1. (20 pts)

(a) Which of the following are HORN clauses?
   i. $B \Rightarrow A$
   ii. $C \land D \Rightarrow B$
   iii. $B \lor \neg C \lor \neg D$
   iv. $B \lor \neg C \lor D$

(b) When is an inference technique refutation complete?

(c) Which type of inference is known to be refutation complete for First Order Logic? Give the formula.

2. (20 pts) Compare A* and Iterative Deepening from the point of view of the properties that they offer (when applied to problems like finding the best path in a graph).
3. (20 pts) (CSPs) The stable matching problem consists of a set of employers $E_1, ..., E_n$ and a set of graduates $G_1, ..., G_m$ that want to be employed. Each graduate $G_i$ has a preference for the employers (defined by an ordered list of employers $P_i[1..n]$, employer $P_i[a]$ is prefered by $G_i$ to employer $P_i[b]$ if and only if $a < b$). Formalize this problem as a CSP.

4. (10 pts) While learning a decision tree, describe two criteria for not growing the tree further?
5. (30 pts) Consider the following pseudocode for state space search:

```c
float search(initialState, goalState, operators, cost, heuristic)
{
    // search returns the cost of reaching the goal state from the initial state
    // initialState is the starting/initial state
    // goalState is the target/goal state
    // operators is a list of possible operators, each operator (op) takes
    // a state and generates another state after applying op.
    // ie, op(state) returns a state
    // cost is a function that returns the cost of applying an operator
    // ie, cost(op)

    // completeness
    if (initialState == goalState)
    {
        return 0; // no cost to reach the goal
    }

    // optimality
    float cost = float.MaxValue;

    // exploring all operators
    foreach (op in operators)
    {
        state new = op(initialState);
        float newCost = cost + cost(op);
        if (new == goalState)
        {
            return newCost; // found the goal
        }
        if (newCost < cost)
        {
            cost = newCost;
            // update the current state
            initialState = new;
        }
    }

    // failure
    return float.MaxValue;
}
```

(a) the last parameter of `search()` is `heuristic`, which is a function; describe `heuristic()` in terms of
   i. input parameter(s) and
   ii. the purpose (return value)

(b) Complete the pseudocode for implementing the best-first search algorithm.

(c) For breadth-first search, what are the `cost` and `heuristic` functions?

(d) Mark and describe changes you need to make in your pseudocode for implementing the breadth-first search algorithm.