Project – Task 1

Investigate the path-blocker game:

https://www.kongregate.com/games/Zabobinozaur/path-blocker

Implement the game mechanics and search algorithm that solves the game for the 10 levels (14x14 grid world each). Keep levels in level01.txt, level02.txt, ..., level10.txt. Choose suitable search algorithm: recursion (with or without backtracking), DFS (tree or graph version), BFS (tree or graph version), depth-limited DFS, Iterative Deepening, or Uniform Cost Search.

In the Java class with the main method add comments answering below question:

1) Why you prefer the search algorithm you choose?
2) Can you achieve the optimal result? Why? Why not?
3) How you achieved efficiency for keeping the states?
4) If you prefer to use DFS (tree version) then do you need to avoid cycles?
5) What will be the path-cost for this problem?

There will be no GUI. Your code will simply output the search result into 10 directories for each level as png files (create the directories if they do not exist). For example:

```
level01\0001.png
level01\0002.png
level01\0003.png
.
.
.
level01\0026.png
level02\0001.png
level02\0002.png
.
.
.
```

If you did not submit a working efficient code, then your project grade will be decreased by 25 points. Submit the Java classes (.java) and levels (as txt files) packed in a pathblocker.zip file. Your zip file will also contain groupMembers.txt with group members as: name surname student-id department year (each student in a separate line). So, there will be only .java and .txt files in your zip file (only one student in the group will submit it).