15-381 - Fall 2001
MIDTERM EXAM: SOLUTIONS
Thursday, October 11, 2001

OPEN UP TO FIVE PAGES OF NOTES

Read the questions carefully, and write your answers into the space provided on the exam itself. You may use the backs of the sheets if you need additional space.

Sign your name here *and on the top of each page*. Thank you!

NAME: ______________________________

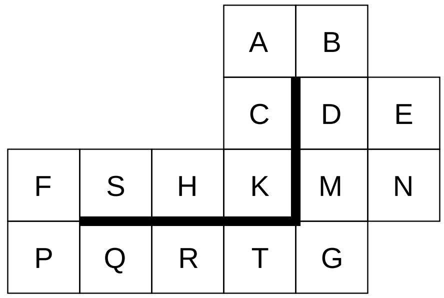
USER ID: ______________________________

GOOD LUCK!

Problem	Description	Max Score	Score
1	Search	15	______
2	Search	15	______
3	Games	10	______
4	Logic	15	______
5	Information Retrieval	15	______
6	Constraint Satisfaction	15	______
7	Planning	15	______
	Total	100	______

Problem 1: Search [*15 points*]

1. Consider the following maze in which the successors of a cell include any adjacent cell in the directions North, South, East, and West of the current cell, except at the boundary of the maze or when a barrier (thick line) exists. For example, $\text{successors}(\mathsf{M}) = \{\mathsf{D}, \mathsf{N}, \mathsf{G}\}$. Assume each move has cost 1.



The problem is to find a path from cell S to cell G. A search method *breakes ties*, if needed, using the alphabetical order of the labels in the cells. So Breadth-First Search (without duplicates) would visit cells in the order: S F H P K C Q A R B T D G.

For parts (a-d) below, what is the order of nodes expanded (plus the goal node if it is found) by each of the following search methods?

(a) Depth-First Search. Assume cycles are detected and eliminated by never expanding a node containing a state that is repeated on the path back to the root.

S F P Q R T G

(b) Greedy Search. Use as the heuristic function $h(state)$ = Manhattan distance from *state* to G assuming there were *no* barriers. For example, $h(\mathsf{K}) = 2$ and $h(\mathsf{S}) = 4$.

S H K C A B D M G

(c) Hill-Climbing Search. Use the same heuristic function as in (b).

S H K

(d) A* Search. Use the same heuristic function as in (b). Remove redundant states.

S H K C F P Q R T G

(e) Is h an admissible heuristic? Justify your answer.

Yes, it never overestimates the distance to the goal. In case the path is clear to the goal, the heuristic estimate exactly matches the shortest distance to the goal. If there is a barrier between the current position and the goal, the heuristic will underestimate the shortest distance because it acts as if there was no barrier.

(f) Is $h_2(state) = \min(2, h(state))$ an admissible heuristic? Justify your answer.

Yes, because it is never larger than h which is already an admissible heuristic.

(g) Is $h_3(state) = \max(2, h(state))$ an admissible heuristic? Justify your answer.

No. For example $h_3(\mathsf{T}) = 2$, while the shortest distance from T to the goal is only 1.

Problem 2: Search [*15 points*]

1. Assume you have an infinite state space with one state identified as the initial state and one as the goal state.

   All operators are reversible with forward-backwards branching factor equal 2, and the depth to the unique solution is 32. Express each answer as an exact calculation (leaving exponents, summations, etc without expansion if you wish) for full credit. Use big-O or an approximate calculation for partial credit.

   (a) How many states are visited in a bi-directional breadth-first search in the worst case?

      $2 \cdot 2^{16}$

   (b) How many states are visited in a forward-only breadth-first search in the worst case, with an island at d=8 and an island at d=16 (but no island at d=24)?

      $2^8 + 2^8 + 2^{16}$

   (c) How many states are visited in the worst case for depth-first search without depth bounds?

      There is no limit.

   (d) How many states are visited in the worst case for birectional breath-first search with islands at d=8, 16, and 24?

