Computer Science 431

Second Hour Exam

Name _____

Friday, Nov. 5
90 pts. (will be adjusted to 100 in the gradebook).

1. Some definitions about heuristic search (5 pts. each) Please define the following terms:

Heuristic

Admissible heuristic

Consistent heuristic

What does it mean to say that heuristic h_1 is "better informed" than heuristic h_2?

2. A* search (15 pts.)

Suppose we have an A* search in progress. S is the start node, and F is the final node. We have expanded node S with the following results (g, h, and f are as in lectures).

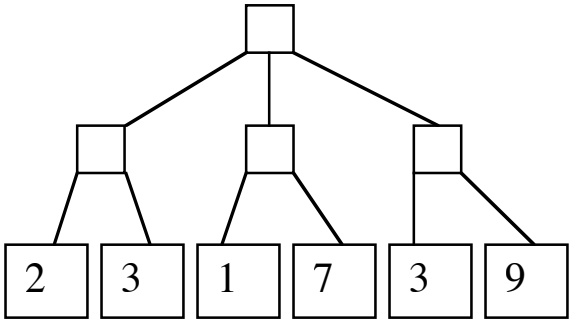
state	g(node)	h(node)	f(node)
S	0	3	
A	1.2	2	
B	1.3	2	
C		1	
D		1	
E		1	
F		0	

a. Finish the calculations for states S, A, and B (i.e., fill in the values of f(node) for these three nodes. The results should go into the table.

b. Which node will be selected next for evaluation (the two possibilities are A and B)? Why is this node selected?

c. Suppose that we can get from A to C with cost (g) 1.2, from node A to D with cost 1.3, from B to D with cost 1.1, and from B to E with cost 1.3 (you will not need all of these values). Expand only the node you selected in step (b) and fill in the details (for the expansion of that node only) in the table. That is, if you expand node A, you will want to complete the table for nodes C and D. If you expand node B you will want to complete the table for nodes D and E. But (in accordance with your choice in part (b)), you should expand only one of the two nodes A and B.

3. (15 pts.) Consider the following game tree:



The maximizer is making the move at the top and has three choices. The minimizer gets to move next and returns board positions listed at the bottom.

a. Which move will the maximizer make, and what is the value of that move?

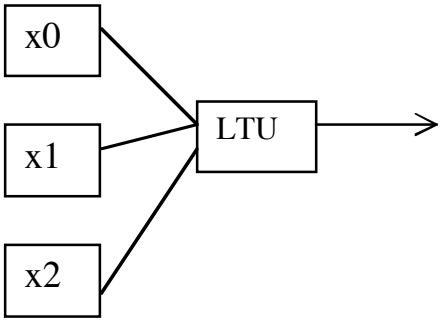
b. One of the nodes in the above diagram actually does not need to be evaluated. Identify the node and explain why it does not need to be evaluated.

4. (5 pts.) What are some of the problems associated with hill-climbing search? As a part of your answer, say what hill-climbing search does.

5. (10 pts.) Genetic algorithms can be applied to a number of problems in graph theory. Here is one of them. A graph of n vertexes is colored with k colors if each vertex is painted with one of the k colors in such a way that no adjacent vertex (i.e., one connected to this vertex by an edge) has the same color. A candidate coloring would be a n-tuple of colors (one for each vertex). For example, if we consider a graph of 5 tuples and are using three colors (red, blue, and green), then one candidate solution might be (red red green blue green). The first vertex is then colored red, the second also red, the third green, and so on.

Describe the general idea of genetic algorithms, using the example above. What would you want to consider in an evaluation function for this problem?

6. (15 pts.) Consider the following perceptron:



Where W0 is 1, W1 is 2, and W2 is –2.

a. Fill in the following table for the values of x1 and x2 given.

x1	x2	fires/fails to fire
0	0	
0	1	
1	0	
1	1	

b. What is the value of x0?

c. Draw the appropriate line and give a geometrical interpretation of the perceptron.

7. (10 pts.)

To the atom **‘my-computer** add the property **‘components** with the value **‘(processor monitor keypad)**

How would you ask the question “does ‘**my-computer** have a **keypad** as one of its ‘**components**?” in lisp?