CSSE 413: Planning and Search Programming Assignment

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Please answer the following questions and submit a pdf copy of this document with your code.

1. Map 0. This map is designed primarily for testing the basic functionality of your code. Nevertheless, your bfs search algorithm displays some interesting behavior. How many states are placed on open? \_\_\_\_\_ Below, explain why bfs places this number of states on open.
2. Map 1. Again, this map is primarily designed for basic testing. It too displays some interesting behavior though. Did your bfs search algorithm find a solution? \_\_\_\_\_ How many states did bfs place on open? \_\_\_\_ Below, explain why bfs places this number of states on open.
3. Map 2. What is the path length of the solution found by bfs? \_\_\_\_ How many states dis bfs place on open? \_\_\_\_ Below, explain why bfs places this number of states on open.
4. Map 3. What is the path length of the solution found by bfs? \_\_\_\_ How many states dis bfs place on open? \_\_\_\_ Below, explain why bfs places this number of states on open.
5. Map 4. Did you robot find a solution? \_\_\_\_
6. Map 5. What is the path length of the solution found by bfs? \_\_\_\_ Does your robot hug the border of the domain? \_\_\_\_ Below, explain why your robot hugs or does not hug the border of the domain.
7. Map 10. This map is designed for basic testing of reaching multiple targets. You will likely have to change the way you represent a state in this domain. What is the path length of the solution found by bfs? \_\_\_\_ How many states did bfs place on open? \_\_\_\_\_
8. Map 11. This map is designed for additional testing of reaching multiple targets. What is the path length of the solution found by bfs? \_\_\_\_ How many states did bfs place on open? \_\_\_\_\_
9. Map 12. This map is designed for in-depth testing of reaching multiple targets. What is the path length of the solution found by bfs? \_\_\_\_ How many states did bfs place on open? \_\_\_\_\_
10. Map 14. In this map, we begin to explore the limits of bfs. What is the path length of the solution found by bfs? \_\_\_\_ How many states did bfs place on open? \_\_\_\_\_
11. Map 15. What is the path length of the solution found by bfs? \_\_\_\_ How many states did bfs place on open? \_\_\_\_\_

We will now look at A\* and compare and contrast it to bfs.

1. What is the behavior of A\* on the following maps:
2. Map 0: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
3. Map 1: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
4. Map 3: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
5. Map 4: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
6. Compare the values from above to those obtained by bfs on the same maps. Are they different? Below, explain any patterns you see.
7. What is the behavior of A\* on the following maps:
8. Map 2: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
9. Map 5: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
10. Compare the values from above to those obtained by bfs on the same maps. Are they different? Below, explain any differences you see.

Four corners. For maps 10-12, you need to develop a different way of representing the state and you need to develop a more complex heuristic. Among others, you need to know which corners you already visited. For example, if the robot is in position <1,2> then there is a difference if the robot visited the upper left or the lower right corner. In other words, you need to keep track of which corners you already visited (or alternatively, still need to visit.) There are few other things you need to keep track, just don’t good overboard. We will grade problem (14) based on the quality of your state representation and the performance of your heuristic function. We will use astar101112() to run the experiments. They key to this problem is to develop a powerful heuristic that does NOT overestimate. For map 12, can you find a solution by placing fewer than 800 nodes on open?

1. Below, list the behavior of your A\* algorithm on the following maps:
2. Map 10: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
3. Map 11: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
4. Map 12: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_

Many targets. Suppose we have many targets. Develop a heuristic to solve maps 14, 15 and 16. We will use your astar141516() on those two maps. Again, ensure that your heuristic underestimates. Just as before, the fewer states you expand, the more points you will get. The challenge problem is to see whether your heuristic can solve problem 15 by expanding fewer than 7000 nodes.

1. Below, list the behavior of your A\* algorithm on the following maps:
2. Map 14: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
3. Map 15: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_
4. Map 16: Solution length: \_\_\_\_ Number of states placed on open? \_\_\_\_\_\_