CS 5100: Foundations of Artificial Intelligence (Spring 2022)	Robert Platt
Student:	Due September 26, 2022

Heuristic Search

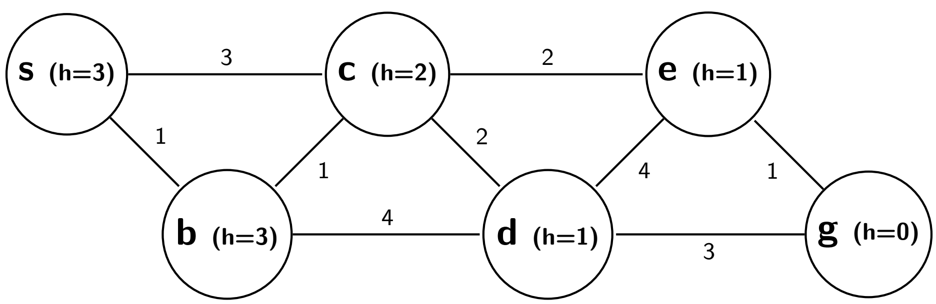
1 Problem 1 [5 points]

1. *This is a continuation of PS1 Q1, applying greedy search and A* to the same graph.*

   Consider the graph below. The agent starts at node s and must reach node g. All edges are undirected, so they can be traversed in both directions (recall this is equivalent to a pair of directed edges).
   Apply greedy search and A* to find a path from s to g.
   Use the graph search version of these algorithms. The heuristic value of each state are given by the h value in the node. Assume that nodes with the same priority are *added* to the queue/stack in alphabetical order. (This means that nodes are removed from a stack in *reverse* alphabetical order; also, note that "s" is part of the alphabet.) For algorithms that use a priority queue, also use alphabetical order to break ties when removing nodes with the same priority (e.g., (a,10) before (c,10)). Show:

   - The order in which nodes are expanded
   - The contents of the frontier after each expansion, in the correct order / with priorities if applicable
   - The contents of the explored set, if applicable
   - The solution path that is returned, if a solution is found

   Hint: You do not have to draw out the search tree. Also recall that nodes technically store *paths* instead of states, but it is fine to just write the corresponding state to denote the search node. The solution is the path stored in the final search node, *not* the list of states expanded.



2 Problem 2 [2 points]

(AIMA 3.21) Prove each of the following statements, or give a counterexample:

(a) Depth-first search is a special case of best-first tree search.

(b) Uniform-cost search is a special case of A* search.

3 Problem 3 [4 points]

(AIMA 6.3) Consider the problem of constructing (not solving) crossword puzzles for fitting words into a rectangular grid. The grid, which is given as part of the problem, specifies which squares are blank and which are shaded. Assume that a list of words (i.e., a dictionary) is provided and that the task is to fill in the blank squares by using any subset of the list. Formulate this search problem precisely, using two different strategies for filling in blanks:

1. Filling in blanks one letter at a time.
2. Filling in blanks one word at a time.

4 Problem 4 [2 points EXTRA CREDIT]

(Modified from AIMA 3.31.a.) Recall that we came up with two different heuristics for the 8-puzzle by relaxing problem constraints. In an 8-puzzle, the set of valid actions are described by the following statement:

> A tile can move from square A to square B if A is adjacent to B **and** B is blank.

We can generate three relaxed problems (leading to three admissible heuristic functions) by removing one or both of the above conditions:

- A tile can move from square A to square B if A is adjacent to B
- A tile can move from square A to square B if B is blank
- A tile can move from square A to square B

Recall that the first relaxation gives us the sum-of-Manhattan-distances heuristic, and the third relaxation gives us the number-of-misplaced-tiles heuristic. The second relaxation leads to a relaxed problem, whose (optimal) solution is known as *Gaschnig's heuristic* (Gaschnig, 1979). (Note that you do not need to look up the details of computing Gaschnig's heuristic - just know that it optimally solves the second relaxed problem.)

Explain why Gaschnig's heuristic is at least as accurate as the number-of-misplaced-tiles heuristic. Hint: Observe that since both are admissible heuristics, it is equivalent to showing that:

$$0 \leq \text{Number of misplaced tiles} \leq \text{Gaschnig's heuristic} \leq h^*$$

In the above, h^* is the true cost-to-go function. In particular, explain why the number of misplaced tiles is always an underestimate of Gaschnig's heuristic.