Mini-max Algorithm (Module-3)

-> Name :- Tejas Balaso Jadhav.

-> Roll No. :- 24

-> Sub :- IS-LAB

-> Class :- BE-I.T.

-> Sem :- VII

Mini-Max Algorithm (Module 3)

-> Mini-max algorithm

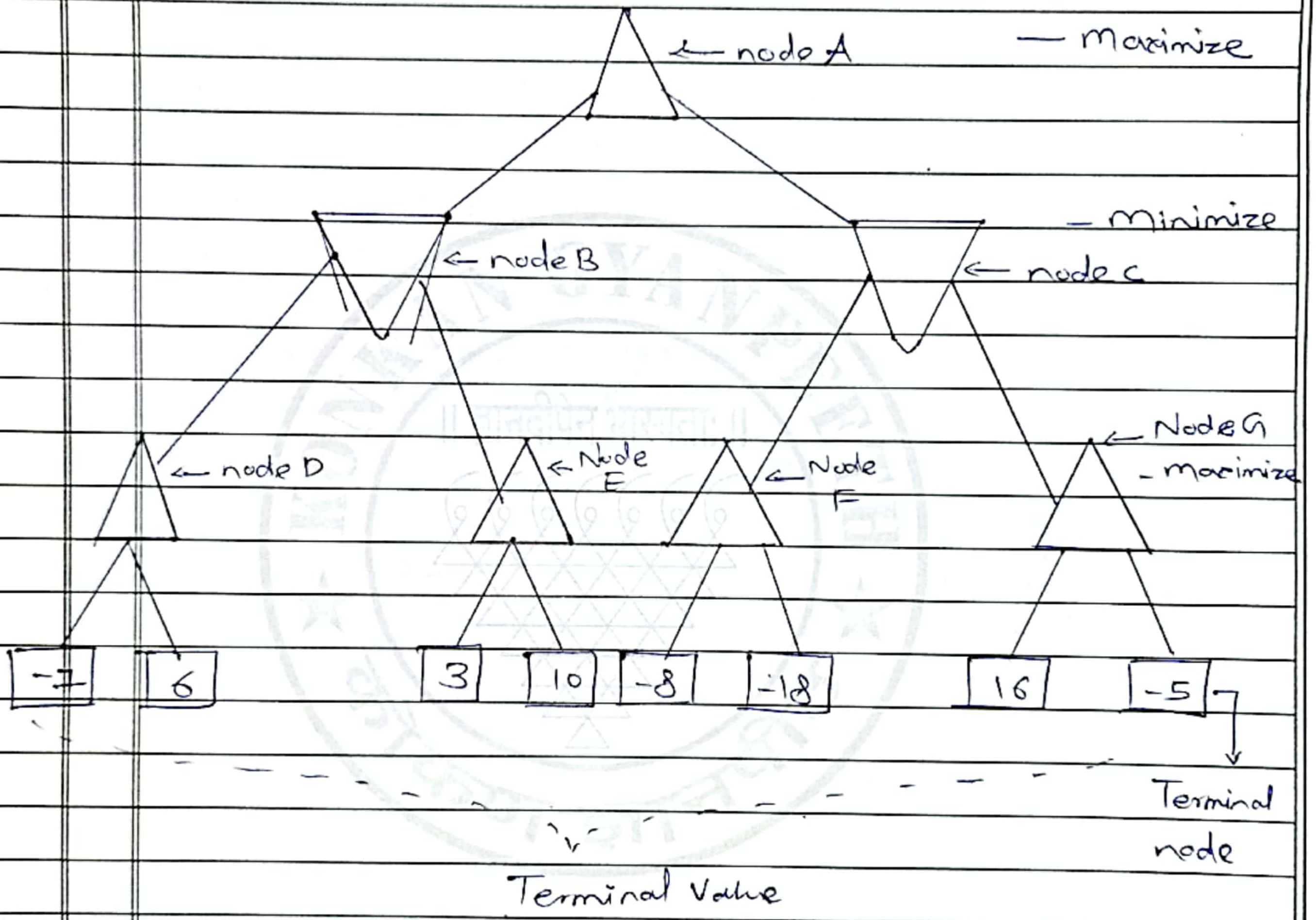
-> Mini-max algorithm is a recursive or back tracking algorithm which is used in decision making & game theory. It provide an optional move for the player assuming that oppenent is also playing.

-> mini-max algorithm uses recurrsion to search through the game-tree.

-> In this algorithm two players play the game one is called Max & other is min.

