CSCE 625 Homework #2 (2023 Fall)

Game Playing ; Logic and Theorem Proving

Total 100 points

See Canvas for submission details.

1 Game Playing

1.1 Minmax Search

Question 1 (5 pts): Using the following figure 1, 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. Assume you explore the successors from left to right.

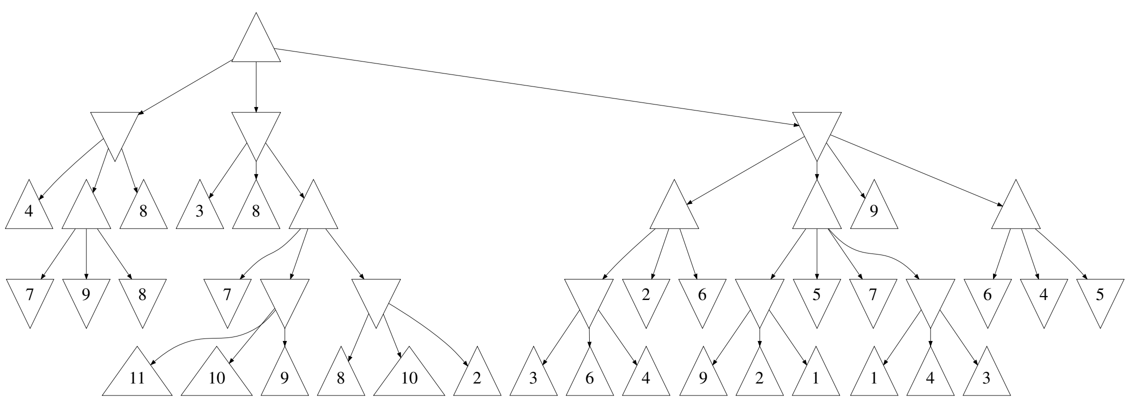


Figure 1: **Game Tree.** Solve using minmax search. Note: △ is the MAX node, ▽ is the MIN node.

1.2 $\alpha - \beta$ pruning

Question 2 (10 pts): Using the following figure 2, use $\alpha - \beta$ pruning to (1) 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. (2) For each node, indicate the final α and β values. (Note that initial values at the root are $\alpha = -\infty, \beta = \infty$.) (3) For each cut that happens, draw a line to cross out that subtree.

Hint: There are 6 places that need to be cut (see how to count cuts: figure 3).

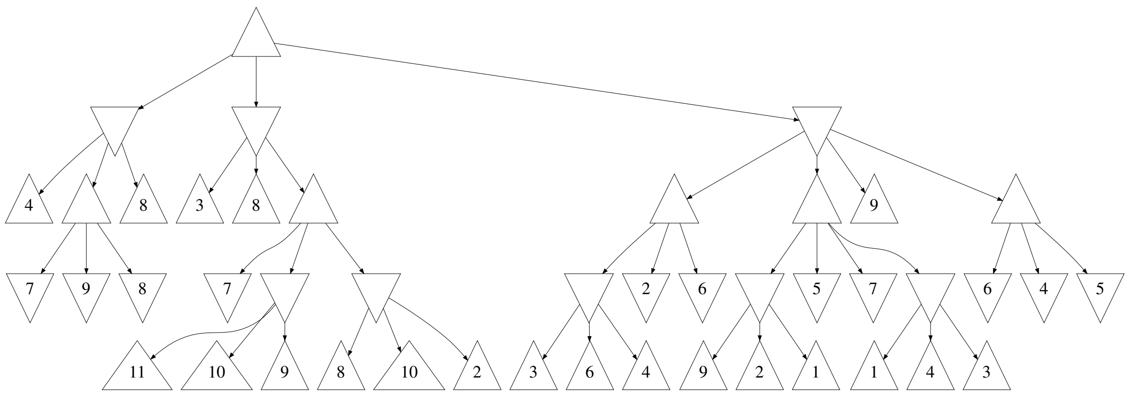


Figure 2: **Game Tree.** Solve using $\alpha - \beta$ pruning. This tree is the same as figure 1. Note: $\triangle$ is the MAX node, $\bigtriangledown$ is the MIN node.

