

## CS-171, Intro to A.I. — Mid-term Exam — Fall Quarter, 2013

YOUR NAME AND ID NUMBER: \_\_\_\_\_

YOUR ID: \_\_\_\_\_ ID TO RIGHT: \_\_\_\_\_ ROW: \_\_\_\_\_ NO. FROM RIGHT: \_\_\_\_\_

**The exam will begin on the next page. Please, do not turn the page until told.**

**When you are told to begin the exam, please check first to make sure that you have all seven pages, as numbered 1-7 in the bottom-right corner of each page.**

**The exam is closed-notes, closed-book. No calculators, cell phones, electronics.**

**Please clear your desk entirely, except for pen, pencil, eraser, a blank piece of paper (for scratch pad use), and an optional water bottle. Please write your name and ID# on the blank piece of paper and turn it in with your exam.**

This page summarizes the points available for each question so you can plan your time.

- 1. (10 pts total, -2 pts each error, but not negative) STATE SPACE SEARCH.**
- 2. (5 pts total, -2 pts for each error, but not negative) TASK ENVIRONMENT.**
- 3. (5 pts total, -1 pt each wrong answer, but not negative) SEARCH PROPERTIES.**
- 4. (15 pts total, 5 pts each) ALPHA-BETA PRUNING NODE ORDERING.**
- 5. (20 pts total, -5 for each error, but not negative) RESOLUTION THEOREM PROVING.**
- 6. (20 points total, 4 pts each) CONSTRAINT SATISFACTION PROBLEMS.**
- 7. (25 pts total, 5 pts each) SEARCH.**



**1. (10 pts total, -2 pts each error, but not negative) STATE SPACE SEARCH.** The *Missionaries and Cannibals* problem is a classic brainteaser. It is a member of a class of river-crossing brainteasers that includes *Farmer to Market*, *Jealous Husbands*, and others (see [https://en.wikipedia.org/wiki/River-crossing\\_problems](https://en.wikipedia.org/wiki/River-crossing_problems)).

*Three missionaries and three cannibals must cross a river using a boat which can carry at most two people. The constraint is that, for both river banks, if there are missionaries present on the bank, they cannot be outnumbered by cannibals (if they were, the cannibals would eat the missionaries). The boat cannot cross the river by itself without people on board. How can all six people get safely across the river?*

We will call states where anyone gets eaten “forbidden” and consider them as part of the state space, but from which there is no return; i.e., there is no way to go to any other state (including back to the previous state) from such a forbidden state. All other states are “allowed.” It is always possible to go from an allowed state to some other state(s).

One way to represent this problem is as a state vector with components (M,C,B), where M=the number of missionaries on the wrong side, C=the number of cannibals on the wrong side, and B=the number of boats on the wrong side. Since everything starts on the wrong side, the start state is (3,3,1). The single goal state is (0,0,0).

**1.a.** (2 pts) What is the maximum branching factor counting BOTH allowed and forbidden states as children?

\_\_\_\_\_.

**1.b.** (8 pts, -2 for each error, but not negative) Draw the state space showing ONLY the ALLOWED states. I.e., do NOT show the FORBIDDEN states.

Represent an allowed state as a circle enclosing its state vector (M,C,B). Connect allowed states that are possible successors of each other. The start and goal states are shown. The first node expansion is done for you as an example.



\*\*\*\* TURN PAGE OVER AND CONTINUE ON THE OTHER SIDE \*\*\*\*

2. (5 pts total, -2 pts for each error, but not negative) **TASK ENVIRONMENT.** Your book defines a task environment as a set of four things, with the acronym PEAS. Fill in the blanks with the names of the PEAS components.

P \_\_\_\_\_ E \_\_\_\_\_ A \_\_\_\_\_ S \_\_\_\_\_

3. (5 pts total, -1 pt each wrong answer, but not negative) **SEARCH PROPERTIES.** Fill in the values of the four evaluation criteria for each search strategy shown. Assume a tree search where  $b$  is the finite branching factor;  $d$  is the depth to the shallowest goal node;  $m$  is the maximum depth of the search tree;  $C^*$  is the cost of the optimal solution; step costs are identical and equal to some positive  $\epsilon$ ; and in Bidirectional search both directions using breadth-first search.

