CS 2710 / ISSP 2160: Artificial Intelligence
MIDTERM EXAM
Fall 2009

This exam is closed book and closed notes. It consists of several parts. The first part is true/false. The second part is problem solving. The third part is short answer. Each question is labeled with its point value. If you are spending too much time on a question, skip it and go on, coming back if you have time.

Student Name_______________________________________

Please also put your name on every page (in case pages get accidentally unstapled)

	Question	**Score**
I. True/False	**1-10 (20 pts)**	
II. Problem Solving	**70 points**	
Search	**1 (15 pts)**	
Constraints	**2 (15 pts)**	
Games	**3 (15 pts)**	
Propositional Logic	**4 (10 pts)**	
First Order Logic	**5 (15 pts)**	
III. Short Answer	**1-3 (10 pts)**	

Total	**100 pts**	

Part I – True/False. 20 points total.
Each problem is worth 2 points. Incorrect answers are worth 0 points. Skipped questions are worth 1 point.

1. True/False. Depth-first search is an optimal uninformed search technique.

2. True/False. On problems with many repeated states, GRAPH-SEARCH is more efficient than TREE-SEARCH.

3. True/False. The minimum remaining value heuristic is a domain-dependent method for deciding which variable to choose next in a backtracking search.

4. True/False. Every existentially quantified sentence in first-order logic is true in any model that contains exactly one object.

5. True/False. The clauses P(F(A,G(A))) and P(F(x,y)) resolve to produce the empty clause.

6. True/False. Simple hill climbing is a complete algorithm for solving constraint statisfaction problems.

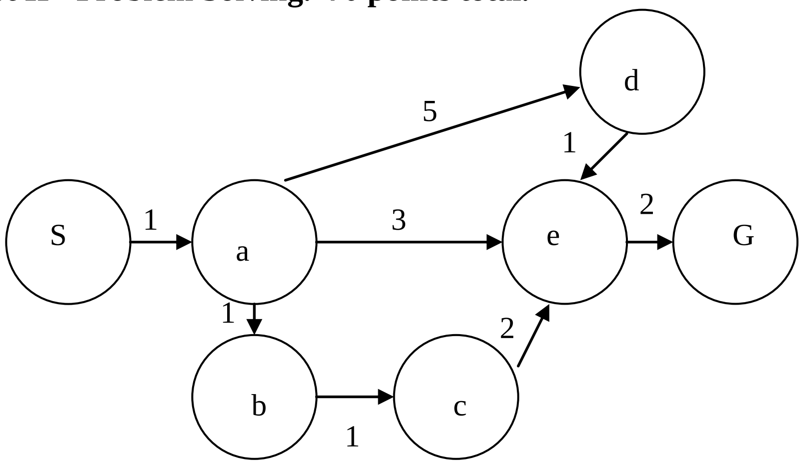
7. True/False. Breadth-first search is complete if the state space has infinite depth but finite branching factor.

8. True/False. Truth tables can be used to establish the truth or falsehood of any propositional sentence.

9. True/False. Forward chaining is complete for first-order logic.

10. True/False. Minimax and alpha-beta can sometimes return different results.

Part II - Problem Solving. 70 points total.



1. Search [15 points] In the graph above, S is the *Start State*, G is the *Goal State*, and the integers on the arcs are the *Costs*.

a) (3 points) Show the node expansion order using **Depth First Search**. Assume that the algorithm explores nodes in alphabetical order whenever possible.

b) (12 points) Consider three heuristics *H1*, *H2*, *H3*. The table below indicates the estimated cost to goal (h value) for each of the heuristics for each node in the search graph.

node	*H1*	*H2*	*H3*
S	6	6	6
a	5	5	6
b	5	4	5
c	4	2	3
d	2	1	2
e	2	1	1
G	0	0	0

A. **(3 points)** Which of the three heuristics *(H1, H2, H3)* are admissible?

B. **(5 points)** Show the node expansion order using **A*** for *H1*. To get any partial credit, you need to show your work by annotating f(n), g(n), and h(n).

C. **(4 points)** Show the node expansion order using **Greedy Best First Search** for *H2*. Show your work by annotating f(n), g(n), and h(n) to enable partial credit.

2. Constraint Satisfaction [15 points]

Four senators show up in the Senate: two demublicans (A and B) and two repocrats (C and D). Each chooses whether to sit on the east or west side, and whether to vote "Yes," "No," or "Waffle." We would like to determine where everyone sits and how they vote. We know the following facts:

- No demublican will vote the same as a repocrat.
- A, B, and C sit together.
- Senator A decides where to sit and then determines his vote based on that: east means no and west means yes.
- Senator B votes yes if some repocrat sits on the east. Otherwise he may vote yes or no.
- Senator D doesn't waffle.
- If senator D doesn't sit to the west, senator C won't either.

a) (7 points) This problem can be formulated as a CSP with 8 variables (votes and seats) where all the constraints are binary. Draw the constraint graph. Use capital letters *A, B, C, D* for variables whose domains are vote assignments for senators A, B, C and D respectively and lowercase *a, b, c, d* for variables whose domains are seat assignments for senators A, B, C, and D respectively.

b) (6 points) For any edges adjacent to d, write down the set of partial assignments the corresponding constraint permits. (Hint: here is what the answer would be for the edge between *A* and *a*: *{{A = Yes, a= West}; {A= No, a = East}}*).