-> Min-Max algorithm is mostly used for gameplays in A.I. Such as chess, checkers, tic-tac-toe This Algorithm computers the minimax decision for the current state.

Step-1: In the first step, the algorithm generates the entire game-tree & apply the utility function to get the utility values for the terminal states. In the below tree diagram. let's take A is the initial state of the tree. Suppose maximizer takes first turn which has worst-case intial value is negative infinity, & minimizer will take next which has worst-case initial value is positive infinidy.



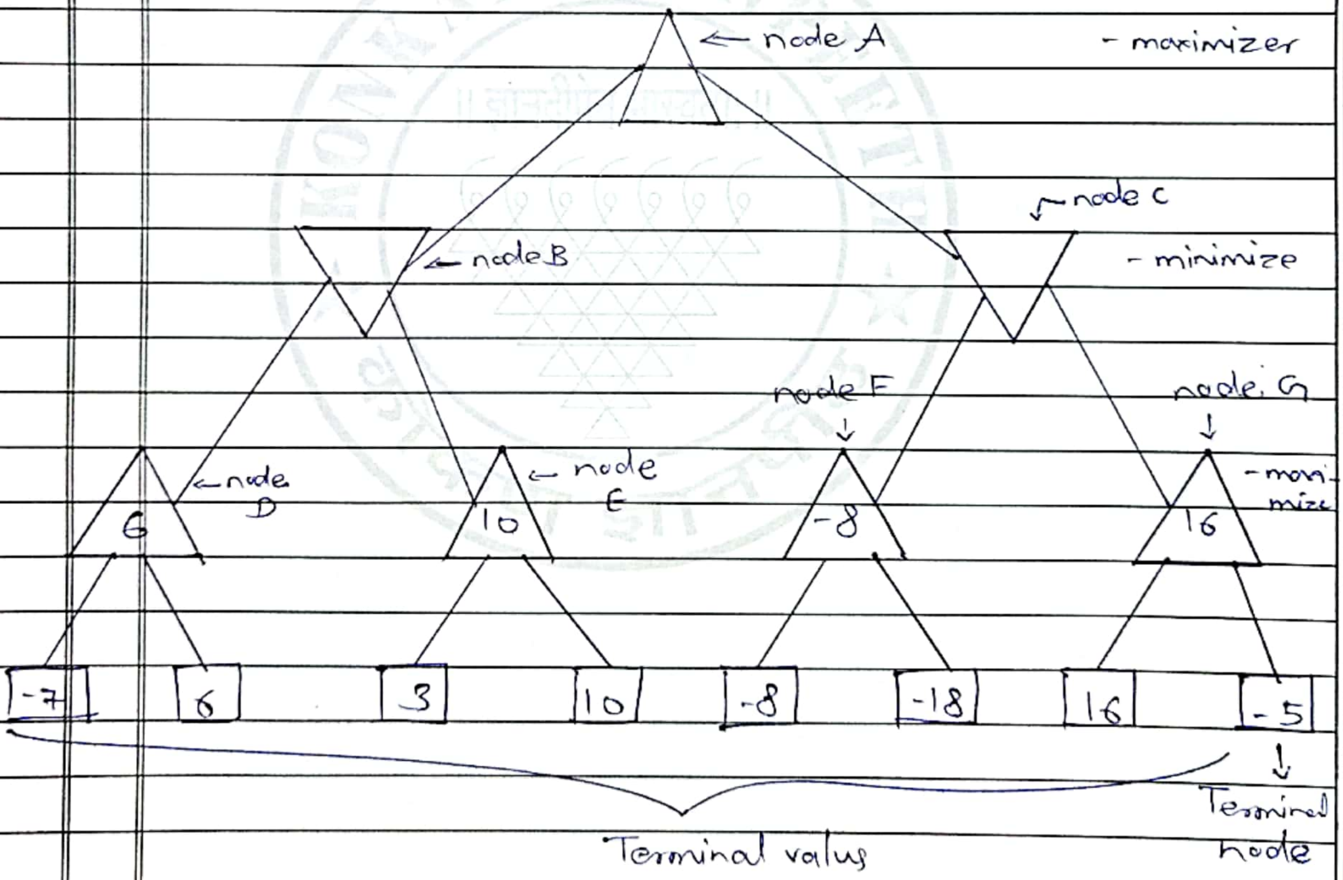
Step 2: Now, first we find the utilities value for the maximizer, its initial value is -∞, so we will compare each value in terminal state with initial value of maximum and determines. the higher nodes values. It will find the maximum among the all.

