CPSC 322 Midterm Practice Questions

The midterm will contain some questions similar to the ones in the list below. Some of these questions may actually come verbatim from the list below. Unless otherwise noted, answers to the following questions should be a only two or three sentences long.

1. What is AI?

1. In class, various definitions of AI were given. It was argued that AI should be defined as "systems that act rationally". Give another definition of AI that was discussed in class, and in no more than two sentences, explain why the \acting rationally" definition is to be preferred.

2. Describe the difference between a deterministic representation and a stochastic representation of an AI system.

3. Explain why the following statement is true or false: In order to pass the Turing test, a computer would have to behave at least as rationally as a human.

4. Compare and contrast an AI system that thinks and acts humanly with one that thinks and acts rationally

2. Search

1. State the five elements that characterize a search problem instance/search algorithm.

2. What is meant by search algorithm *completeness*?

3. What is meant by search algorithm *optimality*?

4. Explain the distinction between optimality and optimal efficiency for search algorithms.

5. What are the advantages of breadth-first search (BFS) over depth-first search (DFS)? What is the advantage of DFS over BFS?

6. Explain what the frontier is in a graph-search problem, and what it is used for.

7. What node(s) is/are in the frontier of a depth-first search?

8. Where b is the maximum branching factor and d is the maximum depth of the search, characterize the maximum size of a DFS frontier.

9. Is the worst-case time complexity different for DFS and BFS? Why or why not?

10. Consider the following generic search algorithm:

```
1    Input: a graph,
             a set of start nodes,
3            Boolean procedure goal(n) that tests if n is a goal node.
4    frontier := {<s> : s is a start node};
5    while frontier is not empty:
6            select and remove path <n_0, … , n_k> from frontier;
7            if goal(n_k)
8                    return <n_0, … , n_k>;
9            for every neighbor n of n_k
10                   add <n_0, … , n_k, n> to frontier;
11   end while
```

Imagine that this generic algorithm was used as the basis for an implementation of depth-first search and of breadth-first search. Which line or lines of the pseudocode above must have different implementations? Briefly, how would those implementations differ?

11. What is the distinction between informed and uninformed search?

12. What is a heuristic?

13. Give the definition of an admissible heuristic.

14. Is the max of two admissible heuristics also admissible? Why or why not?

15. Consider two admissible heuristics h1 and h2. Which one of the following options would yield a better heuristic for use with A*? Briefly explain your answer.

(a) min(h_1, h_2)
(b) max(h_1, h_2)
(c) (h_1+h_2)/2
(d) A* will have the same performance in all cases.

16. Consider a search problem in which all paths have bounded length. In what sense is branch and bound better than A*? In what sense is A* better than branch and bound?

17. In branch and bound, how is the upper bound (UB) calculated?

18. In branch and bound, when do we prune a path *p*?

19. Assume you run uninformed iterative deepening to find solution no more than k steps from the start node (k > 2). In the worst case, how many times the nodes two steps from the start node will be generated? Why?

20. Assume uninformed iterative deepening is running. If the algorithm has reached the stage in which DFS is running with bound depth 6: must it be true that the start state is not

the goal? Why? Must it be true that there are no solutions at level 3? Why?

3. CSPs

1. How many states can be represented by eight variables, each of which can take four values?

2. How many states can be represented by three variables, each of which can take three values? How many states are there for a 3-queen problem?

3. State the condition under which an arc <X, r(X, Y)> is arc consistent.

4. How can we enforce consistency of an arc <X, r(X, Y)>?

5. What does it mean for a network to be arc consistent?

6. What are the possible outcomes of the arc consistency algorithm?

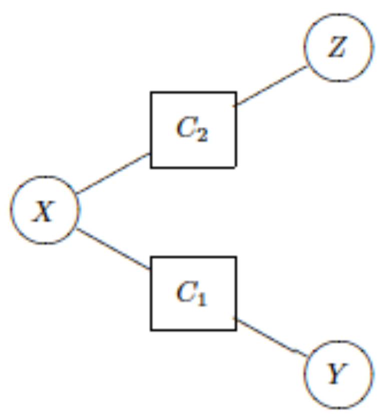

Figure 1: A constraint graph

7. Consider the constraint graph in Figure 1. Imagine that arc consistency has just reduced the domain of X as a result of considering the edge <X, C_1>. Do we have to add the edge <X,C_2> back to the list of "to-do arcs"? Explain why or why not (i.e., don't just state a rule given in class).

8. Consider the constraint graph in Figure 1. Imagine that arc consistency has just reduced the domain of X as a result of considering the edge <X, C_1>. Do we have to add the edge <Y,C_1> back to the list of "to-do arcs"? Explain why or why not (i.e., don't just state a rule given in class).

9. Consider the constraint graph in Figure 1. Imagine that arc consistency has just reduced the domain of X as a result of considering the edge <X, C_1>. Do we have to add the edge <Z,C_2> back to the list of "to-do arcs"? Explain why or why not (i.e., don't just state a rule

given in class).

10. Consider the following statement: every problem that can be defined as a CSP where variables have finite domains and with a finite number of variables can also be defined as a search problem using a state-based representation. Explain why it is true or false; if true, explain what would correspond to a state, and if false, give a counter-example.

11. How can a CSP be solved by search (specify states, neighbors, start state, goal state). What search strategy would you use? Why?

4. Local Search

1. What is the state space of local search for a CSP?

2. What are two key limitations of local search?

3. Define a plateau. Why are plateaus a problem for local search?

4. Why does randomization help in local search?

5. Explain how the next state is chosen in simulated annealing.

6. How would you change a local search algorithm to better find global maxima that are surrounded by plateaus? Assume that the neighbor relation is fixed.

7. How would you design a local search algorithm for a problem where there are no plateaus and no local minima that are not global minima? Assume that the neighbor relation is fixed.

8. What is the difference between a random step and a random restart?

9. Consider two local search algorithms, A and B. A solves x% of the problems it is given within $(.25x)^2$ minutes. B solves y% of the problems it is given within y minutes. Is one algorithm always better than the other? If not, which algorithm would you use when?

5. Planning

1. What is the STRIPS assumption?

2. What are the two parts of an action in STRIPS, and what is the meaning of each part?

3. In forward planning, what is the branching factor for a given state?

4. Explain the role of the horizon in CSP planning. How does the choice of horizon affect the behavior of the planner?

5. In a CSP encoding of a planning problem, what is the relationship between the horizon (h), the number of variables in the CSP (n), and the number of variables (v) and actions (a) in the planning problem?

6. In a CSP encoding of a planning problem, which variables are involved in an effect constraint?