27/10/25

Lab 10

Adversarial Search
Implement Alpha - Beta Pruning

Algorithm

first 1. start at the root node (current game state)
the current player is either Max or min

2. Initialize
   - $\alpha = -\infty$
   - $\beta = +\infty$

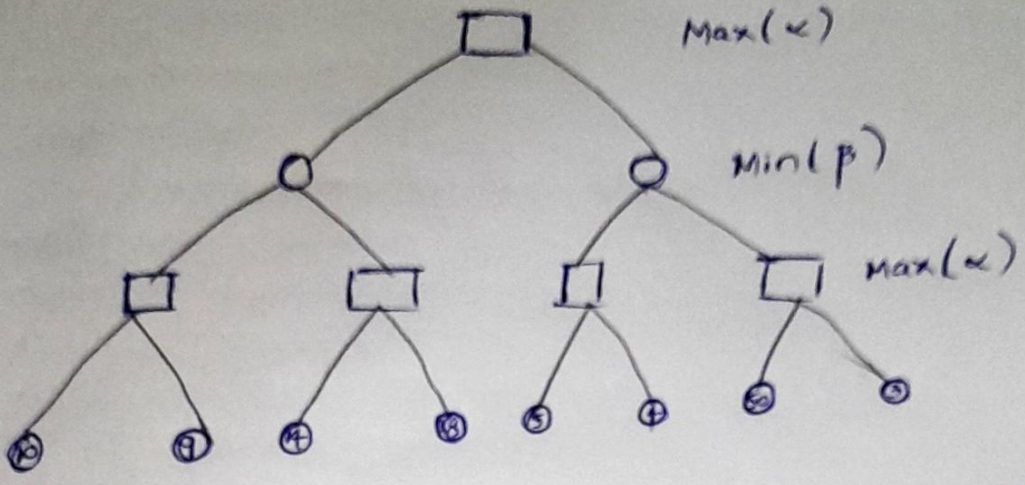
3. If terminal node (end of game):
   → Return the utility (score) of that node

4. If its a Max player:
   - set value = $-\infty$
   - For each child of this node:
     1) compute child_value = AlphaBeta(child, depth-1, α, β, False)
     2) update value = max(value, child_value)
     3) update α = max(value, child_value)
     4) if $\alpha \geq \beta$, then break → (prune remaining branches)
   - Return value

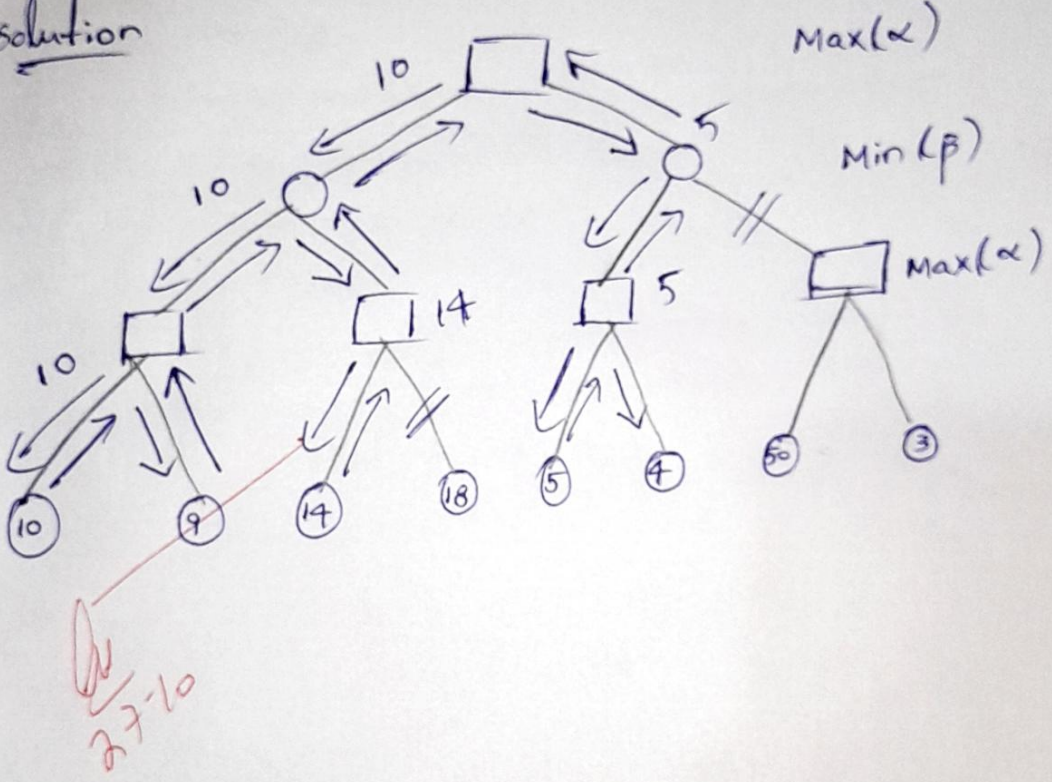
5. If its a Min player:
   - set value = $+\infty$
   - For each child_value =
     1) AlphaBeta(child, depth-1, α, β, True)
     2) update value = min(value, child_value)
     3) update β = min(β, value)

4) if $\alpha \geq \beta$, then break -> (prune remaining branches)

• Return value



solution



27-10