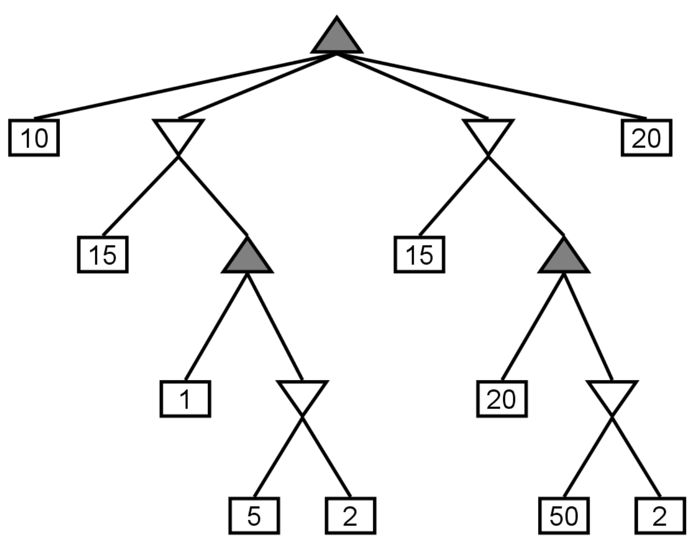
c) (2 points) True/False: The following is a consistent assignment for the entire CSP:
Everyone sits on the west. A and B vote yes while C and D vote no.

3. Games [15 points]

a) (7 points) In the matchstick game, a state consists of some number of matchsticks. Two players alternate turns, taking away either one or two matchsticks on each turn. The player that takes the last matchstick loses (utility -1), and the other player wins (utility +1). Draw the entire game tree for the game starting with 3 matchsticks (assuming the first player is MAX). Next to each node, write the corresponding minimax utilities, assuming optimal play by both players.

b) (8 points) The following questions refer to the minimax tree below. Assume nodes are evaluated in left-to-right order, MAX plays first, and minimax with *alpha-beta pruning* is used.

- **(2 points) True/False**. The minimax value for the root is 50.
- **(4 points)** Put a slash through links corresponding to pruned subtrees (if any).
- **(2 points) True/False**. The leaf that would be reached by optimal play is 20.



4. Propositional Logic; Unification in FOL [10 points]

a) (4 points) Use model checking to show whether $(A \wedge B) \Rightarrow C$ *entails* $(A \Rightarrow C) \vee (B \Rightarrow C)$

c) (3 points) Unify *R(F(y),y,x)* and *R(x, F(A),F(v))*

c) (3 points) Convert $(P \wedge \neg R) \Rightarrow (Q \Rightarrow R)$ into a Horn Clause

5. First-Order Logic Inference [15 points]

a) (8 points) Assume the following Knowledge Base and Goal.

KB = {On(a,b), On(b,c), Green(a), ¬Green(c)}

Goal = $\exists x \exists y$ [On(x,y) ∧ Green(x) ∧ ¬Green(y)]

Using proof by refutation and resolution as the single inference rule, show the resolution proof (3 points) that proves or disproves the goal. Be sure to show the conversion to Conjunctive Normal Form (3 points), and any unifications required (2 points).

b) (7 points) Consider the following KB:

1. *Buffalo(x) ∧ Pig(y) => Faster(x,y)*
2. *Pig(y) ∧ Slug(z) -> Faster (y,z)*
3. *Faster(x,y) ∧ Faster (y,z) => Faster(x,z)*
4. *Buffalo(Bob)*
5. *Pig(Pat)*
6. *Slug(Steve)*

A. (5 points) Use forward chaining to prove *Faster(Bob,Steve).* If several rules apply, use the one with the smallest number. You can either use a proof tree, or can show the chaining process step by step, using the numbering of the sentences to identify how you are using the rules and facts in the KB. For either presentation method, you will need to indicate the unifications.

B.(2 points) Give an example showing a problem that could happen given the KB above, if you used an inference mechanism that was

- Unsound

- Incomplete

Part III - Short Answer. 10 points total.

1. [5 points] Give broad conditions under which A* search reduces to Breadth First search.

3. **[5 points]** A patriotic Latin square is a *3 x 3* table in which each cell is filled with one of the *3* colors *R* (red), *W* (white), *B* (blue) Every color must appear in each row and column exactly once. Here is an example of a valid square:

R	W	B
B	R	W
W	B	R

Here is a partially filled square that cannot be completed.

R		B
B		
		W

Formulate the Patriotic Latin Square Problem as a **search** problem.