Essential Computing Fall 2023

Mini-Project: Random Pits

Roskilde University J. Classen, Bo Holst-Christensen

Description

The purpose of this project is to implement and test a Java program that implements a single-player game where the player has to reach a goal location while avoiding randomly placed pits.

Rules

1. The game is played on a board (grid) consisting of 20×20 cells.
2. Initially, all cells are free, and the player is located in cell $(0, 0)$.
3. In each turn, the player can move in one of the four cardinal directions (north, east, south, west).
4. In each turn, a random number (between 1 and 5) of pits is placed on random spaces of the board that don't already have a pit.
5. The game is won if the player reaches the goal cell $(19, 19)$.
6. The game is over and the player loses, if they stand on a pit.
7. The game is also lost if the player get surrounded by pits or is not able to reach the goal location anymore.

Examples and Hints

Use specific characters to mark specific objects, such as:

- `X` the player
- `.` free cell
- `o` a pit

At the beginning, the user will be shown something like this:

```
X . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
Which direction (n,e,s,w)?
```

If they type "e" in order to go east, the resulting situation could look like this:

```
. X o . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . o . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . o .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
Which direction (n,e,s,w)?
```

Here, it was randomly chosen to place three pits, and to place them at locations $(2, 0)$,

$(15, 1)$, and $(18, 10)$. If the player now types "e" again, they lose due to stepping into a pit:

```
. . Y . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . o . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . o . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . o .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . o . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . .
GAME OVER: You fell into a pit!
```

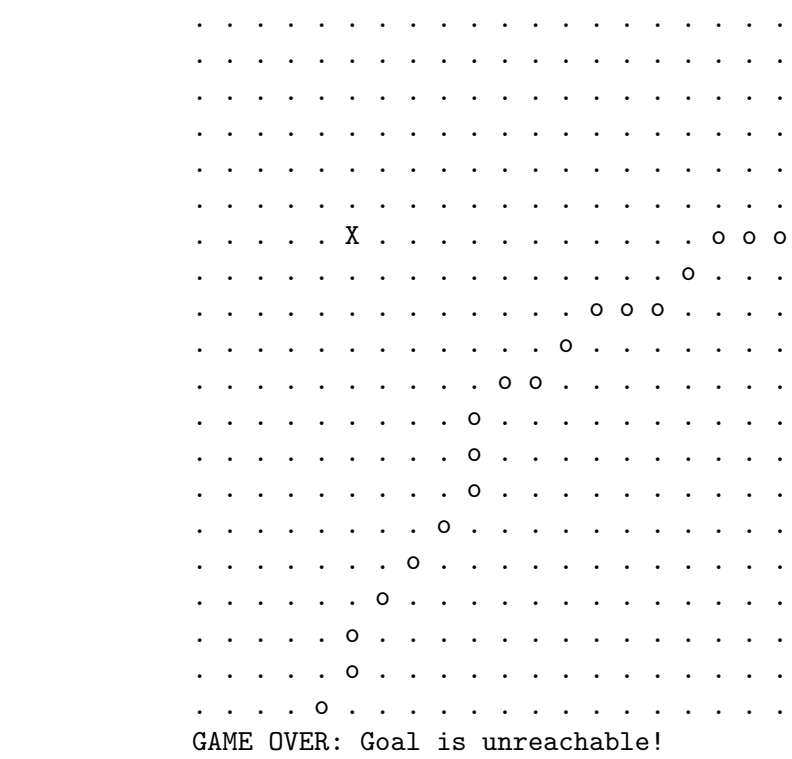
The most challenging part is probably to detect whether the game ended due to the goal having become unreachable. A simple case is where the player is immediately surrounded by pits, as shown below:

```
. . . . . .
. . . . . .
. . . o . .
. . o X o .
. . . o . .
. . . . . .
. . . . . .
```

However, it could also be that none of the direct neighbouring cells of the player are occupied by pits, yet the player is still "trapped" inside a larger circle of pits:

```
. . . . . .
. . . o o .
. . o . . o .
. o . X . o .
. . o . . o .
. . . o o . .
. . . . . .
```

In fact, it might be that the player is simply "cut off" from the goal due to pits being placed in a way that no path to the goal exists anymore:



In each of these case, the game should end immediately, and display the message that the game has been lost.

Evaluation Criteria

Your program will be evaluated on the basis of the following criteria:

- **Functionality.** The application should work as expected.
- **Reflection of the course.** You are expected to use elements from the course in your solution.
- **Well organized and commented code.** A concise architecture with clear, logical break-down of the code into methods and classes, that make it understandable and easy to read.
- **Error Handling.** Try to make sure that the application is solid and can handle incorrect input and or other possible errors.
- **Project Description.** Add a short PDF document (max 2 pages) that describes the reasoning behind your solution.

About the Exam

The exam will be a 15 minute individual oral exam. The exam will mainly be about the assignments (especially the mini project) but questions can be related to the whole course curriculum.

Plagiarism

Notice that it is OK to find inspiration on the internet, but DO NOT copy&paste! If you choose to use parts of other people's code, make sure you a) give them credit for it and b) understand their code fully as you will be examined on the entirety of the code as if had you written it yourself.

If you find code somewhere and want to use it as a part of your solution, you should add a comment and write exactly which parts of the code you have copied, and where it has been copied from (give the URL). Take ownership of the code - you must be able to explain any part of the source code.

Working Together in Groups

You can work alone, in pairs, or groups of up to 3 persons. Clearly state the names of all group members on each page of the documentation.

Deadline

Your solution (.java files in a zip archive) and description (as a PDF document) must be uploaded to eksamen.ruc.dk no later than Wednesday, 22 November 2023 at 10:00.