CIS 730

Introduction to Artificial Intelligence
Fall, 2001

Midterm Exam Solutions

Instructions and Notes

- You have 75 minutes for this exam. Budget your time carefully.
- Your answers on short answer and essay problems shall be graded for originality as well as for accuracy.
- You should have a total of 10 pages; ***write your name on each page***.
- Use only the front side of pages for your answers; you may add additional pages if needed.
- Select ***exactly one answer*** for each true/false and multiple choice question.
- Show your work on problems and proofs.
- In the interest of fairness to all students, no questions shall be answered during the test concerning problems. If you believe there is ambiguity in any question, ***state your assumptions***.
- There are a total of 200 possible points in this exam plus 10 points of extra credit.

Instructor Use Only

1. ______ / 40
2. ______ / 40
3. ______ / 40
4. ______ / 40
5. ______ / 40

Extra ______ / 10

Total ______ / 200

1. True/False (10 questions, 4 points each)

a) T <u>F</u> The traditional *Turing Test* is an interactive test between an artificial and a human agent. – *problem discarded*

b) <u>T</u> F An intelligent agent that maximizes *expected utility* is said to act *rationally*.

c) T <u>F</u> *Beam search* with beam width $w = 1$ is equivalent to greedy best-first search.

d) <u>T</u> F *A/A** with $g(n) = 0$ for every node n is equivalent to greedy best-first search.

e) <u>T</u> F The pathmax heuristic $f(n') = max\ (f(n), g(n') + h(n'))$ can convert any admissible heuristic into one that observes the monotone restriction.

f) T <u>F</u> In the worst case, *alpha-beta pruning* results in evaluation of a number of nodes asymptotically equal to the square root of the number expanded by *minimax* game tree search.

g) T <u>F</u> A *quiescent* branch of a game tree is one where an agent should devote additional computation in search.

h) <u>T</u> F The language of *satisfiable* propositional logical knowledge bases is recursive.

i) <u>T</u> F The language of *satisfiable* first-order logical knowledge bases is <u>not</u> recursive enumerable.

j) T <u>F</u> *Paramodulation* and *subsumption* are additional inference rules designed to handle equality in FOL sentences.

2. Multiple Choice (10 questions, 4 points each)

a) What is the *space complexity* of depth-first search with branch factor *b*, solution depth *d*, and maximum search depth *m*?

A) $O(bm)$
B) $O(bd)$
C) $O(b^m)$
D) $O(b^d)$
E) $O(b^{d/2})$

b) Which term describes a case in heuristic search where there may be zero gradient in the heuristic evaluation function?

A) I. Foothill problem (local optima that are not global optima)
B) II. Plateau problem – *also received credit although D was the intended answer*
C) III. Ridge problem
D) I and II but not III
E) I, II, and III

c) Which searches are guaranteed to achieve *optimal path cost* given an admissible heuristic?

A) I. Algorithm A
B) II. Breadth-First Search
C) III. Branch-and-Bound Search
D) I and II but not III
E) I, II, and III

d) Which of the following types of search produce *anytime* results?

A) Algorithm A with an admissible heuristic
B) Algorithm A with a monotonic heuristic
C) Iterative-deepening A* (IDA*)
D) All of the above
E) None of the above

e) Which of the following components is not part of the formal definition of a two-person game?

A) Initial state (starting position)
B) Operators (legal moves)
C) Terminal test
D) Utility (payoff) function
E) None of the above (all are part of the formal definition of a two-person game)

f) **What term describes the proliferation of axioms that describe non-interference of actions with the state of the world?**
   A) Representational frame problem
   B) Inferential frame problem
   C) Ramification problem
   D) Qualification problem
   E) None of the above

g) **What term describes an axiomatic solution to the proliferation of axioms that describe non-interference with the state of the world?**
   A) Kolmogorov axioms
   B) Successor-state axioms
   C) Situation calculus
   D) First-order predicate calculus
   E) Relational calculus

