Artificial Intelligence: Programming 1 (P1)
Search Algorithms

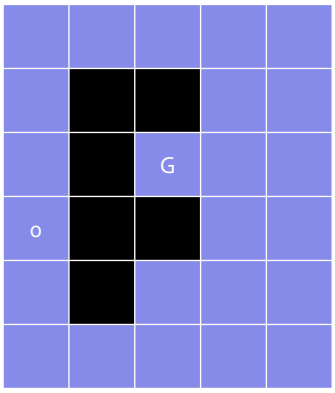
Instructor: Dr. Shengquan Wang

Due Time: 10PM, 9/16/2022

In this programming assignment, we aim to implement two search algorithms (Iterative deepening search and A* search) we have learned in class.

1 Instructions

We consider a maze under a windy condition as shown in the following figure. We assume that the



wind comes from the north and the cost of one step for the agent is defined as follows: 1 for moving southward; 2 for moving westward or eastward; 3 for moving northward. We assume that the square labeled with 0 is the starting square and the goal square is labelled with "G" and all dark-shaded squares are obstacles.

Iterative Deepening Search (IDS) It is also called `iterative deepening depth-first search`. It is used in combination with depth-first graph search (with explored set), that finds the best depth limit. It does this by gradually increasing the limit – first 0, then 1, then 2, and so on – until a goal is found. This will occur when the depth limit reaches d, the depth of the shallowest goal node. Iterative deepening combines the benefits of depth-first and breadth-first search. Implement this algorithm for the above windy maze.

A* Search We use a modified Manhattan distance used in class as the heuristic function $h(n)$ by considering the windy situation. For example, for the start node, the agent has to move at least 2-step eastward and 1-step northward in order to reach the goal. Therefore, we have $h(n) = 2 * 2 + 3 * 1 = 7$ at the start node.

We use a label we did in class to indicates the order of choosing the corresponding unlabeled square and adding it to the frontier. To break tier for unlabeled squares (expanding children nodes), use this order: first westward; then northward; then eastward; then southward. To break tier for labeled squares (picking one child node to expand), the smallest label is picked first.

Follow the same way as done in the class to show the search steps with labels inside circles for the following search algorithms: Depth-first search and A* search (ignoring the subscripts which we use in class). Your outcome should be displayed as these:

```
IDS with depth from 1 to 9:
-----------------   -----------------   -----------------   -----------------
                                        06                  07  08
    ##  ##              04  ##  ##      05  ##  ##          06  ##  ##
01  ##                  01  ##          01  ##              01  ##
00  ##  ##              00  ##  ##      00  ##  ##          00  ##  ##
02  ##                  02  ##          02  ##              02  ##
                        03              03  04              03  04  05
-----------------   -----------------   -----------------   -----------------
-----------------   -----------------   -----------------   -----------------
09  10  11              11  12  13  14      13  14  15  16  17  15  16  17  18  19
08  ##  ##              10  ##  ##          12  ##  ##  18      14  ##  ##  20  21
01  ##                  01  ##              01  ##              01  ##      13
00  ##  ##              00  ##  ##          00  ##  ##  11      00  ##  ##  12  11
02  ##  06              02  ##  06  08      02  ##  06  08  10  02  ##  06  08  10
03  04  05  07          03  04  05  07  09  03  04  05  07  09  03  04  05  07  09
-----------------   -----------------   -----------------   -----------------
-----------------
15  16  17  18  19
14  ##  ##  20  21
01  ##  23  22  13
00  ##  ##  12  11
02  ##  06  08  10
03  04  05  07  09
-----------------

A* Search
-----------------
05  07  09  12  15
03  ##  ##  16  19
01  ##  22  20  23
00  ##  ##  17  21
02  ##  10  13  18
04  06  08  11  14
-----------------
```

2 Submission

Form a group on Canvas if you want to work with another student. Specify the contribution made by each member if you work as a group.

In your report, please provide the screenshots of all outcomes, and the highlighted code segments and the detailed explanation on how you implement the followings for each algorithm:

- Data structure: priority queue for frontier set and hash table for explored set;
- Calculation of $f(n) = g(n) + h(n)$;
- Adding leaves for expansion;
- Picking the smallest $f(n)$.

Each screenshot should include your usernames and the current time, which show that you did it by yourselves. If your output is different from the expected one, provide a reason for the cause. Comment your code in details so that the grader can understand it well.

Your assignment will be graded based on the above guideline and the rubric on Canvas. The report should be written in a ".docx", ".doc", or ".pdf" format. Submit the report and the source code to the assignment folder P1 on Canvas. Any compression file format such as `.zip` is not permitted.