**AI LAB 3**

1. Fill out the table below:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Best First Search** | | | **A\* Search** | | |
| **Maze** | **#nodes explored** | **Solution length** | **Is it optimal?** | **#nodes explored** | **Solution length** | **Is it optimal?** |
| **tiny** | 10 | 10 | No | 15 | 8 | Yes |
| **medium** | 165 | 152 | No | 268 | 68 | Yes |
| **big** | 435 | 210 | Yes | 619 | 210 | Yes |

1. What happens on openMaze for the various search strategies?

**Best First Search:** This search will scan all squares of the left box in the maze and then find the left most way to the food

A screenshot of a video game

Description automatically generated

**A\* Search:** Meanwhile, this algorithm will scan all the map and choose the least cost way to the food.

A screenshot of a video game

Description automatically generated

1. For each exercise where a heuristic is used, clearly show/mention the heuristic function.

**BEFS** uses only the heuristic function to prioritize which node to explore next. A heuristic function that estimates the cost to reach the goal from any given state should be provided. This algorithm uses this heuristic to guide the search.

**A\* Search** uses a total sum of the actual cost g(n) and the heuristic h(n). The heuristic helps A\* to estimate the total cost f(n) = g(n) + h(n).

1. Based on the above, a short discussion/reflection of how the searches compare to each other and to the uninformed searches from Assignment#2.

**A\* Search** is more optimal than **BEFS.** Although BEFS does care about the cost of the path, it only considers the first lowest cost path it saw, no matter how much cost of the rest of the path so it may not choose the best way to the food. Meanwhile, A\* Search can carry both considerations about the shortest path and the lowest cost to choose the most effective way to the food.

* Optimization: **A\*** > **UCS** > **BFS** > **BEFS** > **DFS**
* Completeness: **A\*** > **UCS** > **BFS** > **BEFS** > **DFS**
* Time complexity: **DFS** (most efficient in practical use) > **BEFS** (depends on heuristic) > **A\*** > **UCS** > **BFS**
* Space complexity: **DFS** > **BEFS** > **A\*** > **UCS** > **BFS**
* Heuristic dependence: **A\*** (best if heuristic is good) > **BEFS** > **UCS** > **BFS/DFS** (no heuristic)