

AU 332 Artificial Intelligence: Principles and Techniques

Assignment 1 Due Sep 23rd 11: 59pm

Adhere to the Code of Academic Integrity. You may discuss background issues and general strategies with others and seek help from course staff, but the implementations that you submit must be your own. In particular, you may discuss general ideas with others but you may not work out the detailed solutions with others. It is never OK for you to see or hear another student's code and it is never OK to copy code from published/Internet sources. If you feel that you cannot complete the assignment on your own, seek help from the course staff.

When submitting your assignment, follow the instructions summarized in Section 6 of this document.

1 Graph Traversal

In Figure 1, the start node is A. Please draw an expanding tree structure for the graph using depth-first search algorithm and the breadth-first search algorithm. If the graph has n nodes and the maximum degree for each node is d , what is the complexity of BFS and DFS?

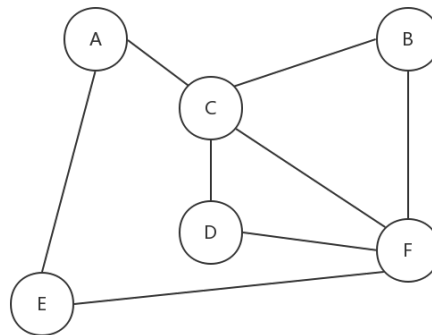


Figure 1: Solve DFS and BFS based on the given graph. Node A is the start node

2 Uniform Cost Search Algorithm

In Figure 2, the start node is A and the goal node is E, calculate the shortest path from node A to E using UCS method. Write step by step update of the fringe list and closed list.

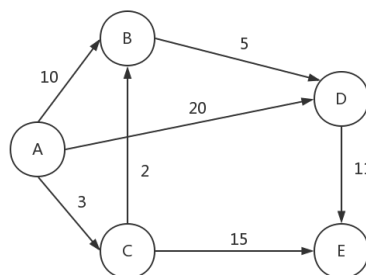


Figure 2: Uniform cost graph search problem

3 A* Algorithm

Write out the complete path finding process (fringe list and close list) from the green grid to the red grid using A* algorithm (blue grids are obstacles). The green grid is the start location and the red grid is the goal location. Write out your choice of h heuristic function. Use the row and column number as reference to the location of the cell. The action space are up, down, left and right. The cost for each action is 1.

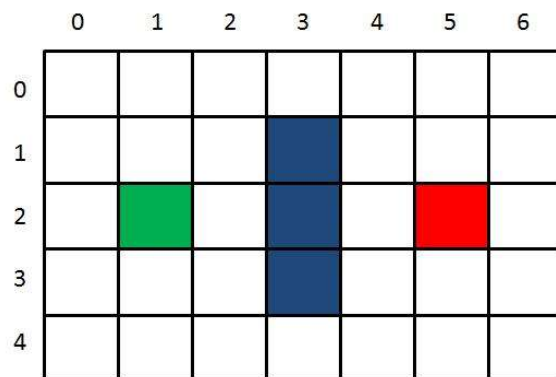


Figure 3: Use A* to solve for the maze

4 Programming Assignment

Please complete the partial code of *BFSvsDFS.py*, *UniformCostSearch.py* and *ASearch.py*.

5 Extra credit

For the programming assignment part, extra credits will be given if you completed the following cases.

1. Check if the goal and start state are valid nodes in the graph, return error handling message.
2. Check whether the graph satisfies the consistency of heuristics.

6 Submission instructions

1. Zip *BFSvsDFS.py*, *UniformCostSearch.py*, *ASearch.py* and a write up pdf to a folder called *HW1_ID_name.zip*
2. Send the zip file to TA 121103451@qq.com