CS 188 Spring 2007 — Introduction to Artificial Intelligence — Midterm Solutions

You have 80 minutes. There are five questions with equal points. Look at all the five questions and answer those that you know best first. At the end of the exam there is a question for extra credit worth 10 points. Please attempt this *only if* you have time. Don't panic!

Mark your answers ON THE EXAM ITSELF. Write your name, SID, Login, and Section number on the top of each page.

For true/false, circle the *True* or *False*

If you are unsure of an answer, provide a brief explanation. All questions can be successfully answered with at most a few sentences.

For official use only

Q. 1	Q. 2	Q. 3	Q. 4	Q. 5	Total	EC
/20	/20	/20	/20	/20	/100	/10

1. (20 points.) Search

(a) **True/False (8 points)**. Each problem is worth 2 points. Incorrect answers are worth 0 points. Skipped questions are worth 1 point.

i. *True/False*. Uniform Cost Search (UCS) is an optimal uninformed search technique both for tree search and for graph search (assume positive step costs and a finite branching factor).
True. Uniform Cost Search will give an optimal solution in both cases, albeit slower than A* with a good heuristic.

ii. *True/False*. Iterative deepening search has linear space requirements ($O(bd)$ where b is the branching factor and d is the depth of the shallowest solution) for both tree search and graph search.
True. IDS needs to store $O(b)$ nodes for each level it explores. If the shallowest solution is at depth d then $O(bd)$ is required.

iii. *True/False*. If $h_1(n)$ and $h_2(n)$ are admissible and consistent heuristics then $h_3(n) = \frac{h_1(n)+h_2(n)}{2}$ is admissible and consistent.
True. You can see this by averaging the inequality conditions for admissible and consistent.

iv. *True/False*. Local beam search with a beam size k is equivalent to a parallel local search with k random restarts.
False. Local beam search with a beam size k will use k best locations to explore further and then again keep the best k locations. Random restart will start from scratch from k different locations.

(b) **Search Strategies (12 points)**

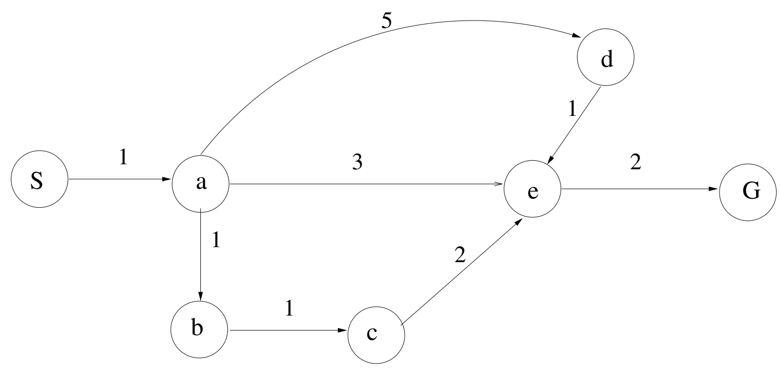


Figure 1. A Search graph with Start State S, goal state G and integer costs on the arcs.

i. Return a possible order of node expansion with the following search algorithms for the search graph in Figure 1. (3 points)

A. Depth First Search (1 point)
Many solutions. Eg. S a b c e G

B. Breadth First Search (1 point)
Many solutions. Eg. S a b d e c G. Common mistakes: expanding c before e

C. Uniform Cost Search (1 point)
S a b c e G or S a b c e d G

ii. Consider three heuristics h_1, h_2, h_3. The table below indicates the estimated cost to goal (h value) for each of the heuristics for each node in the search graph.

node	h_1	h_2	h_3
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

- Which of the three heuristics (h_1, h_2, h_3) are

  A. admissible (2 points)

  B. consistent (2 points)

  h_1, and h_2 are admissible.
  h_1 is also consistent.

- Show the node expansion order using A* for any of the heuristics that is consistent and admissible (4 points).
  For h_1: S a e G or S a e b G
  If indicates that h_2 or h_3 are consistent and use it here, will get full credit too.
  If the answer is almost correct, you will get 2 points.

- Show the node expansion order using greedy best first search for the heuristic h_2 (1 point).
  S a d e G or S a e G

2. (20 points.) CSP

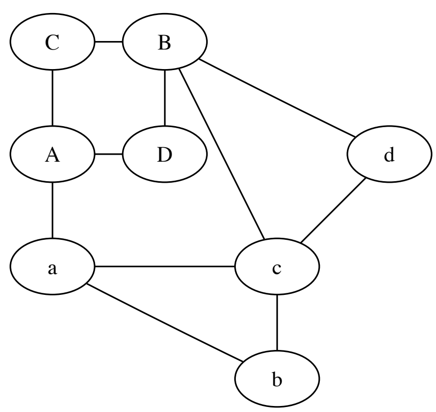
On the day of an important vote, four senators show up in the Senate: two demublicans (A and B) and two repocrats (C and D). Each chooses whether to sit on the east or west side, and whether to vote "Yes," "No," or "Waffle." We know the following facts:

- 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 (it is scary over there).