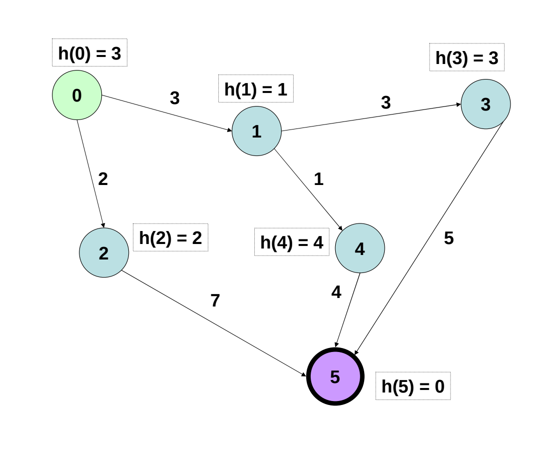
h) **What term describes the proliferation of axioms that describe the occurrence of side-effects from actions?**
   A) Representational frame problem
   B) Inferential frame problem
   C) Ramification problem
   D) Qualification problem
   E) None of the above

i) **Which of the following methods restricts the resolvents allowed in each application of the resolution inference rule?**
   A) I. Unit resolution
   B) II. Input resolution
   C) III. Table-based indexing
   D) I and II but not III
   E) I, II, and III

j) **What property of a planning system describes a representation method used to resolve threats to preconditions?**
   A) I. Regression
   B) II. Partial-order
   C) III. Refutation
   D) I and II but not III
   E) I, II, and III

3. Search (3 parts, 40 points total)

a) (10 points) Data structures for graph search. Write down the adjacency list and adjacency matrix for the following graph.



Adjacency list:
0: [(1, 3), (2, 2)]
1: [(3, 3), (4, 1)]
2: [(5, 7)]
3: [(5, 5)]
4: [(5, 4)]
5: []

Adjacency matrix:
* 3 2 * * *
* * * 3 1 *
* * * * * 7
* * * * * 5
* * * * * 4
* * * * * *

b) (20 points) Uninformed and Heuristic Search. Simulate the behavior of Greedy Best-First, Hill-Climbing, A/A*, and Branch-and-Bound search for the above graph with start node 0 and goal node 5. Show the evolution of the OPEN and CLOSED lists. *Break ties in ascending order of node number (lower-numbered nodes are expanded first in case of a tie).*

(For simplicity, the sort key – cost – is not shown in OPEN and CLOSED lists.)

Greedy Best-First search:

1. *OPEN: [0]*
   CLOSED: []
2. *OPEN: [1, 2]*
   CLOSED: [0]
3. *OPEN: [2, 3, 4]*
   CLOSED: [0, 1]
4. *OPEN: [5, 3, 4]*
   CLOSED: [0, 1, 2]
5. *OPEN: [3, 4]*
   CLOSED: [0, 1, 2, 5]

Returned path: [0, 2, 5], cost = 9

Hill-Climbing search:

1. *OPEN: [0]*
   CLOSED: []
2. *OPEN: [1] – omit backtracking info for simplicity*
   CLOSED: [0]
   (credit given if you stopped here and explained that the R&N algorithm without pathmax would not visit the children of node 1)
3. *OPEN: [3, 4]*
   CLOSED: [0, 1]
4. *OPEN: [5]*
   CLOSED: [0, 1, 3]
5. *OPEN: []*
   CLOSED: [0, 1, 3, 5]

Returned path: [0, 1, 3, 5], cost = 11

A/A* search:

1. *OPEN: [0], f* = ∞*
   CLOSED: []
2. *OPEN: [1, 2], f* = ∞*
   CLOSED: [0]
3. *OPEN: [2, 4, 3], f* = ∞*
   CLOSED: [0, 1]
4. *OPEN: [4, 3, 5], f* = ∞*
   CLOSED: [0, 1, 2]
5. *OPEN: [5, 3, 5], f* = ∞ – allow duplicates in OPEN, check CLOSED*
   CLOSED: [0, 1, 2, 4]
6. *OPEN: [3, 5], g* = 8 – don't have to check 3: f(3) = 6 + 3 = 9*
   CLOSED: [0, 1, 2, 4, 5]

