Name: Berke Derin Berktay
Comp 341 homework 1 report
Question 1: BFS would usually take less time, especially in smaller problems such as the small and medium mazes since it explores every possible neighbor node one after the other whereas the DFS chooses one path to go to as deep as possible and continues like that iteratively which has a lower probability of guessing the right branch whereas the BFS is the safer bet and therefore If the searched node is quite shallow we would prefer BFS but if the search node is quite deep it is better to use the DFS. In terms of the expanded nodes, the DFS will go deeper and deeper whereas the breadth search will go level by level.

Question 2: UCS will check the lower cumulative cost nodes first so in this case, its exactly the same as BFS. A* is better for this case since it will always carry on to the node which has the closest Manhattan distance to the goal state. In many ways they are quite similar since they both consider the cumulative cost, however the way that they calculate the cost is quite different from one another, the uniform cost method is considered to be a blind method since it does not consider the proximity whereas the a star does this with the heuristics, such as the manhattan distance in the example problem. Keep in mind that the a star is going to try to minimize the number of expanded nodes. If you just want to blindly search and not consider the distance from the target use UCS whereas if you do want to consider it use A*. The a* expands to less nodes whereas the UCS expands to more. As this is seen in the maze examples when the code is run.

Question 3: Basically first I have initialized the problem by checking if whether or not the start state is at one of the corners with the four elif statements. Then we basically have an is goalstate that uses the same logic where it checks for all four corners again and if and only if all four corners are visited we return true. Then for getsuccessor method we basically determine the target x and target y coordinates by adding the current state and the action that needs to be taken in order to get to the target coordinate. Then we check whether or not the new target coordinates are wallsa and if not we are good and can check if the=is new target location is one of the target corner locations. Then we add it to the list of successors with the current fulfilled corners , the action route and the cost, which is always one as requested.

Question 6: An admissable heuristic does not overestimate the true cost to a nearest goal whereas a consistent heuristic’s iterative cost never overestimates the actual iterative cost when going from neighbor to neighbor. This is the main difference.A* is a good example of an admissible heuristic. Keep in mind that all consistent heuristics are admissible heuristics but all admissible heuristics aren’t consistent heuristics.This statement derives follows the definition above.