Fall 2019
CSCI (PHIL) 4550/6550 Introduction to AI
Assignment 2: Search

General Information

Deadline: 10:59 am, Thursday Sept. 26
Worth: 100 pts (undergraduates) 120 pts (graduates)

The Assignment

The purpose of this assignment is to make you experiment with **search** methods. Your grade will be based on the correctness of the solution. If pseudocode is required, please be as detailed as possible while writing the pseudocode. Furthermore, it is a good practice to include explanatory comments in your pseudocode.
Note: This assignment is not a group project and everybody should work on it individually.

In order to learn how to write in pseudocode, please see the Appendix B.2 of the textbook. Also, take a look at some examples written in pseudocode in Chapters 3 and 4 of the textbook.

1 Problems

1.1 Water and Jugs Problem (20 points)

You are given two jugs. One holds 4 gallons and the other 3 gallons of water. Assume that you are given no external measuring device, you can fill-up a jug from a pump any time you need, and you can pour water out of a jug or from one into the other. You will start from an initial state (each state would be the status of the two jugs) and get to a final state by a sequence of legal moves. The particular problem is to start from [0,0] (both jugs are

empty) and get to [2,0] (the 4-gallon jug has exactly 2 gallons of water and the three gallon jug is empty).

1. (10 points) Formulate the problem as a search problem by precisely defining the **states**, **initial state**, **transition model**, **goal test**, and **path cost**.

2. (10 points) Solve the problem using your favorite uninformed search technique (choose from breadth-first, depth-first, iterative deepening, or depth-limited search). **Show a trace of the algorithm on the search tree, by drawing the partial search trees that result as you apply the search technique.** See Figures 3.12, 3.16 in the textbook for reference. **Also show the complete solution path (the sequence of states) from the start state to the goal state.**

1.2 8-puzzle (20 points)

Experiment with different search heuristics. We will solve the 8-puzzle problem using different heuristics. See Section 3.2 of the textbook for the formulation of the 8-puzzle problem.

1. (10 points) Using the *Manhattan distance* heuristic (pg. 103), apply the A* method to the 8-puzzle problem. **Show the stages of the A* search with the heuristic values of each node** (for reference see Figure 3.24). If your goal-test hasn't been successful and your search tree has exceeded a depth of 10, you need not expand any further.

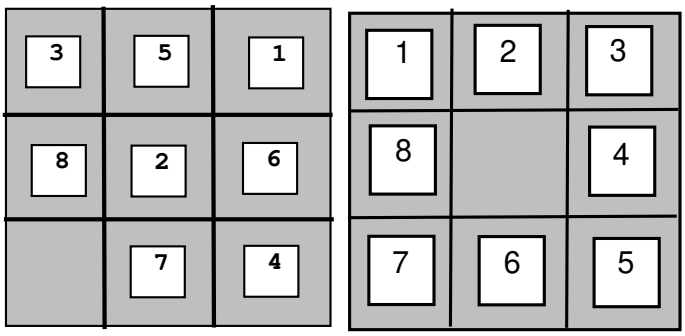

Figure 1: Use the left configuration of the 8 puzzle as your start state and the right configuration as your goal state.

2. (10 points) Consider a new heuristic called *n-MaxSwap* defined as follows: Assume you can swap any tile with the "space". Use the number of steps it then takes to reach the goal state, as the heuristic value. Using the n-MaxSwap heuristic, apply the A* method to the 8-puzzle problem. Show the stages of the A* search with the heuristic values of each node. If your goal-test hasn't been successful and your search tree has exceeded a depth of 10, you need not expand any further. Use the goal state as shown in Fig. 1.

Tip: You may wish to "program" the above heuristic rules in your favorite programming language so that the heuristic value of a state is obtained quickly and correctly.

1.3 The dating game (40 points)

The dating game consists of three males (M), three females (F), and an empty chair. The initial configuration is as follows:

M	M	M		F	F	F

The game has two legal moves with associated costs:

1. A person (male or female) may move into an adjacent empty chair. This has a cost of 1.

2. A person can jump over one or two other persons into the empty chair. This has a cost equal to the number of persons jumped over.

The goal is to pair each male with some female. The goal configuration is shown below:

M	F	M	F	M	F	

Note that though other paired configurations exist, we are only interested in this configuration.

1. (20 points) Formulate the problem as a search problem by defining the **states**, **initial state**, **transition model**, **goal-test**, and **path cost**.

2. (10 points) Define an appropriate heuristic for this problem. Please include comments about what the heuristic measures, and whether it is admissible.

3. (10 points) Apply the IDA* search method along with the heuristic that you have defined, to the dating game. **Show the stages of the IDA* search using appropriately defined cutoffs on the f-cost.**

1.4 Reading assignment (20 points)

Please read 'Chapter 1 - Understanding the Natural and Artificial Worlds' of the book, "The Sciences of the Artificial, Herbert Simon". Please prepare a **two-page** summary of the chapter, including half a page dedicated to your views on the chapter.

Note: Next problem is for graduate students only

1.5 Grid World (20 points)

In this problem we will utilize a simple grid world. We will assign a goal for the agent, and use search techniques to reach the goal. This is the first step toward making the agent autonomous (see the algorithm on page 67 of the textbook).

Consider a Grid-world with the following specifications:

- A 3×3 grid of locations
- A single agent, A, that starts from location (1,1) and can move `left`, `right`, `up`, and `down`.
- A single fixed gold piece, G, that is present in the location (3,2).

The corresponding Grid-world is shown below.

```
-------------
|   | G |   |
-------------
|   |   |   |
-------------
| A |   |   |
-------------
```

First formulate the problem by defining the **state**, **initial state**, **operators**, **goal-test**, and the **path cost function**. Then apply the IDA* search

to find a path from the initial agent location to the gold piece. Show the stages of the IDA* search using appropriately defined cutoffs on the f-cost. Also, write down the sequence of operators that lead the agent to the gold piece.

What and how to hand it in

You'll submit the **typed** pseudocode for your functions, the traces and the summary. The document should include your name, student id, and all the function definitions and search trees. Please include explanatory comments in your pseudocode as much as possible.

Please submit your assignment as a **single zipped file** by the deadline using eLC.

Assignments that are **late** but within a day of the deadline will be penalized 33% of the total number of points. Assignments submitted later than one day will not be accepted.