1. (25 pts) On inference rules:
   
   (a) Discuss the concept of sound.
   
   (b) Discuss the concept of complete.
   
   (c) Consider the modus ponens inference rule (given \((A \Rightarrow B) \land B\), \(B\) is inferred), explain whether it is sound and complete.
   
   (d) When we use the resolution inference rule for proving a logical sentence \(S\), explain why when we achieve an empty clause, \(S\) is inferred to be true.
2. (25 pts) On decision tree learning:

(a) Consider learning a boolean function with \( n \) boolean attributes/variables, explain how many possible boolean functions can be represented by (different) decision trees. [Hint: you might want to start with \( n = 2 \).]

(b) Explain in what situation that the decision tree learning algorithm could have no remaining attributes to use and the examples in a leaf are still not of the same target class.
3. (25 pts) Consider a *modified* version of tic-tac-toe where one can only win if you get three pieces along a diagonal or along an edge of the board (not along the middle rows or columns). Assume that by now you (‘x’) have already placed a piece in the center and the opponent (‘o’) has placed a piece in the N-W (northwest or upper left) corner, and it is again your turn.

(a) Show each step of an alpha-beta pruning to decide what to do next based on a maximum traversal of depth 4 (describe the heuristic you select for node ordering, and the evaluation function that you use)

(b) Assume at ply 3 you placed a piece in the N (north or top center) cell of the board, in ply 4 the opponent placed a piece in the S-W (southwest or lower left) corner and in ply 5 you place a piece in the W (west or middle left) cell. Show each step of mini-max your opponent will use for deciding what to do next. Does he have a ‘draw’ strategy?

```
  o  x  
 x  x  
  o  _  
```
4. (25 pts) On Planning:

(a) What are the 3 main parts of a planning problem when modelled with STRIPS operators? situation calculus?

(b) Model the Sussman anomaly problem using STRIPS operators:

The Susman anomaly problem asks you to move the block objects A, B, and C found in the initial configuration:

```
C
A  B
-------------Table
```

to the final configuration:

```
A
B
C
-------------Table
```

when the possible operations allow moving one block at a time, from the current position to another position on the table or on top of another block.