**Mid-Term Extra Credit**

**ITS 265**

**Ghadir Alfadhl**

1. (10 pts extra credit for mid-term) Use the Lab 3 Python program, Problem 2 where you had used a depth-first search algorithm to plan a path to a goal in a maze (maze-in2 on Brightspace) and modify it as follows:

1. Create a separate Python program that uses breath-first search (BFS) to plan a path to a goal. Write or adapt code for a BFS goal plan (BFS code not included in assignment).
2. Create a separate Python program that uses A\* to plan a path to a goal. Write or adapt code for an A\*algorithm.
3. Compare the performance of your original depth-first search, and the new breath-first search, and A\* algorithms in planning and executing a goal. Use six different goal positions throughout the maze (some close to the starting position and others deep in the maze) to see how the algorithms perform. Create a table that shows the results and write a narrative explaining the results.
4. Submit the BFS and A\* code along with your analysis to Brightspace.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Compare | BFS | DFS | A\* | Goal to start |
| Frist goal | 0.0537264347076416 | 0.030318021774291992 | 0.027638912200927734 | furthest |
| Second goal | 0.2055361270904541 | 0.17069435119628906 | 0.17812514305114746 |  |
| Third goal | 0.048020124435424805 | 0.016988277435302734 | 0.017045021057128906 |  |
| Fourth goal | 0.18331098556518555 | 0.06385636329650879 | 0.07242131233215332 |  |
| Fifth goal | 0.0019996166229248047 | 0.20944547653198242 | 0.2175438404083252 |  |
| Sixth goal | 0.0 | 0.19751930236816406 | 0.20409131050109863 | Nearest |

Breadth-First Search is a tree or graph traversing (searching) algorithm. The slowest one compared to DFS and A\*. However, I noticed that BFS was the fastest when the goal was near to the start. But in another search results, A\* was the fastest compared to BFS and DFS