Assignment 03

Hamza Farooq   
Reg# 2020MSCS513

# Challenge 01

Depth-first search (DFS) implemented in search.py.

|  |  |  |  |
| --- | --- | --- | --- |
| **Maze Type** | **Nodes Explored** | **Cost** | **Result Screenshot** |
| TinyMaze | 14 | 10 |  |
| MediumMaze | 144 | 130 |  |
| BigMaze | 390 | 210 |  |

# Challenge 02

|  |  |  |  |
| --- | --- | --- | --- |
| **Maze Type** | **Nodes Explored** | **Cost** | **Result Screenshot** |
| MediumMaze | 144 | 130 |  |
| MediumMaze Reversed | 264 | 244 |  |

The nodes pushed first into the stack will be dealt last. So, in reverse order, the nodes pushed into the stack last will be dealt first i.e., consider the left subtree from the root node contain the appropriate path. If It will only be explored earlier when it is pushed last.

# Challenge 03

In terms of expanded nodes, depth-first search is better.

|  |  |  |
| --- | --- | --- |
| **Maze Type** | **Search Type** | **Result** |
| TinyMaze | Depth-first search |  |
| Breath-first search |  |
| MediumMaze | Depth-first search |  |
|  | Breath-first search |  |
| BigMaze | Depth-first search |  |
|  | Breath-first search |  |

# Challenge 04

Uniform-cost search is implemented in search.py and results are populated as follow.

|  |  |
| --- | --- |
| **Maze Type** | **Result** |
| MediumDottedMaze |  |
| TinyMaze |  |
| MediumMaze |  |
| BigMaze |  |

# Challenge 05

A\* search is implemented in search.py and results are populated as follow.

|  |  |
| --- | --- |
| **Maze Type** | **Result** |
| TinyMaze |  |
| MediumMaze |  |
| BigMaze |  |
| MediumDottedMaze |  |

# Challenge 06

1. Fill following table with the information.
   1. **Tiny Maze**

|  |  |  |
| --- | --- | --- |
|  | **A\* Heuristic** | **UCS** |
| **Total Cost** | **8** | **1** |
| **Nodes Expanded** | **8** | **14** |
| **Score** | **502** | **502** |

* 1. **Medium Maze**

|  |  |  |
| --- | --- | --- |
|  | **A\* Heuristic** | **UCS** |
| **Total Cost** | **76** | **1** |
| **Nodes Expanded** | **81** | **287** |
| **Score** | **434** | **436** |

* 1. **Medium Dotted Maze**

|  |  |  |
| --- | --- | --- |
|  | **A\* Heuristic** | **UCS** |
| **Total Cost** | **76** | **1** |
| **Nodes Expanded** | **81** | **210** |
| **Score** | **644** | **646** |

* 1. **Big Maze**

|  |  |  |
| --- | --- | --- |
|  | **A\* Heuristic** | **UCS** |
| **Total Cost** | **210** | **5** |
| **Nodes Expanded** | **466** | **636** |
| **Score** | **300** | **300** |

1. **Why Node Expanded is Greater in UCS and Less in A\* Heuristic?**

The uniform cost search is uninformed search. It chooses the node to explore which is simply lower in cost with respect to current state. Meanwhile, A\* search have the information in form of heuristic that helps the agent to find the best path optimally because it knows what path will lead it to the goal in less cost so it expands the relevant nodes only.

1. **Compare other parameters and give reason why they greater/less/equal.**

The cost is minimum in ucs as compared to A\* because ucs will always find a search path that have minimum cumulative cost in weighted graph.

The score is nearly equivalent because both ucs and A\* has their benefits and drawbacks. Ucs gives minimum cost but expands more nodes than A\*. The score is calculated on the base of both the nodes expanded and total cost of the chosen path.

# Challenge 07

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Euclidean** | **Manhattan** | **UCS** |
| **Total Cost** | **152** | **76** | **1** |
| **Nodes Expanded** | **159** | **81** | **287** |
| **Score** | **358** | **434** | **436** |

The Euclidean distance is just displacement from the current state to the goal state and it is always a straight line irrespective to the grid so indeed if the Euclidean distance is minimum, the travelled distance may not be minimum. Meanwhile, the Manhattan distance is not just the displacement, it is considering the grid and changes if the grid changes.