

2019/2020 COMP1037 Coursework 1 – Search Techniques

Introduction

Coursework 1 studies a Matlab program which is adapted from MathWorks. The question is based on your understanding of A* algorithm and extends your skills to develop well-known tree search techniques to solve the maze problem. The primary goal is to give you first-hand experience in implementing search techniques for solving the shortest route problem in Matlab, one of the most adapted high-level languages and analytic tools in business and research.

Marks

Coursework 1 accounts for 15% of the COMP1037 marks. Corresponding marks will be awarded for answering each of the sub-questions in FAIcw1.doc. Report should be written using this file as the template (keeping the same format of margin, font size and spacing, etc.).

Plagiarism vs. Group Discussions

As you should know, there is **no tolerance of plagiarism** and any breach of which will be dealt with according to the University's standard policies. Please be very careful not to cross the boundary into plagiarism while having general discussions regarding the coursework to promote the generation of new ideas and to enhance the learning experience. The important part is that when you sit down to actually do the work and write the answers, you do it individually. If you do this, and you truly understand what you have written, you will not be guilty of plagiarism. Do NOT, under any circumstances, share code or share figures, graphs or charts, etc.

Deadline and submission procedure

The submission deadline is 12pm on the 27th April 2020 via Moodle. Late submission results into a 5% reduction of your coursework mark for each weekday. Any work handed in after the 4th May will receive zero marks.

Name your submission file: **FAIcw1-XXX.zip** (incl. 1. your report '**FAIcw1-XXX.pdf**', 2. your **AStar-MazeSolver-XXX** code folder, 3. Your **BFSMazeGenerator-XXX** code folder), where XXX should be your student ID number, and submit a single compressed file via Moodle.

For question (b), your code needs to be runnable by calling the function with name '**AStarMazeSolver(maze)**', the input of function is a 'maze' generated by the maze generator. Any code cannot be successfully called by the above function name will be given zero marks for the code part.

If you can't submit your coursework on time due to Extenuating Circumstances, please contact your personal tutor first. I am only granting an extension of submission based on his / her recommendation.

Please remove the above text while writing your coursework report.

2019/2020 COMP1037 Coursework 1 – Search Techniques

This part is based on the maze generator demo (DFSMazeGeneration-master). The maze generator is a project written by some student using Matlab. He has adopted a tree search approach to randomly generate a maze with user-defined size and difficulty. **(15 marks)**

(a) Read the DFSMazeGeneration-master code, modify the code into a maze generator following **BFS** tree search strategy. [5 marks]

- i) The BFS maze generator needs to be successfully called by command **'main'**. (2 marks)
- ii) In the report, use **screenshot of code show what changes you have made. Explain why you make these changes.** (3 marks)

(b) Write a maze solver using A* algorithm. [10 marks]

- i) The solver needs to be successfully called by command **'AStarMazeSolver(maze)'** within the Matlab command window, with the assumption that the 'maze' has already been generated by the maze generator. The maze solver should be able to **solve any maze generated by the maze generator.** (2 marks)
- ii) In the report, show **what changes you have made** and **explain why you make these changes.** You can **use a screenshot** to demonstrate your code verification. (4 marks)
- iii) Your code need **display all the routes that A* has processed with RED color.** (2 marks)
- iv) Your maze solver should be about to display the **final solution with BLACK color.** (2 marks)

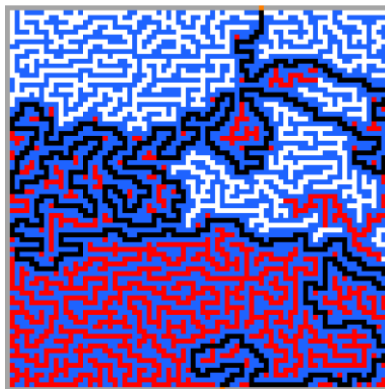


Figure 1. Sample output