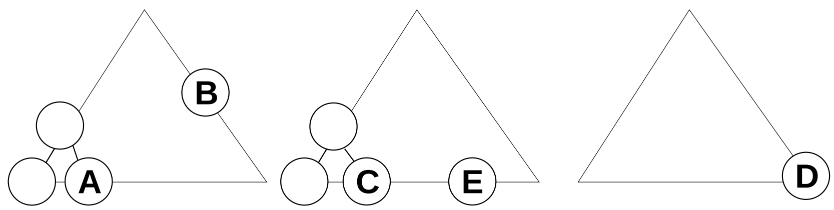
CPSC 625-600 Homework #1 Solution
October 8, 2007

Yoonsuck Choe

October 9, 2007

1 Uninformed Search



(a) Tree 1 (b) Tree 2 (c) Tree 3

Figure 1: **Search Trees.**

Consider the three search trees in Figure 1. Suppose the branching factor is b and the tree is full. Suppose branches are visited from the left to the right. Nodes A, B, C, D, and E the *goal nodes* in the trees. Assume that nodes A, C, D, and E are at depth n; and A and C are the k-th node of their parents (i.e., they are children of the left-most node at depth $n-1$). Node B is at depth m ($< n$). Node D is the last node to the right at depth n. Depth n is the last level of all the trees. Further assume that the exploration of each depth level proceeds from the left to the right.

Question 1 (4 pts): If $n > 2b$, which one shows a case where both depth-first and breadth-first have identical **time** complexity? (Tree 1, 2, 3, none, or any combination of the three)

> Tree 1: DFS will visit $n + k - 1$ nodes to reach node A. BFS will visit $1 + b + b^2 + ... + b^m$ nodes to reach node B. If $n + k - 1 = 1 + b + b^2 + ... + b^m$, the answer is the time complexity of DFS=BFS. In most cases, it will not be the case. Tree 3: DFS and BFS have the same time complexity.

Question 2 (4 pts): Which one shows a case where depth-first can be complete but non-optimal? (Tree 1, 2, 3, none, or any combination of the three) Explain why.

> Tree 1 is the only answer, becasue DFS finds (A) at depth n (thus it is complete, since it found a goal), while there is a goal (B) at depth $m < n$, thus it is suboptimal.

Question 3 (4 pts): Assume $b = 5$, $k = 2$, and $n = 7$. What is the number of nodes visited in case of Tree 2 for depth-first (and breadth-first)? Note that the "number of node visited" is defined as the number' of goal checks.

DFS $= 7 - 1 + 2 = 6 + 2 = 8$. BFS $= 1 + 5 + 5^2 + ... + 5^6 + 2 = 19533$.

Question 4 (4 pts): Assume $b = 10$, $k = 3$, $m = 4$, and $n = 20$. In which case does depth-first outperform breadth-first in terms of time complexity (= nodes visited)? (Tree 1, 2, 3, none, or any combination of the three)

Tree 1 and Tree 2.

- Tree 1: DFS $= 20 - 1 + 3 <$ BFS $= 1 + 10 + ... + 10^3 + 3$.
- Tree 2: DFS $= 20 - 1 + 3 <$ BFS $= 1 + 10 + 10^2 + ... + 10^6 + 3$.

2 Informed Search

Answer the following questions regarding informed search strategies.

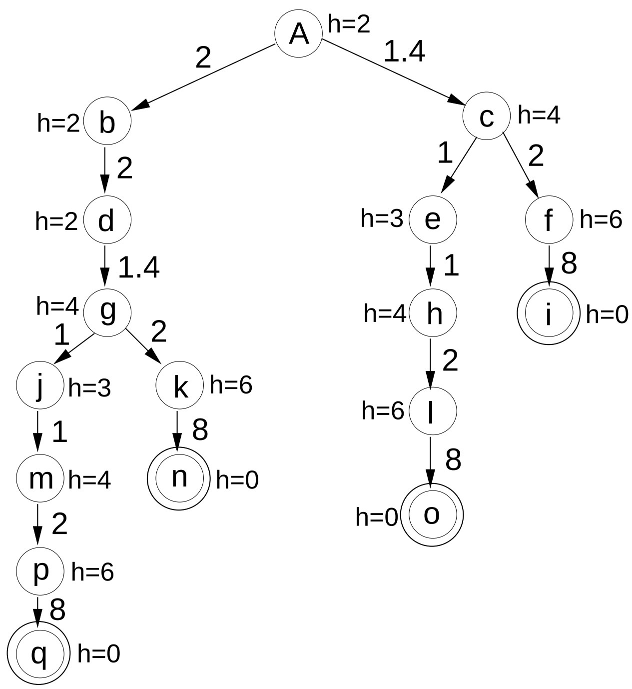


Figure 2: **Search tree.** An example search tree is shown with path cost on each edge and heuristic function value next to each node. The leaf nodes are goal states.

Question 1 (6 pts): Is the heuristic admissible? Explain why.

Yes, because at each node n, $h(n)$ is less than the actual cost from n to the nearest goal (i.e. the sum of path cost from n to any of the closest goals).

