

Homework 2

Maze Pathfinder Exercise

You are given a maze as a 10 x 10 grid of ones and zeros. The zeros are open spaces in the maze and the ones are impenetrable walls. The maze is mapped to 2D Cartesian coordinates.

The Maze

```
1 1 1 1 1 1 1 0 1 1
1 0 1 1 0 1 1 0 1 1
1 0 0 0 0 0 1 0 0 1
1 1 1 1 1 0 1 1 0 1
1 0 1 0 1 0 0 0 0 1
1 1 1 0 1 1 0 1 1 1
1 0 0 0 0 0 0 0 1 1
1 0 1 1 0 1 1 0 1 1
1 0 0 0 1 0 0 0 1 1
1 0 1 1 1 1 1 1 1 1
```

There is a single entrance at the bottom of the maze and a single exit at the top. Diagonal moves are prohibited; the only allowable move directions are up, down, left, right. The maze is given in a text file in the assignment folder on BlackBoard.

Find a solution path for the maze using the following algorithms.

1. Depth-first search (30 points)
2. Breadth-first search (30 points)
3. A* search (40 points)

Requirements:

- i. Python implementation.
- ii. The implementation must read the provided maze file from *stdin*.
- iii. The directional priority must be: up, right, left, down.
- iv. Use the L2 (Euclidean) norm for the distance heuristic.
- v. Mark all visited cells with the number 2 (overwrite the zeros).
- vi. Final solution path must overwrite the 0/2s with the number 5 (without backtracking).
- vii. For each search algorithm output to *stdout* the maze with overwritten cells indicating visited cells and cells making up the solution path.
- viii. Provide usage on how to run each algorithm from command-line: e.g., `python search.py -h`
- ix. Command-line option to run each algorithm: e.g., `"-b"`, `"-d"`, `"-a"`
- x. Provide your Python implementation as a standalone, single text file so it can be demonstrated on a standard Linux system.