RobotPathFinder

Arrays Assignment for ICS3U1
Michael Pu - December 2017
Mr. Radulovic - Don Mills Collegiate Institute
Work log can be found in the commit history of this GitHub repository

Note: If you are reading a PDF version of this README file, some of the links may not work properly. The same README file with working links can be found in the GitHub repository above.

Description

- Reads and parses a map file into 2D array format
- Computes the shortest path from the starting position to the goal
- Displays the map and animates the shortest path

Algorithm

- Theoretically, it is able to finds the shortest path from the start to the goal in any map, when resource constraints are not taken into account
- Testing of the algorithm was done in two parts
    i. The automated testing after every minor change was done using a JUnit test, which tests the basic functionality of the algorithm.
    ii. The manual testing after major changes was done manually using the test files found in src/tests, which tests the algorithm's ability to find the shortest path in maps of various sizes and structure. Variations of these maps (more dead ends, different arrangement of walls, different positions for start and goal position, etc.) were also used in testing which are not included.
- Using these tests, the algorithm is **confirmed** to find a path which *appears* to be the shortest without any backtracking. For some of these test cases, especially for the ones with map dimensions of greater than 100 cells, it would be infeasible to manually verify that it was the shortest path.

Project Organization

Package	Description
algorithm	files related to calculating the shortest path in the maze
backend	files related to the processing and calculations that is not part of the algorithm
frontend	files related to the JavaFX components displayed on screen
tests	test cases files and JUnit test files for the algorithm

Features

- Application icon
- A file select dialog to select the map
- Validates the map to ensure that it is of the proper dimensions, there are no invalid characters, and there is a possible path
- Buttons at the top of the animation window to easily restart the animation or select another map

- Window resizes automatically according to the size of the screen (90% of the height or width of the screen, depending on which is smaller)
- The size of each cell changes automatically depending on the dimensions of the map and the size of the window
- Keeps track of the number of moves made by the robot

Map Legend

Cell Contents	Meaning
0	Obstacle
1	Empty Space
R	Starting Position
G	Goal Position

Each character in the map should be separated with a whitespace character with a new line separating each row.