Question 2 (6 pts): Given the search tree in figure 2, calculate the $f(n)$ value for each node (A to q).

$f(A) = 2$, $f(b) = 2 + 2 = 4$, $f(c) = 4 + 1.4 = 5.4$, $f(d) = 2 + 2 + 2 = 6$, $f(e) = 3 + 1.4 + 1 = 5.4$, $f(f) = 6 + 1.4 + 2 = 9.4$, $f(g) = 4 + 2 + 2 + 1.4 = 9.4$,

$f(h) = 4+1.4+1+1 = 7.4$, $f(i) = 0+1.4+2+8 = 11.4$, $f(j) = 3+2+2+1.4+1 = 9.4$ $f(k) = 6+2+2+1.4+2 = 13.4$, $f(l) = 6+1.4+1+1+2 = 11.4$, $f(m) = 4+2+2+1.4+1+1 = 11.4$, $f(n) = 0+2+2+1.4+2+8 = 15.4$, $f(o) = 0+1.4+1+1+2+8 = 13.4$, $f(p) = 6+2+2+1.4+1+1+2 = 15.4$, $f(q) = 0+2+2+1.4+1+1+2+8 = 17.4$.

Question 3 (12 pts): Given the search tree in figure 2, list the node **visit order** and the **goal state** reached for the two search methods: (1) greedy search and (2) A*. When the evaluation function values are tied while sorting, give preference to the one that has been added latest. That is, given two nodes n and m, if $f(n) = f(m)$, then visit n if n is newer (i.e., m has been in the node list for a longer while than n).

- Greedy search: uses $h(n)$ and best-first.
  $A \rightarrow b \rightarrow d \rightarrow g \rightarrow j \rightarrow m \rightarrow c \rightarrow e \rightarrow h \rightarrow i \rightarrow o$
  The total cost to the goal is 13.4.
- A*: use $f(n)$, and best-first. $A \rightarrow b \rightarrow c \rightarrow e \rightarrow d \rightarrow h \rightarrow g \rightarrow j \rightarrow f \rightarrow i$
  The total cost to the goal is 11.4, thus optimal.

3 Game Playing (30 points)

3.1 Minmax Search

Question 1 (5 pts): Using the following figure 3, use minmax search to assign utility values for each internal node (i.e., non-leaf node) and indicate which path is the optimal solution for the MAX node at the root of the tree.

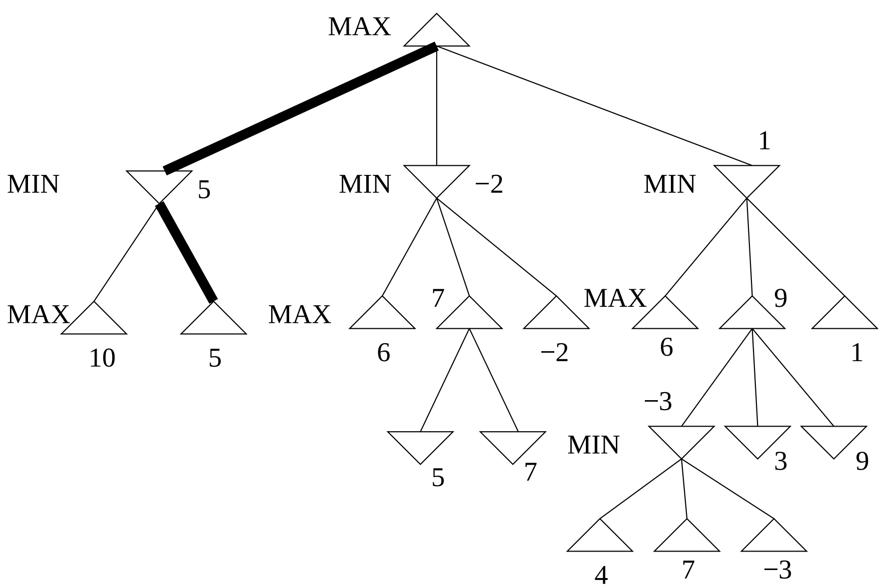


Figure 3: **Game Tree.** Solve using minmax search.

See figure 3.

3.2 $\alpha - \beta$ pruning

Question 1 (5 pts): Using the following figure 4, for each node, indicate the final utility value.

Question 2 (9 pts): For each cut that happens, draw a line to cross out that subtree.