      $2 \cdot (2^8 + 2^4)$

2. In the usual way, consider that, for a search node n, $d(n), g(n)$, $h(n)$, and $f(n)$ represent respectively, the depth of n, the cost of reach n from the initial search node in the current path, the heuristic estimate of the cost of reaching the goal from n, and the total evaluation value of n.

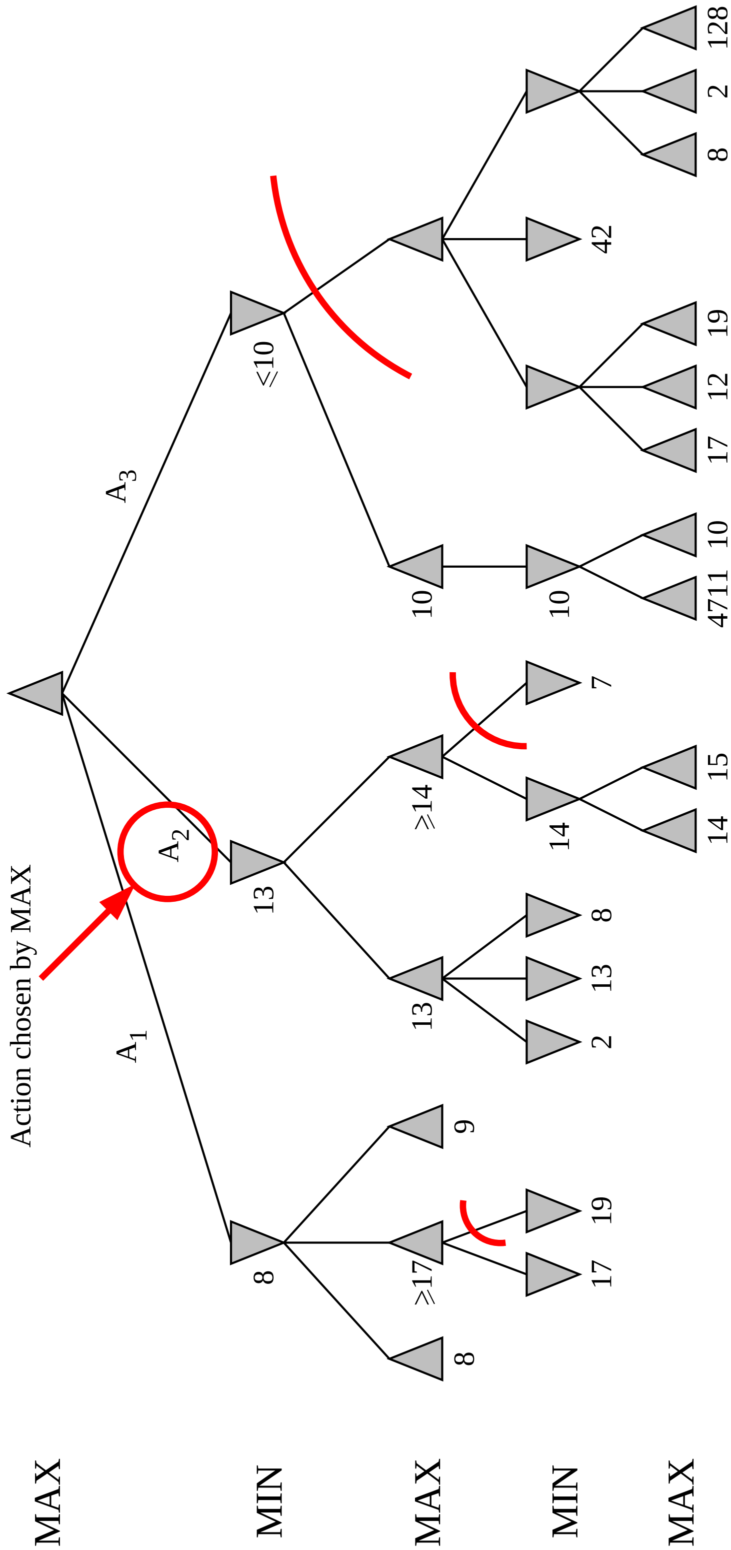
   Write expressions for f for the following types of search:

(a) Breadth-first search	$f(n) = d(n)$
(b) Uniform cost search	$f(n) = g(n)$
(c) Depth-first search	$f(n) = -d(n)$
(d) Best-first search	$f(n) = h(n)$
(e) A*	$f(n) = g(n) + h(n)$

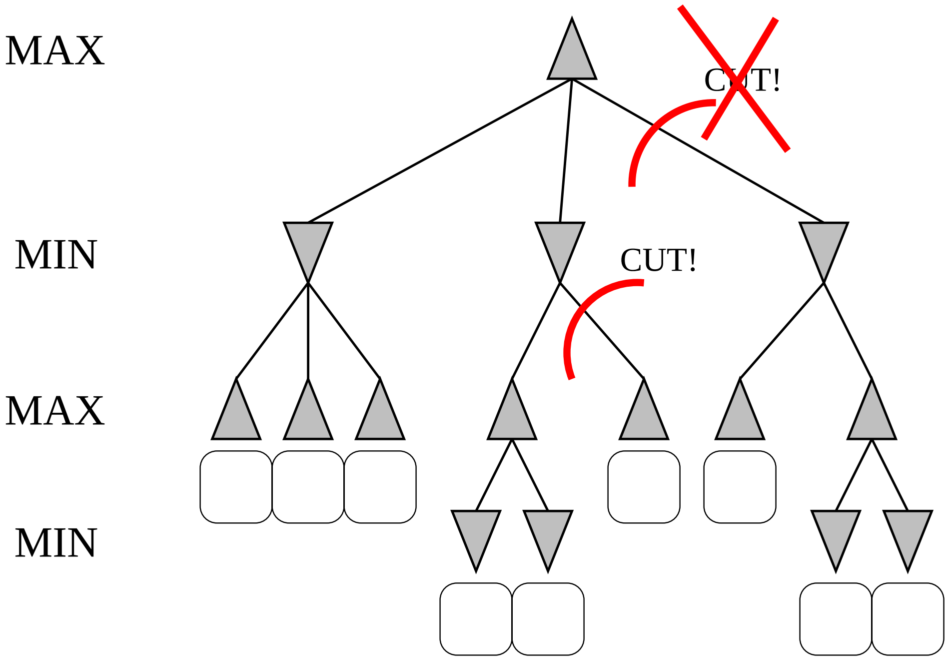
3. Could you perform a better search than exhaustive search if there are negative costs in paths? Justify your answer.

Problem 3: Games [*10 points*]

1. Show which branches (if any) would be pruned by alpha-beta. Write the propagated node values in the figure, and indicate which action MAX would choose.



2. Specify terminal values in the boxes below such that alpha-beta would make one or both of the two indicated cuts, *or* argue if one or both of the cuts cannot possibly be the result of alpha-beta pruning.



Without looking at any of the nodes in the subtree representing the third action MAX can perform at the top-level, we cannot possibly know that this action is worse than the best action seen so far. Because alpha-beta is conservative, it would never make such a cut.

Problem 4: Logic [*15 points*]

Consider a knowledge base consisting of the conjunction of the following propositions:

$$\neg A \implies B$$
$$B \implies A$$
$$A \implies (C \wedge D)$$

1. Compile the above propositions into a new knowledge base written in *disjunctive form*. **Number** the resulting clauses, as you will need to refer to these clauses later.

   1. $A \vee B$
   2. $\neg B \vee A$
   3. $\neg A \vee C$
   4. $\neg A \vee D$

2. Which of the clauses in your knowledge base (if any) are *not* Horn clauses? *Justify your answer.*

   $A \vee B$ is not a Horn clause, because it has more than one positive literal.

3. Prove the proposition $A \wedge C \wedge D$ using Modus Ponens only, *or* show that this is not possible.

   It is not possible to prove $A \wedge C \wedge D$ using Modus Ponens only. This follows from the fact that Modus Ponens is not applicable to any pair of clauses in the knowledge base.