Returned path: [0, 1, 4, 5], cost = 8 (optimal)

Branch-and-Bound search:

1. *OPEN: [0], g* = ∞*
   CLOSED: []
2. *OPEN: [2, 1], g* = ∞*
   CLOSED: [0]
3. *OPEN: [1, 5], g* = ∞*
   CLOSED: [0, 2]
4. *OPEN: [4, 3, 5], g* = ∞*
   CLOSED: [0, 2, 1]
5. *OPEN: [3, 5, 5'], g* = ∞*
   CLOSED: [0, 2, 1, 4]
6. *OPEN: [5, 5', 5''], g* = ∞*
   CLOSED: [0, 2, 1, 4, 3]
7. *OPEN: [], g* = 8*
   CLOSED: [0, 2, 1, 4, 3, 5]

Returned path: [0, 1, 4, 5], cost = 8 (optimal)

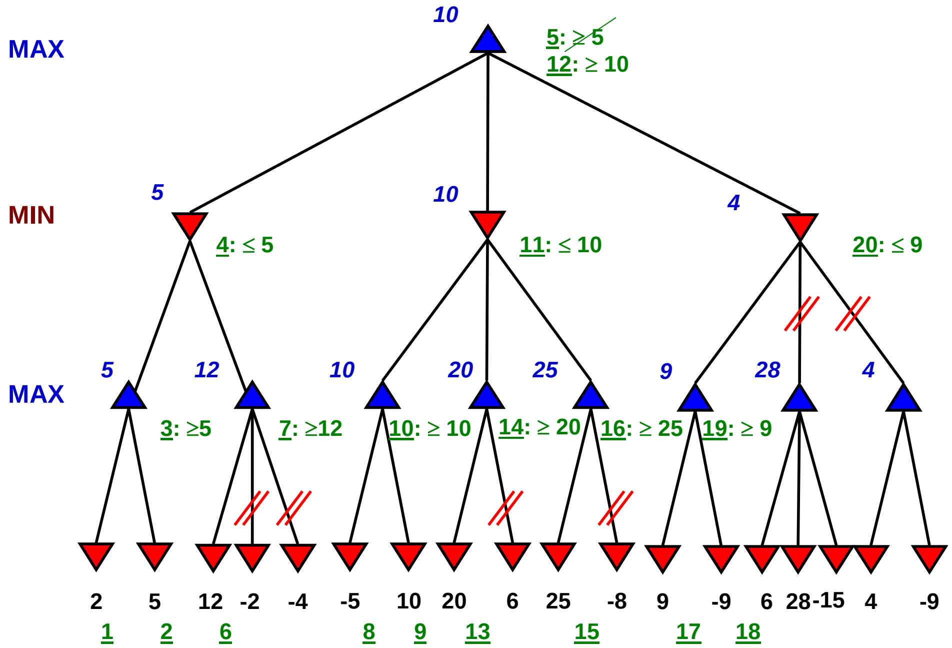
c) (10 points) Explain the relationship between "Algorithm A" and "Algorithm A*" (define any term you use to describe functions computed by each) and the relationship among A/A*, Hill-Climbing, and Greedy search.