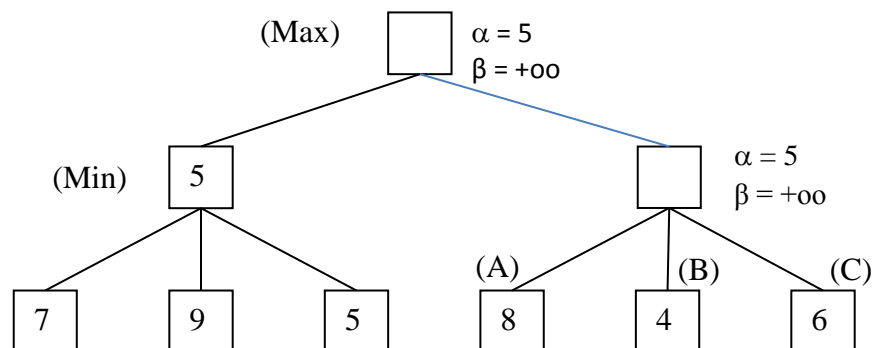
Note that these conditions satisfy all of the footnotes of Fig. 3.21 in your book.

Criterion	Complete?	Time complexity	Space complexity	Optimal?
Breadth-First				
Uniform-Cost				
Depth-First				
Iterative Deepening				
Bidirectional (if applicable)				

4. (15 pts total, 5 pts each) **ALPHA-BETA PRUNING NODE ORDERING.** Alpha-beta pruning effectiveness depends on the order in which node values are encountered. A bad node order can yield no pruning. A good node order can yield maximal pruning. This question asks about node ordering in relation to alpha-beta pruning effectiveness.

In the game tree below, the agent has searched the left branch and determined it is worth 5. It is about to search the right branch. How does the left-to-right order of the node values on the right branch affect how many nodes can be pruned in that branch?

**Below, write A, B, and C in an order that gives a correct answer to each question.**



4.a. (5 pts) What node ordering prunes two nodes? \_\_\_\_\_

4.b. (5 pts) What node ordering prunes one node? \_\_\_\_\_

4.c. (5 pts) What node ordering prunes zero nodes? \_\_\_\_\_

**5. (20 pts total, -5 for each error, but not negative) RESOLUTION THEOREM PROVING.**

You are engaged in Knowledge Engineering for the Wumpus Cave. You have interviewed an expert on the Wumpus Cave who told you, among other things, "A breeze in square (1,1) is equivalent to a pit in square (1,2) or a pit in square (2,1)." You translated this into propositional logic as, " $(B11 \Leftrightarrow P12 \vee P21)$ ," and then into Conjunctive Normal Form as " $(\neg B11 \vee P12 \vee P21) \wedge (\neg P12 \vee B11) \wedge (\neg P21 \vee B11)$ ."

Now it is time for the first "live" test of your system. An agent has been lowered down into the Wumpus cave, and reports back by radio, "Square (1,1) has a breeze. Also, I went into square (1,2) and I did not die, so it does not have a pit." You translate this knowledge into propositional logic as " $(B11) \wedge (\neg P12)$ " and add it to your knowledge base.

Next your system is asked to perform inference. The agent asks by radio, "Is it true that square (2,1) has a pit?" You translate this query into propositional logic as the goal sentence " $(P21)$ ." You form the negated goal as " $(\neg P21)$ ." Your knowledge base plus negated goal is:

- $(\neg B11 \vee P12 \vee P21)$
- $(\neg P12 \vee B11)$
- $(\neg P21 \vee B11)$
- $(B11)$
- $(\neg P12)$
- $(\neg P21)$

Run resolution on this knowledge base until you produce the null clause, " $(\quad)$ ", thereby proving that the goal sentence is true. The shortest proof I know of is only three lines long. It is OK to use more lines, if your proof is correct. **SHOW YOUR WORK.**

Repeatedly choose two clauses, write one clause in the first blank space on a line, and the other clause in the second. Apply resolution to them. Write the resulting clause in the third blank space, and insert it into the knowledge base.

Resolve \_\_\_\_\_ and \_\_\_\_\_ to give \_\_\_\_\_

Resolve \_\_\_\_\_ and \_\_\_\_\_ to give \_\_\_\_\_

Resolve \_\_\_\_\_ and \_\_\_\_\_ to give \_\_\_\_\_

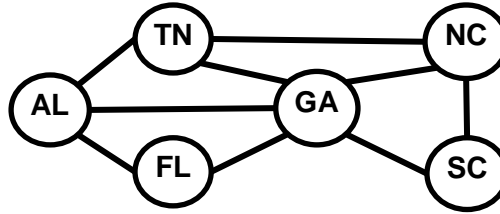
Resolve \_\_\_\_\_ and \_\_\_\_\_ to give \_\_\_\_\_

Resolve \_\_\_\_\_ and \_\_\_\_\_ to give \_\_\_\_\_

Resolve \_\_\_\_\_ and \_\_\_\_\_ to give \_\_\_\_\_

**\*\*\*\* TURN PAGE OVER AND CONTINUE ON THE OTHER SIDE \*\*\*\***

6. (20 points total, 4 pts each) CONSTRAINT SATISFACTION PROBLEMS.