- for node D, max (-7, -∞) => max (-7, 6) = 6

- for node E ; max (3, -∞) => max (3, 10) = 10

- For node F; max (-8, -∞) => max (-8, -18) = -8

- for node G; max (16, -∞) => max (16, -5) = 16



Step 2:- Now, First we find the utilities value for the maximizer, it's initial value is -∞, so we will compare each value in terminal.

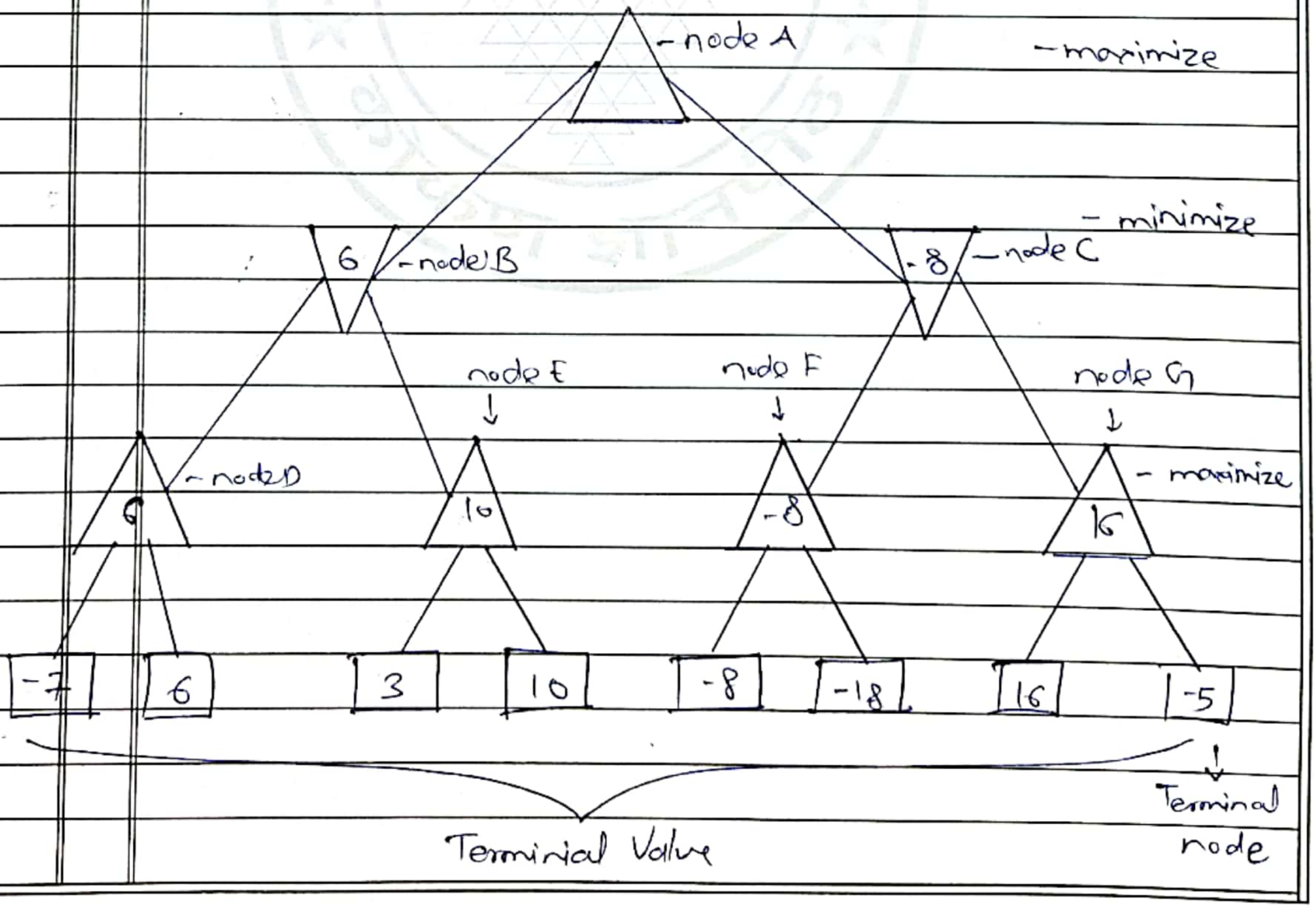
~~state with initial value of maximize & determines the higher nodes values. It will find the maximum among the all.~~

-> ~~For node D max~~

Step 3 :- In the next step, its a turn for minimizer, so it will compare all nodes value with too, & will find the 3rd layer node value.