Algorithm A refers to Algorithm A with an admissible heuristic h, such that for all nodes n, $h(n) \leq h^*(n)$, the true minimum cost to a goal node.*

Hill-climbing is a heuristic search that sorts only the children of the current head of the OPEN list by h(n) and prepends them to the front.
Greedy best-first search is a heuristic search that sorts the entire OPEN list by h(n).
A is a heuristic search that sorts the entire OPEN list by $f(n) = g(n) + h(n)$.*

4. Game Trees and Utilties (2 parts, 20 points each).

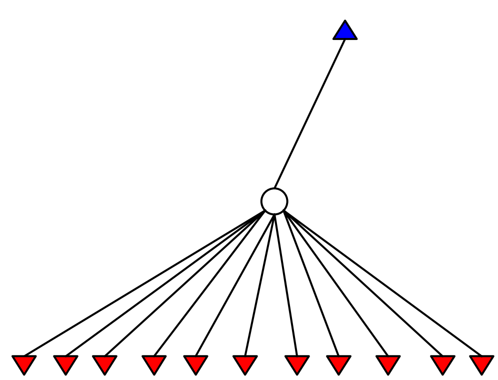
a) (20 points) Minimax Game Tree Search and Alpha-Beta Pruning. Consider this game tree:



i. (10 points) Simulate the behavior of the ***minimax algorithm*** on the above game tree.

ii. (10 points) Simulate the behavior of minimax with **alpha-beta (α-β) pruning** on the above game tree. *Show your work* (mark the ~~pruned branches~~ and number the static evaluations and all α and β inequality updates, in order).

b) (20 points) Expectiminimax. Draw an example of a portion of an *expectiminimax* game tree where the *chance nodes* correspond to the toss of two dice and the branches to the sum. What effect does this have on the expectation?



The branches of the chance node are (left to right) {2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}. The weights on these branches are {1/36, 2/36, 3/36, 4/36, 5/36, 6/36, 5/36, 4/36, 3/36, 2/36, 1/36}, which weights the expectation of each MIN node accordingly in calculating expected utility.

5. Logic and Resolution (3 parts, 40 points total).

a) Clausal form (10 points) Write down the steps for converting an arbitrary first-order logic (FOL) sentence into clausal form, and apply them to the following sentence:

$\forall$ x, y . P(x, y) $\Rightarrow$ $\exists$ z . Q(x, z) $\wedge$ ¬R(y, z)

I)*mplications out:* $\forall$ x, y . ¬P(x, y) $\vee$ $\exists$ z . Q(x, z) $\wedge$ ¬R(y, z)
N)*egations inward:* $\forall$ x, y . ¬P(x, y) $\vee$ $\exists$ z . Q(x, z) $\wedge$ ¬R(y, z) – *not needed*
S)*tandardize variables apart*: $\forall$ x, y . ¬P(x, y) $\vee$ $\exists$ z . Q(x, z) $\wedge$ ¬R(y, z) – *not needed*
E)*xistentials out (Skolemize)*: $\forall$ x, y . ¬P(x, y) $\vee$ Q(x, f(x,y)) $\wedge$ ¬R(y, f(x,y))
U)*niversals implicit*: ¬P(x, y) $\vee$ Q(x, f(x,y)) $\wedge$ ¬R(y, f(x,y))
D)*istributive law (conjunctive normal form):*
(¬P(x, y) $\vee$ Q(x, f(x,y))) $\wedge$ (¬P(x, y) $\vee$ ¬R(y, f(x,y)))
O)*perators out*:
{¬P(x, y), Q(x, f(x,y))}
{¬P(x, y), ¬R(y, f(x,y))}
R)*ename variables*:
{¬P(x_1, y_1), Q(x_1, f(x_1,y_1))}
{¬P(x_2, y_2), ¬R(y_2, f(x_2,y_2))}

b) Sentences in FOL (15 points) Write the following sentences in FOL.

i. For every action, there is an equal and opposite reaction.

$\forall$ x . *Action*(x) $\Rightarrow$
$\exists$ y . *Reaction*(y, x) $\wedge$
Equal (*magnitude*(x), *magnitude*(y)) $\wedge$
Opposite (*direction*(x), *direction*(y))

Notes

- *Equal-Magnitude* (x, y) $\wedge$ *Opposite-Direction* (x,y) was acceptable, but not *Equal* (x, y) $\wedge$ *Opposite* (x,y), even though these are syntactically equivalent.
- Even *Equal-Magnitude-Opposite-Direction* (x,y) was acceptable, but not x = y $\wedge$ *OppositeDirection* (x,y) or similar statements.

ii. Everyone loves someone who loves everyone.

