

Pontificia Universidad Javeriana
Departamento de Ingeniería de Sistemas
Curso: Análisis de Algoritmos
Profesor: Danilo Castro Téllez, M.Sc.

Proyecto del curso
Forma de trabajo: Individual o parejas

1. El problema

SameGame es un juego para una persona que utiliza una cuadrícula de $m \times n$, con m y n entre 2 y 50, en donde cada cuadro está pintado de un color escogido de una lista de $k > 4$ colores. En cada turno, el jugador escoge un grupo de dos o más cuadros vecinos del mismo color (se consideran vecinos dos cuadros si comparten una arista horizontal o vertical) y todos los cuadros pertenecientes a este grupo son eliminados del tablero. Los cuadros que queden sin soporte vertical caen, como en Tetris. Si una columna se queda sin cuadros, se elimina corriendo todo el juego a la izquierda para volver a formar un tablero compacto. Esta acción ocurre incluso cuando la columna en quedar vacía es la de más a la izquierda. Se considera que la fila inferior y la columna de más a la izquierda tienen índice 0.

El objetivo del jugador es eliminar todos los cuadros del tablero y obtener de esta forma $(mn)^2$ puntos. Si no se eliminan todos los cuadros del tablero, cada jugada que elimine x cuadros tiene como puntaje el valor $(x - 2)^2$, y el puntaje del juego es la suma de los puntajes de las jugadas.

2. El reto

Usted y su equipo deben construir un jugador de SameGame que sea capaz de, dado un tablero de $m \times n$ coloreado con k colores diferentes, retornar una secuencia de jugadas que resulten en el máximo puntaje posible a partir de la configuración dada. Si una jugada de una solución resulta ser ilegal porque escogió una casilla vacía, indeterminada o sencilla, el puntaje total de todo el juego es de 0.

El jugador diseñado por ustedes no debe utilizar ningún medio de persistencia, es decir, tiene prohibido leer o escribir cualquier elemento externo, ni debe depender de ninguna librería externa aparte de la biblioteca Java estándar. La solución debe construirse utilizando Java 8 y el código debe estar escrito en español.

Para llegar a este resultado deben realizar las siguientes entregas:

Formalización del jugador: Definición de tipos de dato, signatures y especificación de métodos.

Diseño detallado del jugador: Diseño de los métodos en pseudocódigo utilizando la formalización indicada por el profesor, análisis de complejidad temporal y espacial en peor caso para todos ellos.

Implementación de la solución: Implementación de la solución en lenguaje Java 8 y utilizando la codificación de caracteres UTF-8.

3. El torneo

3.1. Formato

Las soluciones participarán en un torneo para determinar la mejor. Dicho torneo tendrá lugar en la segunda semana de exámenes finales. Para determinar el ganador, cada solución se enfrenta a las demás de la siguiente manera:

- En una primera ronda todas las soluciones son sometidas a un conjunto de 5 tableros diferentes, y sus puntajes obtenidos son registrados. Las tres soluciones con mayores puntajes “pasan el corte” y clasifican a las rondas finales. Los demás equipos ocupan los lugares del 4 en adelante.
- En una segunda ronda las soluciones clasificadas 2 y 3 se enfrentan a 5 tableros adicionales para determinar el segundo finalista. El equipo perdedor se ubica en la posición 3.
- La gran final enfrenta al ganador de la ronda anterior con el primer clasificado en la primera fase, a 5 tableros adicionales.

3.2. El robot

La manera en la cual las soluciones se ejecutan es por intermedio de un robot que las carga y llama los métodos correspondientes siguiendo las reglas del juego. El robot decide sobre la legalidad de las jugadas y su contabilización de puntaje es inapelable.

El robot puede descalificar a un jugador, entre otros, en los siguientes escenarios:

- Cuando una operación de un jugador afecte un cuadro vacío.
- Cuando una operación de un jugador afecte un cuadro inexistente o indeterminado.
- Cuando una operación de un jugador afecte un cuadro que no sea parte de un grupo de al menos 2 casillas contiguas del mismo color.
- Cuando una operación de un jugador arroje una excepción inesperada.
- Cuando una operación de un jugador tome más de 2 segundos.