We would like to determine where everyone sits and how they vote.

(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. Please use capital letters A, B, C, D for variables whose domains are vote assignments for senators A, B, C, and D respectively and lowercase letters a, b, c, d for variables whose domains are seat assignments for senators A, B, C, and D respectively.

Solution:



Partial credit was given for nearly correct graphs based on the number of mistakes.

(b) (7 points) On 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: $\{\{A = Yes, a = West\}; \{A = No, a = East\}\}$).

Solution:
d to B: $\{\{d = East, B = Yes\}; \{d = West, b = Yes\}; \{d = West, b = No\}\}$
d to c: $\{\{d = East, c = East\}; \{d = West, c = East\}; \{d = West, c = West\}\}$

Grading: one point was taken off for each missing or incorrect option. Solutions which were wrong due to mistakes in (a) were given some extra partial credit.

(c) (6 points) Give a consistent assignment for the entire CSP.

Solution: Everyone sits on the west. A and B vote yes while C and D vote no.

Grading: if the assignment was almost right, partial credit was awarded based on how many constraints were satisfied.

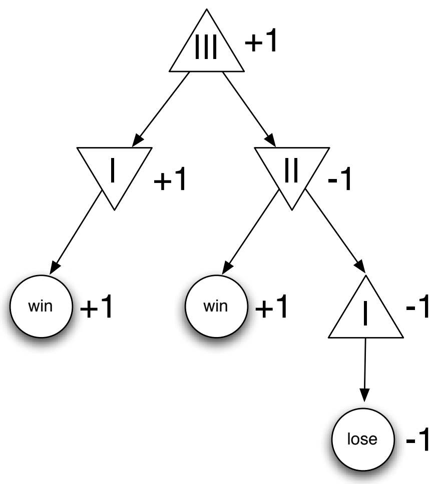
3. (20 points.) Games

(a) (Game trees and minimax (8 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, using upwards-pointing triangles for MAX-nodes and downwards-pointing triangles for MIN-nodes (assuming the first player is MAX). Next to each node, write the corresponding utility for MAX, assuming optimal play by both players.

Solution: pictured at right

Point Breakdown:

- Graph structure (3 pts)
- Terminal utilities (2 pts)
- Minimax (internal node) utilities (3 pts)



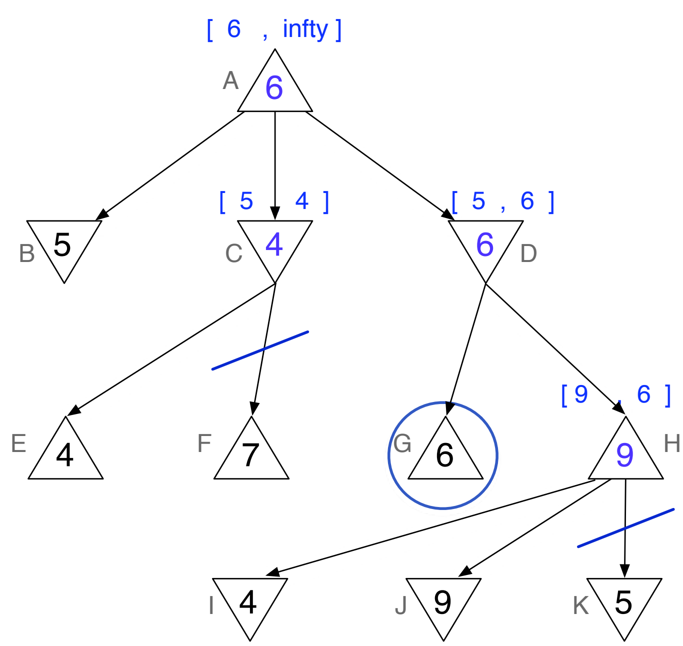
(b) (α-β pruning) (12 points). The following question refers to the figure below. Assuming nodes are evaluated in left-to-right order, annotate each interior node with the utility (for MAX) it would be assigned by minimax with alpha-beta pruning, as well as its corresponding final values of $[\alpha, \beta]$. Again MAX plays first and upwards pointing triangles are MAX nodes and downwards pointing nodes are MIN nodes. Put a slash through links corresponding to pruned subtrees (if any), and circle the leaf that would be reached by optimal play. Assume that the initial values for α and β at the root are $[-\infty, \infty]$.