$\forall$ x . $\forall$ y . *Loves*(x, y) $\Rightarrow$ $\forall$ z . *Loves*(z, x)

Note: $\forall$ x . $\exists$ y . *Loves*(x, y) $\wedge$ $\forall$ z . *Loves*(y, z), a very different interpretation, was also acceptable.

iii. ♬ We all live in a yellow submarine ♬

$\exists$ x . *Submarine*(x) $\wedge$ *Yellow*(x) $\wedge$ $\forall$ y . *Among-Us* (y) $\Rightarrow$ *Live-In* (y, x)

Notes: This is to be distinguished from
$\forall$ y . *Among-Us* (y) $\Rightarrow$ $\exists$ x . *Submarine*(x) $\wedge$ *Yellow*(x) $\wedge$ *Live-In* (y, x)
Also, *Among-Us* was optional.

c) Proof (15 points). Use *first-order refutation resolution* to prove the following theorem given the specified knowledge base (KB). Write down the *unifiers* for each application of the generalized resolution rule. You may use conjunctive or implicative normal form, whichever is more convenient for you.

- **Knowledge base:**
  - For every married couple, there is some habit of the husband's that the wife does not like. (*Hint*: introduce only the predicates needed to complete this proof!)
  - Joe is Susan's husband.
- **Theorem**: Susan does not like all of Joe's habits.

Note that the rule (first axiom) is the same as in 5(a). With alpha renaming, it is:

∀ husband, wife . *Husband-Of*(husband, wife) ⇒
∃ habit . *Has* (husband, habit) ∧ ¬*Likes*(wife, habit)

If you put this in clausal form in (a), you have it already:
C1. {¬*Husband-Of*($husband_1$, $wife_1$), *Has*($husband_1$, habit-of($husband_1$, $wife_1$))}
C2. {¬*Husband-Of*($husband_2$, $wife_2$), ¬*Likes*($wife_2$, habit-of($husband_2$, $wife_2$))}

The other axiom is:
C3. {*Husband-Of*(Joe, Susan)}

The negated query is:
¬∃ bad-habit . *Has* (Joe, bad-habit) ∧ ¬*Likes*(Susan, bad-habit)
whose clausal form is:
C4. {¬*Has* (Joe, bad-habit), *Likes*(Susan, bad-habit)}

There are 4 unit resolutions (though you could complete the proof in fewer using implicative normal form), and the substitutions are obviously all of the form {$husband_i$ / Joe}, {$wife_j$ / Susan}, {bad-habit / habit-of ($husband_i$, $wife_i$)}. I give the answers in input resolution form:

C5. C1 with C3 – resolvent *Husband-Of,* result *Has*
C6. C4 with C5 – resolvent ¬*Has,* result *Likes*
C7. C2 with C6 – resolvent ¬*Likes*, result ¬*Husband-Of*
C8: C3 with C7 – resolvent ¬*Husband-Of*, result NIL

Extra Credit (10 Points) Give an example of a *Sussman anomaly* in the Blocks World. Write down the operator axioms, the initial state and goal state descriptors, and give a sketch of how a *partial-order planning* (*POP*) system would solve the problem.

Initial State:
On(B, A)
On(A, Table)
On(C, Table)

Goal State:
On(B, A)
On(C, B)
On(A, Table)

Operators:
MOVE (x, y, z): $\forall x . \forall y . On(x, y, s) \Rightarrow$
$\neg On(x, y, Result(Move, x, y, z)) \wedge On(x, z, Result(Move, x, y, z))$

A POP system must be able to order actions so that C is unstacked from A to make way for B, detecting the threat (interference) that is not eliminated by the MOVE (C, Table, B).