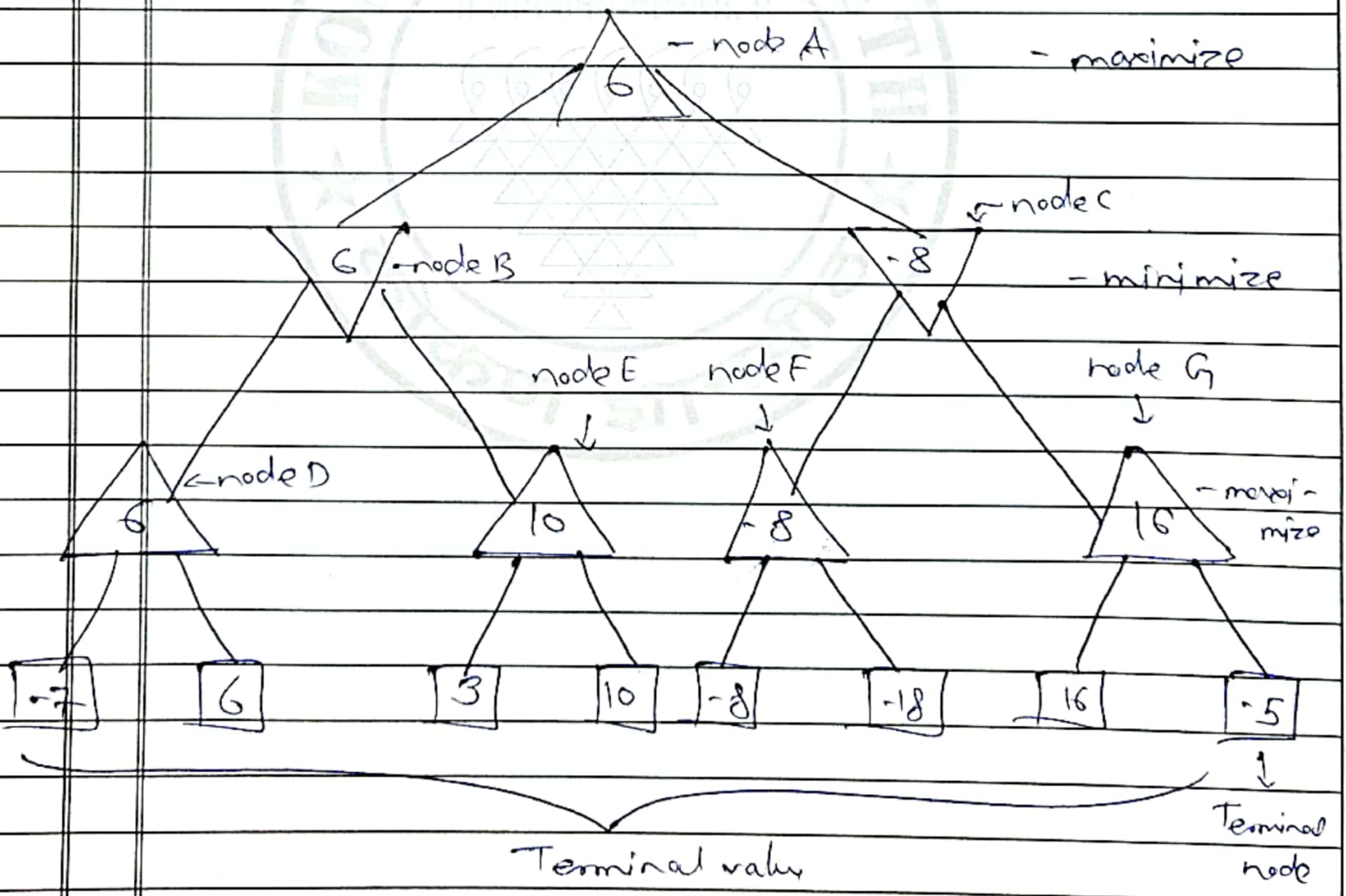
- for node B = min (6, 10) = 6

- for node C = min (-8, 16) = -8



Step 4:- Now it's a turn for maximizer, & it will again choose the maximum of all nodes values & find the maximum value for the root node. In this game ~~tt~~ tree, there are only 4 layers, here we reach immediately to the root node, but in real games, there will be more than 4 layes.

- For node A max (6, -8) = 6



That way the complete workflow of the minimax algorithm with two player game.