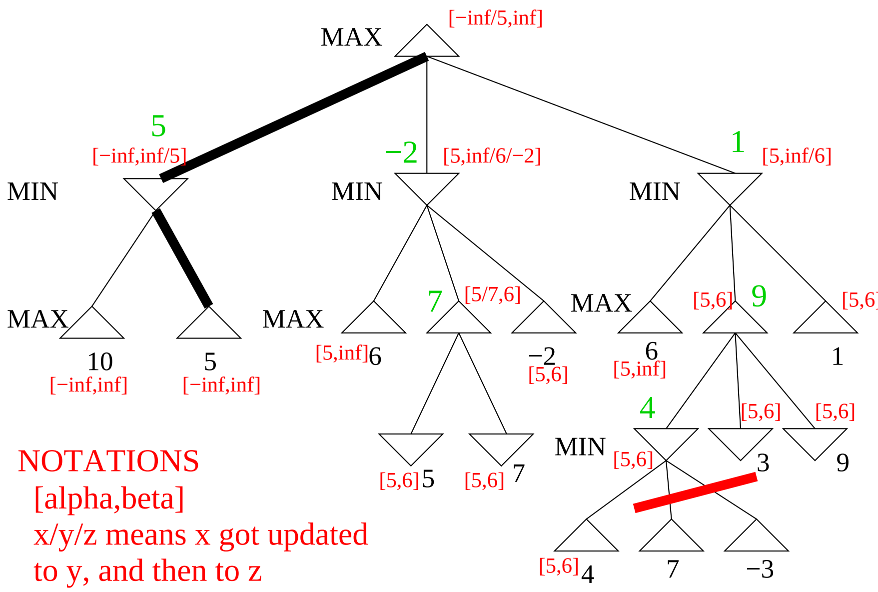


Figure 4: **Game Tree.** Solve using $\alpha - \beta$ pruning. This tree is the same as figure 3.

See figure 4.

4 Propositional Logic

4.1 Normal forms

In all of the problems in this section, show each step of the derivation and indicate which axioms (or other rules) you used: For example, *distributive law, definition, etc.*

Question 1 (4 pts): Convert $\neg(P \rightarrow \neg S) \vee (\neg(Q \rightarrow R) \wedge S)$ into conjunctive normal form.

1. $\neg(\neg P \vee \neg S) \vee (\neg(\neg Q \vee R) \wedge S)$: remove implication
2. $(P \wedge S) \vee ((Q \wedge \neg R) \wedge S)$: De Morgan
3. $(P \vee (Q \wedge \neg R)) \wedge S$: Distributive (in reverse)
4. $((P \vee Q) \wedge (P \vee \neg R)) \wedge S$: Distributive
5. $(P \vee Q) \wedge (P \vee \neg R) \wedge S$: Associative

Question 2 (4 pts): Convert $\neg T \rightarrow (R \wedge (S \rightarrow (P \wedge Q)))$ into disjunctive normal form.

1. $T \vee (R \wedge (\neg S \vee (P \wedge Q)))$: remove implication
2. $T \vee (R \wedge \neg S) \vee (R \wedge P \wedge Q)$: distributive, and associative

Question 3 (4 pts): Convert $(R \wedge S) \rightarrow (Q \rightarrow \neg(P \wedge \neg T))$ into horn normal form. After that, show the equivalent expression with a single implication ($\rightarrow$) and some conjunctions ($\wedge$).

1. $\neg(R \wedge S) \vee (\neg Q \vee \neg(P \wedge \neg T))$: remove implication
2. $(\neg R \vee \neg S) \vee (\neg Q \vee (\neg P \vee T))$: De Morgan
3. $\neg R \vee \neg S \vee \neg Q \vee \neg P \vee T$: Associative
4. $(R \wedge S \wedge Q \wedge P) \rightarrow T$: implication

4.2 Valid vs. inconsistent

Question 1 (5 pts): Show that $(P \wedge Q) \vee \neg(R \wedge \neg(P \rightarrow \neg Q))$ is valid.

1. $(P \wedge Q) \vee (\neg R \vee (\neg P \vee \neg Q))$: implication and De Morgan
2. $(P \wedge Q) \vee (\neg R \vee \neg(P \wedge Q))$: De Morgan
3. $(P \wedge Q) \vee \neg R \vee \neg(P \wedge Q)$: Associative
4. $((P \wedge Q) \vee \neg(P \wedge Q)) \vee \neg R$: Associative
5. $True \vee \neg R = True$

Question 2 (5 pts): Show that $(P \vee Q) \wedge \neg(\neg Q \rightarrow P)$ is inconsistent.

1. $(P \vee Q) \wedge \neg(Q \vee P)$: implication
2. $(P \vee Q) \wedge \neg(P \vee Q)$: commutative
3. $False$

4.3 Theorem proving

Given:

1. $S \vee \neg P$
2. $\neg S \vee R$
3. $R \rightarrow T$
4. $(R \wedge S \wedge T) \rightarrow Q$

show that $P \rightarrow Q$ is a logical consequence of the above using **resolution**. Precisely follow the steps below.

Question 1 (5 pts): Convert the above problem into a form that is suitable for resolution. This may involve converting some expressions into CNF, and other steps such as including the conclusion part ($P \rightarrow Q$).

Negate conclusion: $\neg(\neg P \vee Q) = P \wedge \neg Q$

1. $S \vee \neg P$
2. $\neg S \vee R$
3. $\neg R \vee T$

4. $\neg R \vee \neg S \vee \neg T \vee Q$
5. P
6. $\neg Q$

Question 2 (14 pts): With the resulting resolution problem from the above, prove the initial theorem using resolution. Show every step.

7. 1,5: S
8. 2,7: R
9. 3,8: T
10. 4,6: $\neg R \vee \neg S \vee \neg T$
11. 7,10: $\neg R \vee \neg T$
12. 8,11: $\neg T$
13. 9,12: $False$