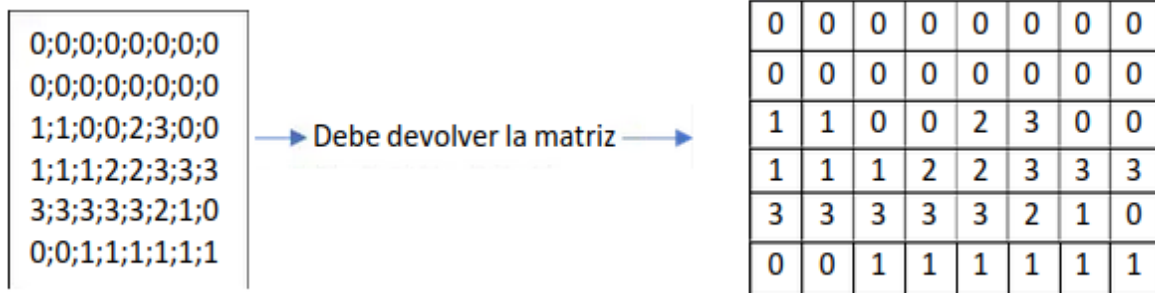
4	5	7
---	---	---

Ejercicio 2 : (2 puntos): Implementa el método tableroToMatriz que genera una matriz a partir del fichero que se pasa como parámetro. El fichero tendrá números enteros separados por ; al estilo CSV.

Por ejemplo, dada la llamada al método tableroToMatriz:

```
int m[][] miMatriz=tableroToMatriz("miTablero.txt");
```

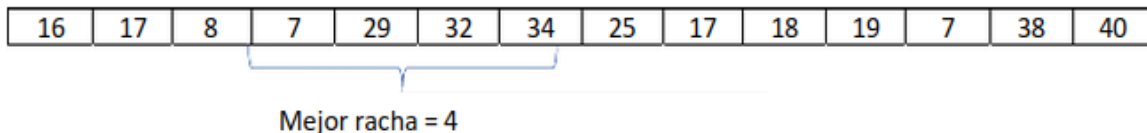
y con el fichero miTablero.txt siguiente:



Ejercicio 3: (2 puntos): Implementar el método insertarPuntuación que calcula la puntuación más alta de las partidas de un jugador y la inserta en la tabla HighScores del fichero HighScore.sql. Si hubiera más de una partida con la misma puntuación, se tomará la última partida jugada. Una vez insertado, este método, además calculará el ranking del jugador (mediante el método setRanking).

Ejercicio 4: (2 puntos): Implementa el método mejorRacha. Este método recorrerá la lista de partidas de un jugador de principio a fin y, sin utilizar estructuras de datos auxiliares (sólo variables con tipo de dato nativo) devolverá el máximo número de partidas que, de forma consecutiva, ha mejorado su puntuación.

Por ejemplo, dadas las siguientes puntuaciones:



Ejercicio 5: (2 puntos): Implementa el método mediaPais que retorne la media de las puntuaciones de los jugadores de un país y que tengan entre 18 y 25 años. El método toma como parámetros:

- El país del cual se quiere calcular la media de puntuaciones.
- Un array de jugadores.