**(a) Artifical intelligence (A.I.) can be useful in describing behaviours of NonPlaying Characters (NPC’s) in games. The implementation of A.I. is usually separated into sensing, thinking and acting sections. If we intend to implement a Finite State Machine (FSM) to deal with the thinking section…**

**(i) What is an FSM? (2 Marks)**

An FSM is defined by a list of its states, its initial state, and the conditions for each transition.

**(ii) How would a FSM be implemented? (2 Marks)**

enum States = {standing, fight, search, chase}

enum Transitions = {see\_something, is\_attacked, hear\_something, none}

States currentState = States.standing;

Transitions currentTransitions = Transitions.none;

switch(currentStates){

case States.standing:

switch(currentTransitions){

case Transition.see\_something:

return States.chase;

case Transition.is\_attacked:

return States.fight;

case Transition.hear\_something:

return States.search;

case Transition.none:

return currentState;

}

}

**(iii) Discuss, in the context of this example, the purpose of the sensing section. (2 Marks)**

From the above code the sensing section will fill the currentTransition

**(iv) Discuss, in the context of this example, the purpose of the acting section. (2 Marks)**

From the above code the acting section will change the action the player is doing at a specific game loop when the transition is changed

**(v) What are Hierarchical FSM’s and Markov Models? What advantages do these have over FSM’s? (2 Marks)**

**(b) The AI used in games like Chess and Draughts would be seen as a Search Problem**

**(i) How do Search Problems work? (3 Marks)**

Search Problem creates a tree which will describe all possible moves till the end of the game

**(ii) What is the Utility Function? (1 Mark)**

A utility function for a given player assigns a number for every possible outcome of the game with the property that a higher number implies that the outcome is more preferred.

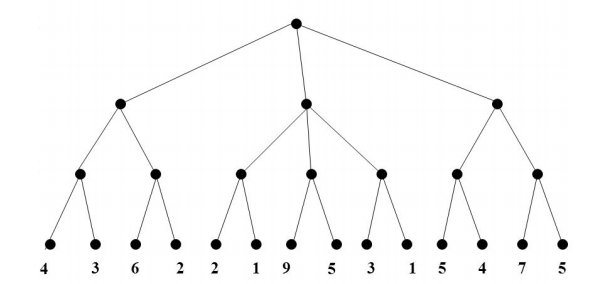
**(iii) What is an Evaluation Function? (1 Mark)**

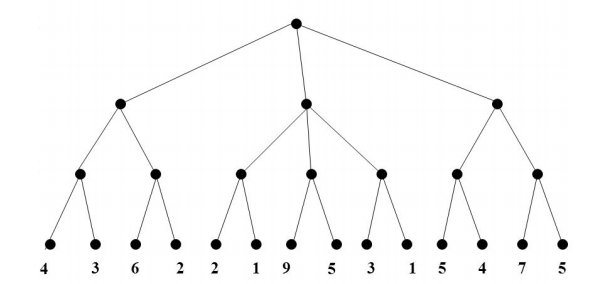
An evaluation function, also known as a heuristic evaluation function or static evaluation function, is a function used by game-playing programs to estimate the value or goodness of a position in the minimax and related algorithms.

**(iv) Compare the practicality of the implementations of the AI for Tic-Tac-Toe against a game like chess or draughts. What are the main issues in producing reasonable AI? How could the performance be improved etc. (5 Marks)**

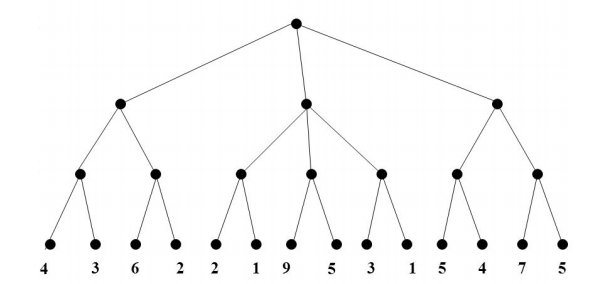
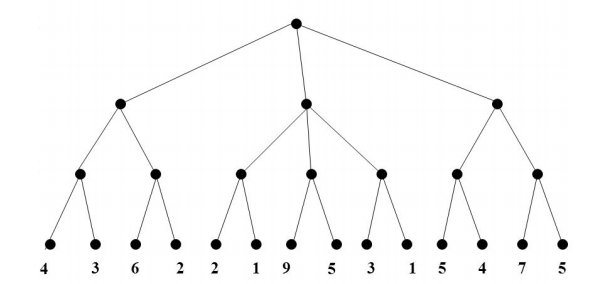
The main issue in producing reasonable AI is there would be too many outcomes to code. For chess there can be 30^n for each move. This would be a massive issue because it would be impossible to code all the outcomes of the game. Ways to reduce this problem would be pruning and symmetry.

**(v) Apply the Min/Max algorithm to determine which is the correct next move, show the full tree with filtered weights in your answer.**

c



**4 6 2 9 3 5 7**



**5**

**4 2 5**