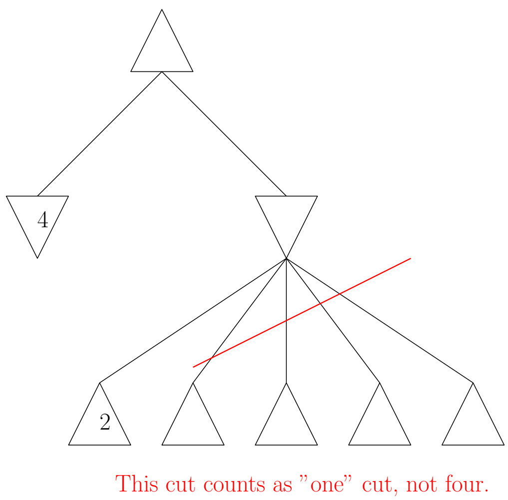


Figure 3: **Game Tree.** Counting cuts

Question 3 (5 pts): In Minmax search, we used a depth-first exploration through the use of recursion. We know that Minmax gives an optimal solution, however, we also know that depth-first search is suboptimal. Explain why Minmax gives an optimal solution even when it is using a depth-first exploration.

Question 4 (Progamming: 15 pts): Implement the alpha-beta pruning algorithm in Python (use colab), and test with the problem in Figure 1. An implementation of the minmax algorithm is provided as `minmax.ipynb`, to get you started. Use the nested list structure to represent the game tree, as shown in the minmax implementation.

Note that minmax and alpha-beta can be quite different.

2 Propositional Logic

In this section, assume P, Q, R, S, T, U, V are atoms (propositions).

2.1 Inference rule

Question 5 (5 pts): Using a truth table, show that the resolution inference rule is valid (if the premises are true, the conclusion is also true, or, $((P \vee Q) \wedge (\neg Q \vee R)) \rightarrow (P \vee R)$ is valid). Note: valid means "true under all interpretations".

$$\frac{P \vee Q, \;\; \neg Q \vee R}{P \vee R}$$

P	Q	R	$(P \vee Q)$	$(\neg Q \vee R)$	$(P \vee Q) \wedge (\neg Q \vee R)$	$(P \vee R)$	$((P \vee Q) \wedge (\neg Q \vee R)) \rightarrow (P \vee R)$
T	T	T					
T	T	F					
T	F	T					
T	F	F					
F	T	T					
F	T	F					
F	F	T					
F	F	F					

2.2 Normal forms

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

Question 6 (9 pts):

(1) Convert $P \vee \neg(\neg Q \vee R)$ into conjunctive normal form.

(2) Convert $P \vee (R \wedge \neg S) \vee (R \wedge Q)$ into conjunctive normal form.

(3) Convert $\neg((P \wedge Q) \wedge (R \rightarrow S))$ into disjunctive normal form.

2.3 Theorem proving

Using resolution, show that $S \vee T \vee \neg Q$ is a logical consequence of the following premises:

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

Question 7 (8 pts): Transform the above problem into a set of clauses (premises and the conclusion), suitable for resolution-based theorem proving.

- Turn each axiom in the list of premises above into conjunctive normal form.

- One premise may result in multiple clauses.
- For example, a premise $\neg((P \wedge \neg R) \vee S)$ will convert to CNF as $(\neg P \vee R) \wedge \neg S$, which results in two clauses:
  - Clause 1: $\neg P \vee R$
  - Clause 2: $\neg S$

- Don't forget to negate the conclusion $(\neg P \vee S)$, before adding to the clause list. Multiple clauses may (or may not) result from the negated conclusion.

Write your resulting clauses in the following format:

C1:

C2:

C3:

C4:

C5:

C6:

...

Question 8 (8 pts): Use resolution to derive **False**. Show every step. DO NOT USE any other infernce rule.

3 First Order Logic (Basics)

Important: In this section, assume that w, x, y, z are variables; A, B, C, D are constants; and $f(\cdot), g(\cdot), h(\cdot)$ are functions; and $P(\cdot), Q(\cdot), R(\cdot)$ are predicates.