Solution: pictured at right

Point Breakdown:

- Minimax vals, optimal leaf (4 pts)
- $[\alpha.\beta]$ vals, pruned subtrees (8 pts)

The emphasis in grading this part was on correctly identifying the pruned subtrees. If this was done correctly, the precise values of α and β were not as important.



4. (20 points.) Logic

(a) (Propositional Satisfiability (6 points)). For each of the sentences below, indicate whether it is satisfiable, unsatisfiable, or valid.

i. $P \Rightarrow P$

Solution: valid

ii. $P \Rightarrow \neg P$

Solution: satisfiable but not valid

iii. $P \Leftrightarrow \neg P$

Solution: unsatisfiable

iv. $(P \Rightarrow Q) \Rightarrow ((P \wedge R) \Rightarrow Q)$

Solution: valid

(b) (Propositional Resolution (8 points))

Consider the following future scenario, where vaccum cleaner agents exhibit affective states of happiness or sadness. Here is a typical agent, Jan.

- If Jan is humming, then Jan is happy.
- If there is no dirt in the house then Jan is happy.
- Jan is not happy.

Given the following, use propositional resolution to prove that there is dirt in the house and Jan is not humming.

First create the propositions and rules from the sentences above (2.5 points), convert them to CNF (2.5 points) and then use resolution to prove the assertion that there is dirt in the house and Jan is not humming (3 points).

Solution:

Here are the propositions expressed by the English sentences:

$$\begin{aligned} Hum &\Rightarrow Happy \\ NoDirt &\Rightarrow Happy \\ &\neg Happy \end{aligned}$$

The corresponding CNF sentence is

$$(\neg Hum \vee Happy) \wedge (\neg NoDirt \vee Happy) \wedge \neg Happy$$

The first and third clauses may be resolved to yield $\neg Hum$, and the second and third may be resolved to yield $\neg NoDirt$.

Although we were looking for a proof by resolution, credit was given for most reasonable proofs.

(c) (Translation from Natural Language to FOL (6 points)).

Use the following vocabulary to express the assertions in the sentences to follow.

- Male(x) means that x is male.
- Female (x) means that x is female.
- Loves (x, y) means that x loves y.
- Married (x, y) means that x is married to y.
- Respect (x, y) means that x respects y.

i. If Sam loves everybody then Sam loves himself.
$\forall x[Love(Sam, x)] \Rightarrow Love(Sam, Sam)$
1 point for a reasonable attempt.

ii. All women love the man that they are married to.
$\forall x \forall y[Female(x) \wedge Male(y) \wedge Married(x, y) \Rightarrow Loves(x, y)]$
1.5 points for using $\exists y$.
1 point for a reasonable attempt.

iii. No woman loves a man who does not respect all women.
$\neg\exists x \exists y \exists z[Female(x) \wedge Male(y) \wedge Female(z) \wedge Loves(x, y) \wedge \neg Respect(y, z)]$
1.5 points for almost correct solutions.
1 point for a reasonable attempt.

5. (20 points.) Probability and Bayes' Nets

- Probability Basics (7 points)

(a) (4 points) Suppose we wish to calculate $P(x|y,z)$, and have no information about any independence relationships between the variables X, Y, and Z. Identify all of the following sets of values that are *sufficient* for the calculation.

i. $P(Y,Z), P(X), P(Y|X), P(Z|X)$

ii. $P(Y,Z), P(X), P(Y,Z|X)$

iii. $P(X), P(Y|X), P(Z|X)$

*Solution:*only ii is sufficient. Without any independence assumptions, we need at least one term containing all 3 variables to compute the desired result. By Bayes Rule,

$$P(x|y,z) = \frac{P(y,z|x) * P(x)}{P(y,z)}$$

Grading: 0, 1, 2, or 3 correct yielded 0, 1, 2, or 4 points (respectively).

(b) (3 points) Now suppose we know that $P(Y|X,Z) = P(Y|X)$ for all values of X, Y, and Z. Now indicate all of the sets above which are sufficient for performing the calculation.

Solution: i, ii, and iii are all sufficient. By Bayes Rule and Chain Rule,

$$P(x|y,z) = \frac{P(y,z|x) * P(x)}{P(y,z)} = \frac{P(x) * P(z|x) * P(y|x,z)}{\sum_x P(x) * P(z|x) * P(y|x,z)} = \frac{P(x) * P(z|x) * P(y|x)}{\sum_x P(x) * P(z|x) * P(y|x)}$$

Grading: +1 for each correctly classified set.

