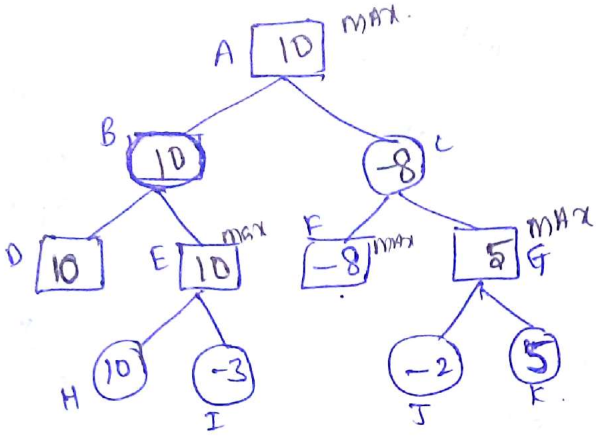
AI

Subhani shaik
MT18117

6) Min-MAX Algorithm :-

□ = MAX
○ = min



at E :- 10, -3. takes max ⇒ 10.

at B : takes min (10, 10) = 10

at G :- max (−2, 5) = 5

at C :- min (−8, 5) = −8.

at A :- MAX (10, −8) = 10.

⇒ At root MAX = 10.

Alpha-Beta pruning:-

Beta :- is the minimum upper bound of possible solutions.

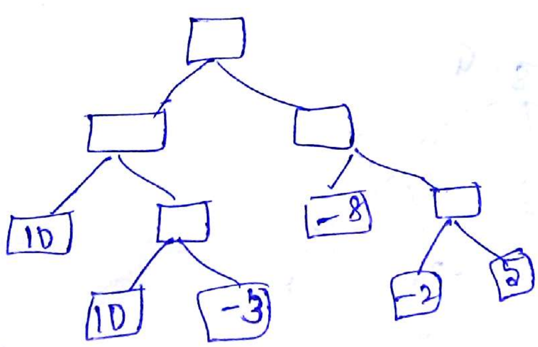
Alpha :- is the maximum lower bound of possible solutions.

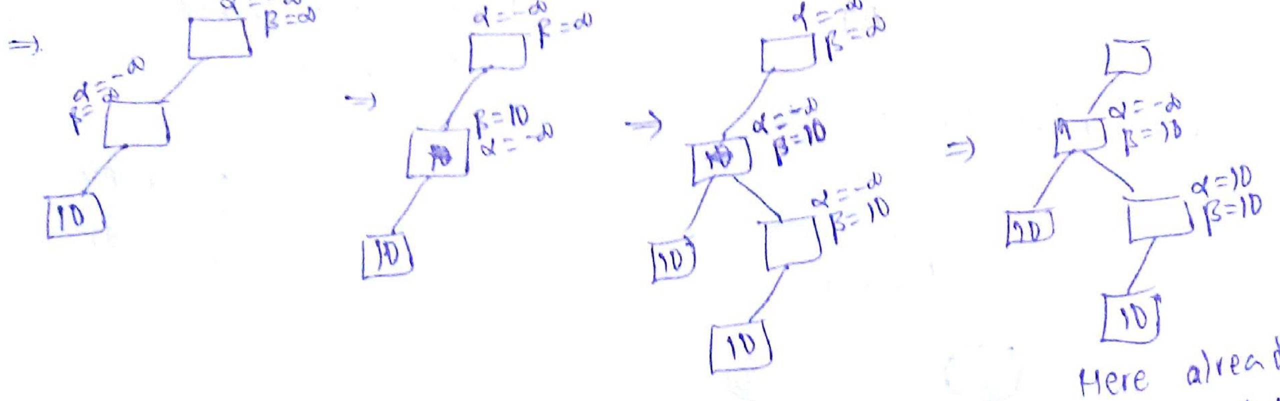
Initially $\alpha, \beta \Rightarrow -\infty, \infty$ respectively.

