Haman Meryeme
Pathammavong Emilie

Khani Hossein

Projet de java : Infinity loop

Programmation java avancée

2021/2022

Solve

1. Methods in Generator

- public static Grid **generateLevel**(String fileName, Grid inputGrid) :

Takes an empty file and grid, and appends pieces in the cell depending on the possibilities given by it and its neighbors. It therefore creates an already solved level, and then shuffles the position of the pieces in each cell then puts the solvable grid in a file. To do that, we calculate the number of connectors of each piece and according to this number we associate a type among the five possible ones (VOID, ONECONN,LTYPE,...).

- public static **gridToFile**(String name,Grid grid) :

Takes a grid and writes it in a file.

2. Methods in checker

- public static boolean **check**(Grid grid) :

. Initializes a double dimension array that keeps track of checked pieces, i.e the piece matches all its neighbors.
. Checks if the grid is solved or not. To do that, recursively checks if a piece and all of its neighbors matches all of their neighbors, going until given depth.

- public static Grid **buildGrid**(String inputFile) throws FileNotFoundException :

Builds a grid based on a file, to be used in isSolution.

3. Methods in Solver

- public Grid **solve**(Grid grid, int i, int j) :

A recursive method. If the grid isn't resolved, tests for the given piece and grid all its possible orientations, and also does so for next pieces recursively. If the grid is resolved, the program stops here and returns the solved grid.

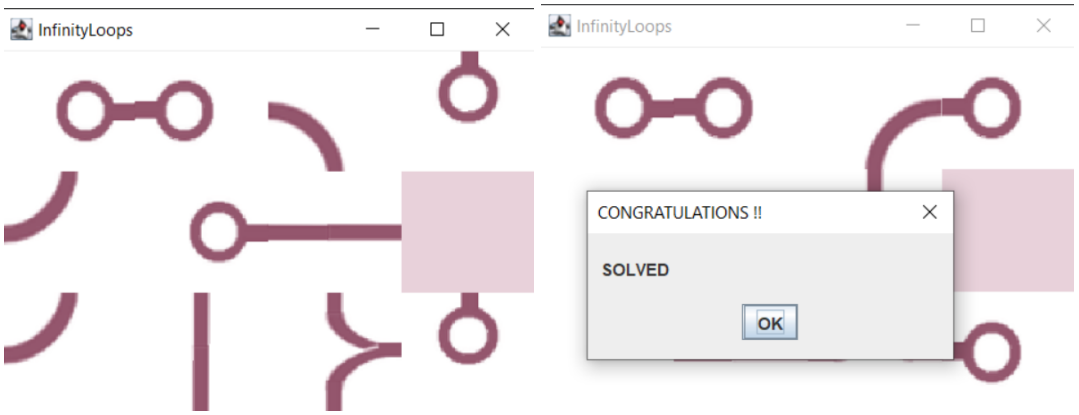
- public Grid **solveGrid**(Grid grid) :

Solves the grid by calling the **solve**(Grid grid, 0, 0) (0,0 to start from the first line and the first column)

GUI

4. GUI

The GUI class represents the graphical interface that allows to play, generate, check and solve a grid.



Test

CheckerTest(), **GeneratorTest**(), **OrientationTest**(), **PieceTest**() : these classes contain the tests of some methods that can be tested

Execution

To run the program, please use the jar in the project folder **(JavaInfinityLoop.jar)**, and generate the grids in the grids folder

Conclusion

If we had more time, we could have improved our solver by implementing a method using threads, as well as optimizing it by not doing it exhaustively but choosing the starting piece etc.

Note :

All defined methods contain the javadoc, please check it to understand our methods in detail.

If you encounter any difficulties in running the program (which is unlikely) please contact us by mail.