Project 1 – Path-based Location Search HINTS

Hints & Tips

1.) Don’t worry about the autograder; your code won’t pass it so we will ignore it. Instead just focus on getting the test cases described working (typically, a tinyMaze, mediumMaze and bigMaze).

2.) Make sure you use the util.Stack(), util.Queue() and util.PriorityQueue() versions of those data structures, as they are required for the pacman game’s full functionality (just some behind the scenes stuff that you don’t need to worry about). You may need to have a line of code that says “import util”

3.) For **numbers 1-4**, you should only be writing code in the depthFirstSearch(), breadthFirstSearch(), uniformCostSearch(), and aStarSearch() functions, respectively.

4.) Focus on getting **number 1** coded up correctly. Once you get number 1 right (DFS), number two should be a trivial change of data structures. In fact, DFS and BFS can both call the same function with a different parameter.

5.) Once done with problems 1 and 2, focus on getting **problem 3** right. You can probably start with your code for problems 1 and 2, and add a few small changes to get problem 3 working properly.

6.) **Number 4** (a*) will again be a slight modification of number 3 (UCS). As was the case with problems 1 and 2, you may be able to call the same function for UCS/A* with a different parameter (that’s how similar they are).