$\alpha \equiv$ MAX

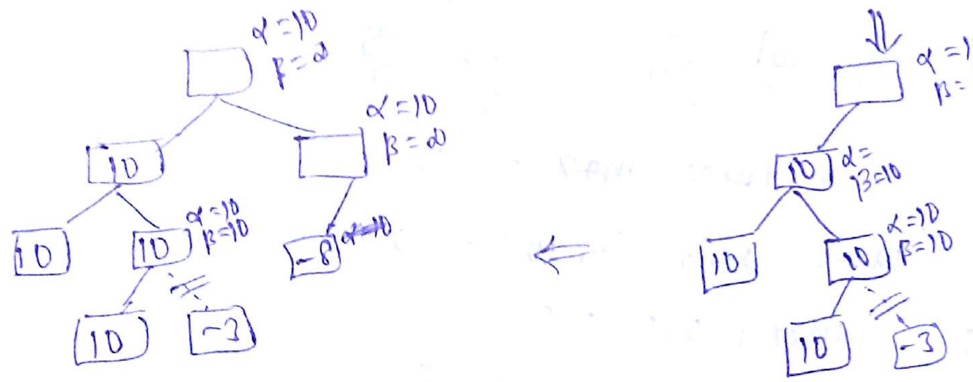
$\beta \equiv$ MIN.

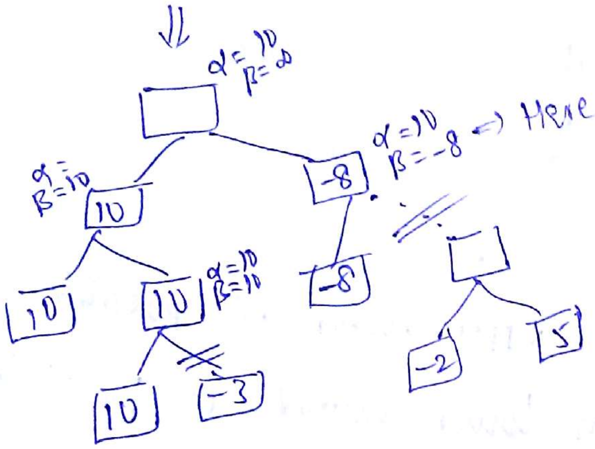
Given



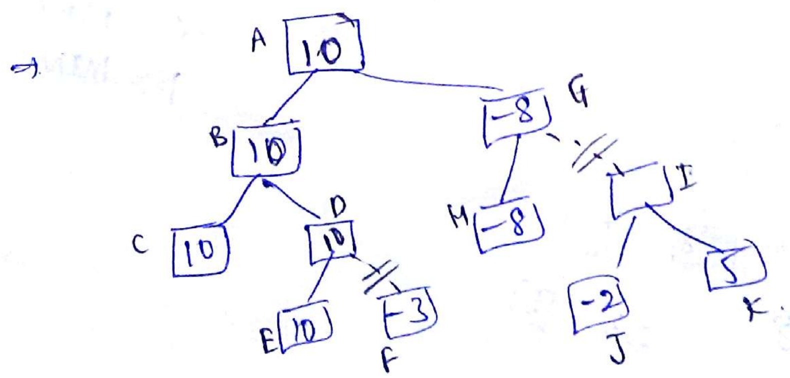


Here already. $\beta=10$.
So, no need to visit
another child.



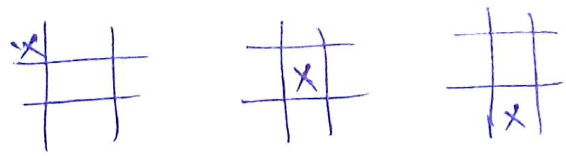


$\alpha=10$, $\beta=-8$ ⇒ Here already $\alpha=10.$, & $\beta=-8$.
So, no need to visit Right child
bcz, even if we visit it
we will get value <-8.
So prune it.



⇒ By. Alpha-Beta pruning nodes F, I are
pruned.

4) Given 'A' performs first move randomly in one of the three moves
→ at [1] or
→ at [5] or
→ at [8].



A, B are using Minimax algorithm. ⇒ 'A' tries to win & 'B' tries to reduce A's win.

but individually both try to win.

⇒ Initially 'A' has 3 choices.
later 'B' has 8 choices
'A' → 7 choices. etc
⋮

⇒ 'A' works on choices of 3, 7, 5, 3, 1
Initially given.
'B' works on choices of 8, 6, 4, 2 choices

If both A, B plays optimally.
Then winning probability of 'B' is '0'.
'A' is '0'.
&

& no one will lose. ⇒ probability of lose = 0.
but probability of Draw = 1.

⇒ If both players A, B plays optimally.

probability:
win (B) = 0
Draw = 1 ⇒ always. bcz we know that A, B both plays to win.
lose = 0.