4. Prove the proposition $A \wedge C \wedge D$ using resolution. *Indicate in your proof the propositions that you apply resolution to.*

   Proof by contradiction:

1.	$A \vee B$	premise
2.	$\neg B \vee A$	premise
3.	$\neg A \vee C$	premise
4.	$\neg A \vee D$	premise
5.	$\neg A \vee \neg C \vee \neg D$	assumption
6.	A	resolution 1, 2
7.	C	resolution 3, 6
8.	D	resolution 4, 6
9.	$\neg C \vee \neg D$	resolution 5, 6
10.	$\neg D$	resolution 7, 9
11.	$\perp$	resolution 8, 10

Problem 5: Information Retrieval [*15 points*] Consider the following collection of just two documents:

d_1 "State space search is a classical artificial intelligence paradigm, with an initial state and a goal state and.."

d_2 "NASA will search for its lost Martian space probe which never made it to martian orbit 10 months after launch..."

1. Represent each document as a vector: extract all unique words from the collection for your full vocabulary, alphabetize, remove stopwords, i.e., words in the set {a an and the of in to it its is for from which that}, and represent the vectors using only counts (term frequencies). Ignore stemming and morphology here.

   The words are: 10, after, artificial, classical, goal, initial, intelligence, launch, lost, made, martian, months, nasa, never, orbit, paradigm, probe, search, space, state, will, with

$$d_1 = \langle 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 3, 0, 1 \rangle$$
$$d_2 = \langle 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 2, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 0 \rangle$$

2. Represent as a vector and then calculate the cosine similarity of the query "state space" with each document.

$$q = \langle 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0 \rangle$$

$$\frac{d_1 \bullet q}{|d_1| \cdot |q|} = \frac{1 \cdot 1 + 3 \cdot 1}{\sqrt{18} \cdot \sqrt{2}} = \frac{2}{3}$$
$$\frac{d_2 \bullet q}{|d_2| \cdot |q|} = \frac{1 \cdot 1}{\sqrt{17} \cdot \sqrt{2}} = \frac{1}{\sqrt{34}}$$

3. If the query is "seek missing Mars spacecraft" what will the cosine similarity be to both documents? Specifically, what would you need to do so that this query has a chance of finding d_2?

   The similarity is 0. You need query expansion of near-synonyms via thesaurus (e.g. seek/search, missing/lost, Mars/Martian, ...), or find statistical correlations among such terms over a much larger corpus and use these to expand the query by adding highly-related terms before computing cosine similarty.

Problem 6: Constraint Satisfaction [*15 points*]

Consider a constraint satisfaction problem in which you have to find a value assignment for a set of variables restricted by a set of constraints. There are six boolean variables corresponding to the place and time that a class will take place namely: in Wean Hall, in-weh, in Hamburg Hall, in-hh, and in the College of Fine Arts, in-cfa, and at 10am or 8am in different days, at-10amMWF, at-10amTTh, and at-8am. There are seven constraints listed below given to the problem solver.

1. at-10amMWF $\vee$ at-10amTTh $\vee$ at-8am
2. $\neg$ at-10amMWF $\vee$ in-cfa $\vee$ in-hh
3. $\neg$ at-10amTTh $\vee$ in-hh
4. $\neg$ in-cfa $\vee$ $\neg$ in-weh
5. $\neg$ in-cfa $\vee$ $\neg$ in-hh
6. $\neg$ in-hh $\vee$ $\neg$ in-weh
7. in-weh

1. State in English the meaning of the constraint number 2 in the form of an if-then statement.

   If the class is at 10am on Mondays, Wednesdays, and Fridays, then the class is in the College of Fine Arts or in Hamburg Hall.

2. Apply the DPLL algorithm to find a truth assignment for all the six variables. Clearly show the sequence of constraints considered by your DPLL run.

