COMP-424: Artificial intelligence, Winter 2018

Homework 1

Due on *myCourses* Tuesday Jan 30, 11:59pm.

General instructions.

- This is an individual assignment. You can discuss solutions with your classmates, but should only exchange information orally, or else if in writing through the discussion board on *myCourses*. All other forms of written exchange are prohibited.
- Unless otherwise mentioned, the only sources you should need to answer these questions are your course notes, the textbook, and the links provided. Any other source used should be acknowledged with proper referencing style in your submitted solution.
- For each problem, you can solve manually, or write a program to help you. You can use a programming language of your choice. You can modify code from other sources if you provide adequate citation; this cannot be code from other students in the class.
- Submit a single pdf document containing all your pages of your written solution on your McGill's *myCourses* account. You can scan-in hand-written pages. If necessary, learn how to combine many pdf files into one.
- Submit any code developed to answer questions as a separate file to McGill's *myCourses*.

Question 1: Six-Puzzle

* You will probably find it easier to write a program to solve this problem.

Consider a variant of the eight-puzzle, which we saw in class, where we reduce the size of the puzzle to 3x2, in order to keep the number of states manageable. Now, consider the following initial and goal states:

Initial state:

1	4	2
5	3	

Goal state:

	1	2
5	4	3

a) Show the solution path (i.e., the sequence of puzzle states from the initial to the goal state) found by each of the following algorithms, assuming transitions have unit cost. **You must ensure that puzzle states that have been explored are not added to the search queue.** Given multiple states to explore that are otherwise equivalent in priority, the algorithm should prefer the state that involves moving a lower-numbered piece.
   i. Breadth first search
   ii. Uniform cost search
   iii. Depth first search
   iv. Iterative deepening

b) Suppose now that transitions have differing costs. In particular, the cost of a transition is equal to the number of the piece that is moved (e.g., moving the "4" costs 4). If we employ the Manhattan distance heuristic for the original unit cost version of the eight-puzzle presented in class (Lecture 3, slide 13, h_2), would this heuristic still be an admissible heuristic for A* search in the new variant? Justify your answer.

c) Design an admissible heuristic that dominates the heuristic from part b, under the same transition cost scheme as part b.

d) Now consider another variant of the problem in which moving pieces to the left or right costs 2, whereas moving pieces up or down costs 0.5. Would the Manhattan distance heuristic from part b still be admissible? Justify your answer.

Question 2: Search algorithms

* You will probably find it easier to solve this problem by hand.
(From Russell & Norvig)

a) Describe a state space in which iterative deepening search performs much worse than depth-first search (for example, $O(n^2)$ vs $O(n)$).

Prove each of the following statements, or give a counterexample:

b) Breadth-first search is a special case of uniform-cost search.
c) Depth-first search is a special case of best-first tree search.
d) Uniform-cost search is a special case of A* search.

Question 3: Optimization

* You will probably find it easier to write a program to solve this problem.

Consider the following function: $Y = \sin(X^2/2)/\log_2(X+4)$, in the range $X=[0, 10]$.

a) Apply hill-climbing, starting from each of the following starting points: $X_0=\{0, 1, 2, \ldots, 10\}$ and step sizes $\Delta X = \{0.01, 0.02, \ldots, 0.1\}$.
b) Repeat using simulated annealing with a range of different temperatures and annealing schedules (of your choosing). Consider each of the different starting points in (a). Consider only those step sizes from part (a) that produced a good result (use your own judgment in defining this; briefly explain your reasoning and method.)

For each part report the number of steps to convergence and final result (X*,Y*) for each case. Use plots and/or tables to report your results in an organized manner. Don't forget to include labels & captions.

Question 4: Constraint satisfaction

* You will probably find it easier to solve this problem by hand.
(Adapted from Russell & Norvig)

Consider the problem of placing *k* rooks on an *n* x *n* chessboard such that no two rooks are attacking each other, where *k* is given and $k \leq n^2$.

a) Formulate this problem as a Constraint Satisfaction problem. Describe the set of variables, and their domain. List the set of constraints.

b) Show the search tree generated with applying backtracking search, without forward checking for the problem with $k = 3$ and $n = 3$. Show clearly the domain for each variable at every point in the search.

c) Show the search tree generated with applying backtracking search, this time using forward checking for the problem with $k = 3$ and $n = 3$.