Lab 7

2 March 2017

In this lab, you'll add to your Maze class a method doesPathExist that uses a stack to determine whether there's a path from start to finish. Before you come to lab, you should read the descriptions below and the code I've provided in /home/shared/162-1/lab7/, which includes *my* solution to Lab 3.

On paper (in your notebook is fine), take notes on any parts of my code that you find confusing or surprising, or that you just aren't sure how they work. Also, write out the additional maze examples as described below.

Back to the maze

I know some of you didn't fully complete Labs 2 and 3, and I don't want that to hold you back on this lab, so I did put an entire solution in the shared directory that you can use as your starting point.

However, many of you had a complete or very-nearly-complete version of those labs, which I encourage you to build on in this lab. If so, copy your Lab 3 work into this week's directory and add methods west, north, and south to Location (following the pattern of east); and look through my solution just to see if there was anything else you missed. (Do look at test_Maze.u to see how print was tested; if you had a mostly working Maze but didn't test it, feel free to copy over just the test file from my version.)

Test cases

Before we get into the maze solving, let's think about examples and test cases. Before, we were mostly concerned with making examples that would test reading and printing—and less concerned with the maze-ness of the examples. Now, though, we want examples that will be able to effectively test a method that asks, "is there a path from the start to the finish?".

So, let's add another few test cases. In your notebook, draw out a few very short and simple mazes—some of which have solutions and some of which don't. Start with two that are super-simple: one where the start is surrounded on four sides by walls, and one where the start and finish are adjacent and surrounded by walls. Then, add more that are still relatively small but with added complexity: some with and some without a solution, some with actual branching, and at least one where the border of the grid is not all walls, with the correct path to the finish skirting the edge of the grid.