Question 9 (6 pts): Convert to prenex normal form (2 points each):

1. $\forall x \ \neg(\exists y\, P(x, y) \wedge \neg Q(x))$
2. $\neg \exists x\, (P(x) \rightarrow \neg(\exists y\, Q(x, y)))$
3. $\neg \exists x\, (\neg(\forall y\, Q(x, y)) \vee \neg P(x))$

Question 10 (5 pts): Skolemize the expressions (1 point each):

1. $\exists x P(x)$
2. $\forall x \exists y P(x, y)$
3. $\exists x \exists y \forall z P(x, y) \wedge Q(y, z)$
4. $\forall x \exists y \exists z P(x, y, z) \wedge Q(y, z)$
5. $\forall x \forall y \exists z P(x, y) \wedge Q(x, y, z)$

Question 11 (9 pts): Convert the following into a standard form (prenex, CNF, skolemization: 3pt each):

$$\forall x\, [\neg(\exists z(\neg Q(z) \wedge R(x,z))) \rightarrow \exists y P(x,y)]$$

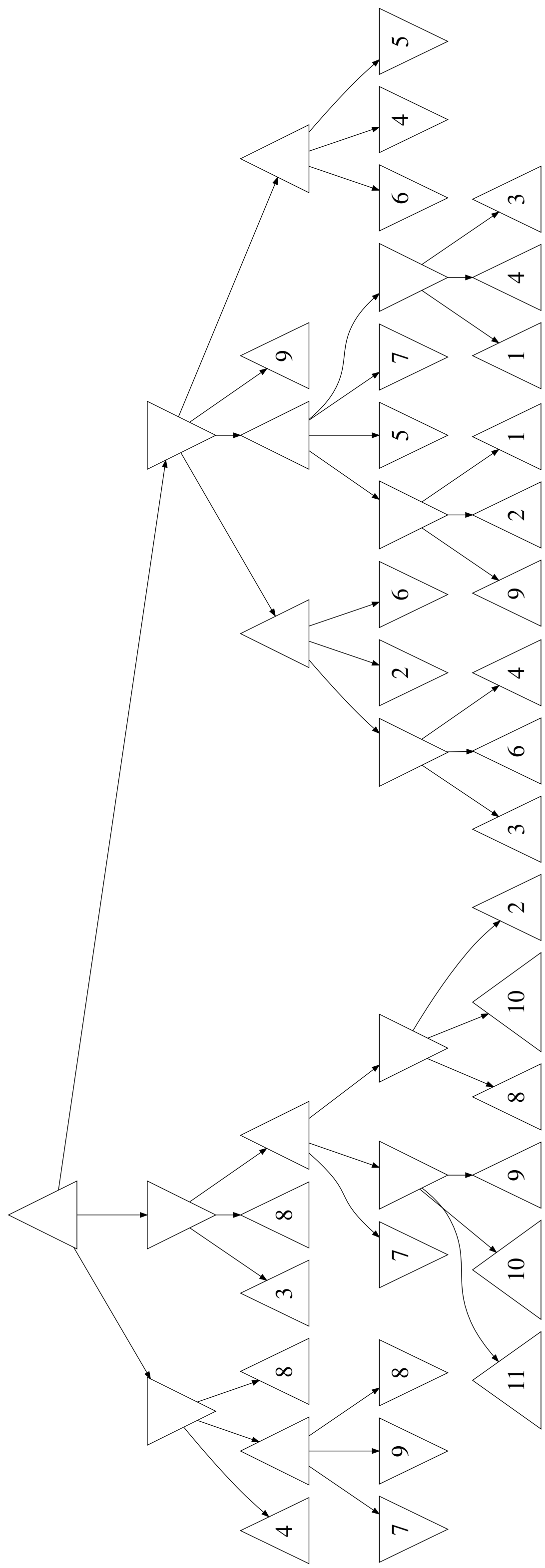
Question 12 (3 pts): Apply the following substitutions to the expressions (1 point each);

1. Apply $\{x/f(A)\}$ to $P(x,y) \vee Q(x)$.
2. Apply $\{x/A, y/f(z)\}$ to $P(x,y) \vee Q(x)$.
3. Apply $\{y/x\}$ to $P(x,y) \vee Q(x)$.

Question 13 (12 pts): For each of the following, (1) find the unifier, and (2) show the unified expression. For example, given $P(A)$ and $P(x)$, the unifier would be $\{x/A\}$, and the unified expression $P(A)$. If the pair of expressions is not unifiable, indicate so. (3 points each):

1. $P(x, f(x))$ and $P(A, f(B))$
2. $P(x, A)$ and $P(y, y)$
3. $P(x, f(x), y)$ and $P(A, f(g(w)), z)$
4. $P(A, f(y), y, y)$ and $P(x, f(g(x)), g(A), w)$

Full-size print of the game tree, for practice, etc. (copy 1)



Full-size print of the game tree, for practice, etc. (copy 2)