   (a) Set in-weh to $true$ to satisfy constraint 7. The resulting constraint set:
       1. at-10amMWF $\vee$ at-10amTTh $\vee$ at-8am
       2. $\neg$ at-10amMWF $\vee$ in-cfa $\vee$ in-hh
       3. $\neg$ at-10amTTh $\vee$ in-hh
       4. $\neg$ in-cfa
       5. $\neg$ in-cfa $\vee$ $\neg$ in-hh
       6. $\neg$ in-hh

   (b) Set in-hh and in-cfa to $false$ to satisfy constraints 6 and 4. The resulting constraint set:
       1. at-10amMWF $\vee$ at-10amTTh $\vee$ at-8am
       2. $\neg$ at-10amMWF
       3. $\neg$ at-10amTTh

   (c) Set at10-amTTh and at-10amMWF to $false$ to satisfy constraints 3 and 2. The resulting constraint set:
       1. at-8am

   (d) Set at-8am to $true$ to satisfy constraint 1. This leaves no more constraints.

   The satisfying assignment has in-weh and at-8am set to $true$ and all other variables set to $false$.

3. Now remove the constraint number 7.

   (a) at-10amMWF $\vee$ at-10amTTh $\vee$ at-8am
   (b) $\neg$ at-10amMWF $\vee$ in-cfa $\vee$ in-hh
   (c) $\neg$ at-10amTTh $\vee$ in-hh
   (d) $\neg$ in-cfa $\vee$ $\neg$ in-weh
   (e) $\neg$ in-cfa $\vee$ $\neg$ in-hh
   (f) $\neg$ in-hh $\vee$ $\neg$ in-weh

   Show that there exists a truth assignment with at-10amTTh set to $true$.

   Set at-10amMWF and in-hh to $true$, and all other variables to $false$.

Problem 7: Planning [*15 points*]

Consider the following planning domain with three operators Op1, Op2, and Op3, and four literals, g1, g2, g3, and g4. A problem is defined by its Start and Finish steps.

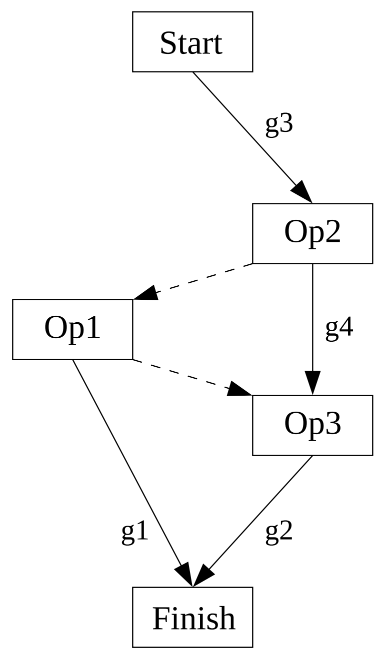
	Op1	Op2	Op3
pre	–	g3	g4
add	g1	g4	g2
del	g2 and g3	–	–

	Start	Finish
pre	–	g1 and g2
add	g2 and g3	
del	–	–

1. What is a plan to solve the given problem?

    Op2, Op1, Op3

2. Show the partial plan generated by a partial-order planner. In your drawing, clearly distinguish the causal links and the additional ordering links to resolve threats, by using solid arrows for the causal links and dashed arrows for the additional ordering links. Label the causal links with the goal they support.



EXTRA CREDIT:

Show that GPS, using means-ends analysis and linear planning, is will not be able to solve the problem. (Remember that means-ends analysis makes GPS plan only for goals that are not true in the state.)

	Op1	Op2	Op3
pre	–	g3	g4
add	g1	g4	g2
del	g2 and g3	–	–

	Start	Finish
pre	–	g1 and g2
add	g2 and g3	
del	–	–

For GPS, it is more convenient to think of the problem in terms of:

```
Initial state: g2 and g3
Goal statement: g1 and g2
```

Hint: Note that, by means-ends-analysis, GPS initially considers planning only for g1, as g2 is true in the initial state.

GPS proceeds as follows:

	Plan	**State**	**Unachieved goals**
1	-	g2 and g3	g1
2	Op1	g1	g2
3	Op1, Op3	g1 and g2	g4
4	Op1, Op3, Op2	g1 and g2 and g4	g3

At this point there is no operators that achieves g3, so GPS fails.