- Bayes Nets (13 points)

(a) (6 points)

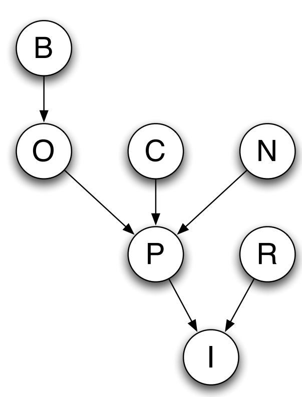
Construct a Bayes net that corresponds to the following scenario for cell phone use:

- Having a charged battery (B) is required for a cell phone to be operational (O).
- Having an operational cell phone, being in the coverage area (C), and knowing the number to call (N) are all required for the placing of a call (P).
- Placing a call and the recipient being near the phone (R) are required to initiate a conversation (I).
- Assume there are no other dependencies.

Your Bayes net needs to show your model of the structure of the scenario information, no specific conditional probability values need be represented.

Solution: pictured at right

Grading: -1 for each incorrect arc.
All arcs backwards or no arrows: 2-3 / 6 points.



(b) (3 points)
Which of the following independence properties are true of your network

- $B \perp\!\!\!\perp I$
  no
- $N \perp\!\!\!\perp C$
  yes
- $O \perp\!\!\!\perp C|B$
  yes
- $B \perp\!\!\!\perp R|I$
  no
- $B \perp\!\!\!\perp C|I$
  no
- $B \perp\!\!\!\perp C|I, R, O$
  yes

Grading: 0.5 points per question. blanks were counted as "no"s.

(c) (4 points)
Give an expression in terms of conditional probabilities in the network (you don't need any numbers, just the expression) to compute the following quantities.

- $P(b, o, c, n, p, r, i)$
- $P(o, c, i)$

Solution:

$$\begin{aligned} P(b, o, c, n, p, r, i) &= P(b)P(c)P(n)P(r)P(o|b)P(p|o, c, n)P(i|p, r) \\ P(o, c, i) &= \sum_{b \in \{T,F\}} \sum_{n \in \{T,F\}} \sum_{p \in \{T,F\}} \sum_{r \in \{T,F\}} P(b)P(c)P(n)P(r)P(o|b)P(p|o, c, n)P(i|p, r) \end{aligned}$$

6. (10 points.) Extra Credit Questions

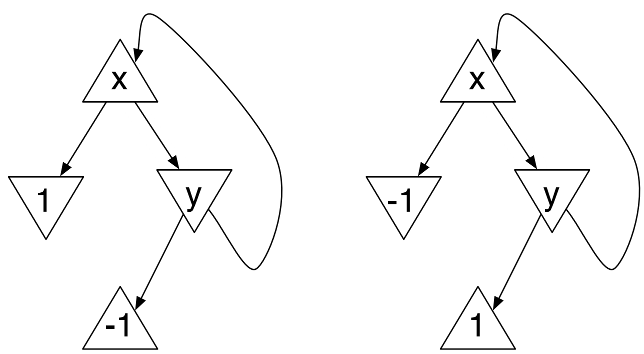
(a) (Heuristic Search (4 Points)).

The heuristic function h used by the A* algorithm has the same *fixed absolute error* at all the nodes if for all nodes n, $h^*(n) - h(n) = c$ where c is some positive constant. Here $h^*(n)$ is the cost of the lowest cost path from n to a goal node, so the cost of the lowest cost path from the start node through n to a goal node is $f^*(n) = g(n) + h^*(n)$. Suppose A* uses h with the property stated above. What can you say about the nodes that A* visits. Explain your answer.

Solution: With such an h, A* will only investigate nodes on some optimal solution path (e.g., the same as if it was searching with h^*). Shifting the heuristic by a constant doesn't affect the behavior of the algorithm, because all f values are shifted by the same amount and only the relative *ordering* of f values determines the order of node expansion.

If there is only a single solution path, or the A* algorithm is modified to break ties by choosing the deepest node first, A* will expand a path directly leading from the start state to the goal, solving the problem in linear time.

(b) (Minimax in Loopy Games (6 Points)).



(a) (b)

Some games (e.g., those where players are allowed to pass) have loops in their game graphs. In this problem, you will apply what you know about minimax to analyze the two simple loopy game graphs provided above. Again, upwards-pointing triangles are MAX nodes, downwards-pointing triangles are MIN nodes, and all utilities are for MAX. What, if anything, can you conclude about the utilities x and y in these graphs? For each graph, if there is only one possible pair of values, specify it; otherwise, write down as many constraints (i.e. inequalities) on the values of x and y as you can.

Solution:

(a) $x = 1$, $y = -1$. Here the loop does not matter, because neither x nor y would have incentive to choose the branch leading to a loop (since it can only decrease their eventual utilities)

(a) $-1 \leq x, y \leq 1$, $x \geq y$. Here the loop does matter, because both x and y may hope for a better utility by choosing the loopy option. Thus, there is no single well-defined utility for x and y (unless the game specifies that loops have a particular value, such as 0). However, using the minimax principle, we can derive the preceding constraints on the possible values of x and y.