You are a map-coloring robot assigned to color this Southeast USA map. Adjacent regions must be colored a different color (R=Red, B=Blue, G=Green). The constraint graph is shown.

6a. (4pts total, -2 each wrong answer, but not negative) FORWARD CHECKING.

Cross out all values that would be eliminated by Forward Checking, after variable GA has just been assigned value G, as shown:

AL	TN	FL	GA	NC	SC
R G B	R G B	R G B	G	R G B	R G B

6b. (4pts total, -2 each wrong answer, but not negative) ARC CONSISTENCY.

AL and FL have been assigned values, but no constraint propagation has been done. Cross out all values that would be eliminated by Arc Consistency (AC-3 in your book).

AL	TN	FL	GA	NC	SC
B	R G B	R	R G B	R G B	R G B

6c. (4pts total, -2 each wrong answer, but not negative) MINIMUM-REMAINING-VALUES HEURISTIC.

Consider the assignment below. TN is assigned and constraint propagation has been done. List all unassigned variables that might be selected by the Minimum-Remaining-Values (MRV) Heuristic: \_\_\_\_\_.

AL	TN	FL	GA	NC	SC
R B	G	R G B	R B	R B	R G B

6d. (4pts total, -2 each wrong answer, but not negative) DEGREE HEURISTIC.

Consider the assignment below. (It is the same assignment as in problem 6c above.) TN is assigned and constraint propagation has been done. List all unassigned variables that might be selected by the Degree Heuristic: \_\_\_\_\_.

AL	TN	FL	GA	NC	SC
R B	G	R G B	R B	R B	R G B

6e. (4pts total) MIN-CONFLICTS HEURISTIC. Consider the complete but inconsistent assignment below. GA has just been selected to be assigned a new value during local search for a complete and consistent assignment. What new value would be chosen below for GA by the Min-Conflicts Heuristic? \_\_\_\_\_.

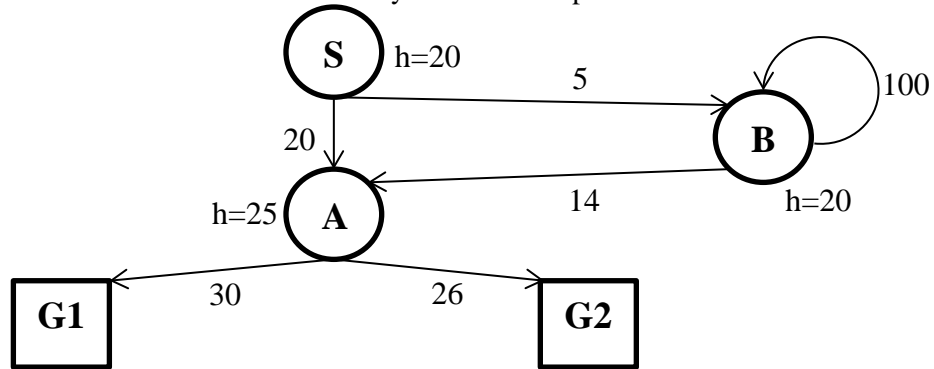
AL	TN	FL	GA	NC	SC
B	G	R	?	G	B

**7 (25 pts total, 5 pts each) SEARCH.** Execute Tree Search through this graph (i.e., do not remember visited nodes, so repeated nodes are possible). It is not a tree, but pretend that you don't know that.

Step costs are given next to each arc. Heuristic values are given next to each node (as  $h=x$ ). The successors of each node are indicated by the arrows out of that node.

**Successors are returned in left-to-right order. (Note: B is a successor of itself).**

For each search strategy below, show the order in which nodes are expanded (i.e., to expand a node means that its children are generated), ending with the goal node that is found. Show the path from start to goal, or write "None". The first one is done for you as an example.



**7a. DEPTH FIRST SEARCH.**

Order of node expansion: S A G1

Path found: S A G1

**7b. (5 pts) BREADTH FIRST SEARCH.**

Order of node expansion: \_\_\_\_\_

Path found: \_\_\_\_\_

**7c. (5 pts) UNIFORM COST SEARCH.**

Order of node expansion: \_\_\_\_\_

Path found: \_\_\_\_\_

**7d. (5 pts) GREEDY (BEST-FIRST) SEARCH.**

Order of node expansion: \_\_\_\_\_

Path found: \_\_\_\_\_

**7d. (5 pts) ITERATED DEEPENING SEARCH.**

Order of node expansion: \_\_\_\_\_

Path found: \_\_\_\_\_

**7e. (5 pts) A\* SEARCH.**

Order of node expansion: \_\_\_\_\_

Path found: \_\_\_\_\_

**\*\*\*\* THIS IS THE END OF THE MID-TERM EXAM \*\*\*\***