EC558
Assignment # 1

Solving a Maze Using Search Algorithms

Due Date: Thursday 24/4/2025

Introduction

The purpose of this project is to investigate different uninformed and informed search algorithms, and to design search heuristics. This will be accomplished by implementing an agent that navigates through a 2D maze. This assignment can be done in teams of no more than two students.

Problem statement

Assume a 25 × 25 two-dimensional maze. Each square in the maze has an (x,y) coordinate, with the bottom-left corner being (0,0) and the top-right corner being (24,24). Each position in the maze can be either empty or blocked. In addition, there are two "special" positions, the starting position and the exit position, both of which are selected randomly.
The agent can only move up, down, left or right, but never diagonally. It also cannot enter blocked positions or move outside the maze. Its objective is to find a path from its starting position to the exit position, preferably the cheapest one. The cost of a path is the number of positions the agent has to move through, including the starting and exit position.
An example of a maze is given in Figure 1.

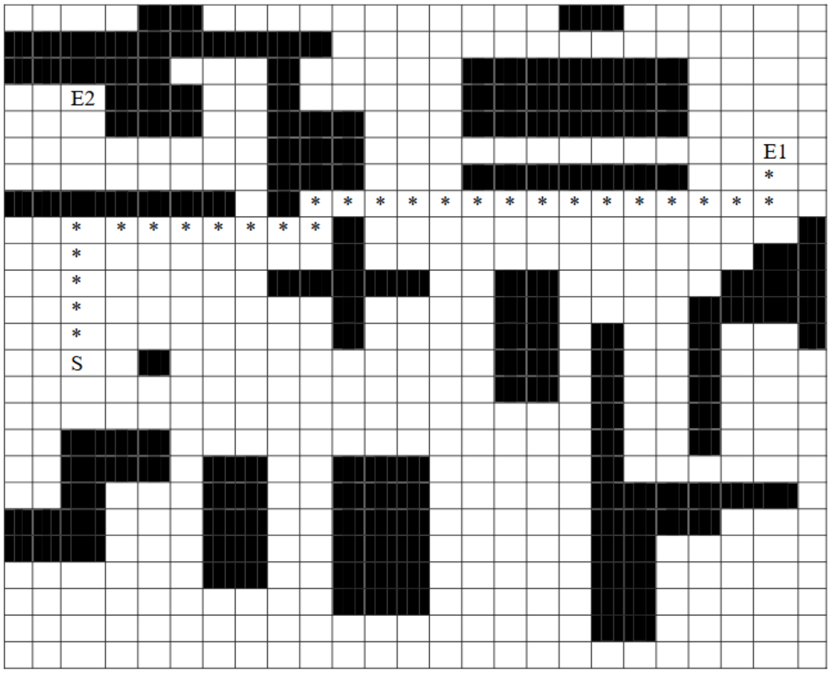

Figure 1: Example maze. The start position is marked S, the exit positions are marked E1 and E2, empty positions are blank and blocked positions are black. The path, marked by stars, has a cost of 30.

Requirements

For this assignment, you must implement code to find a path from the starting position to the exit position using

(1) Breath-First Search,
(2) Depth-First Search,
(3) A* Search.

For the A* Search, you must define an appropriate heuristic function, and justify your choice. **The expected Outputs** for each algorithm:

- The shortest path found (if one exists)
- The number of nodes expanded
- Execution time in milliseconds
- A visualization of the explored nodes (optional for extra clarity)

You can also implement and compare any other search method that is of interest to you. The above three search heuristics represent the minimum requirements for the assignment

Deliverables

You are expected to turn in a short report (the shorter the better) which should include the following:

1) A short description of your implementations of the search methods. In particular, for the A* Search, explain and justify your chosen heuristic function.

2) Sample output of each search method you implemented on the maze.

You will have to test each search technique two times:

1. With the agent starting at (0,0) and ending (24,24)
2- select randomly the start and end point

3) An analysis and comparison of the different search methods you implemented,
based on relevant criteria, along with appropriate conclusions.
You must also send the code you implemented in MS Teams. All this material must be handed
over prior to the due date.

Grade

Your grade in this project will be function of the quality and completeness of the report you hand in. The minimum requirements of the project are sufficient to have 100%, but extra work is worth bonus points (assuming of course it is well done and documented in the report).
Handing in the project after the deadline gives you automatically 0.

Marking Scheme:

1. Implementation of the search technique.

- Depth-first search
    - o Code (10 marks)
    - o Sample output (10 marks)
- Breadth-first search
    - o Code (10 marks)
    - o Sample output (10 marks)
- A* search
    - o heuristic function (10 marks)
    - o Code (10 marks)
    - o Sample output (10 marks)

2. The comparison should be carried based on these criteria:
    - **Memory requirement:** length of the queues used during the search (10 marks)
    - **Time requirement:** the number of times that any nodes (or positions) have been checked. (10 marks)
    - **Solution optimality**: the cost of the generated solution (10 marks)
3. The student can get up to bonus marks for every additional method that he/she would like to apply provided that it is included in the comparisons.