Date 11/12/202
Page

Lab-8

* Alpha Beta pruning

$\alpha \leftarrow -\infty$

$\beta \leftarrow +\infty$

$v \leftarrow$ max-value (state, α, β)

return the outlier a in ACTIONS(state) with value v

max-value function(state, α, β)

if TERMINAL-TEST(state):

return UTILITY(STATE)

$v \leftarrow -\infty$

For each action in ACTIONS(state)

$v \leftarrow$ MAX (v, MIN-VALUE(RESULT(state, α, β))

if $v \geq \beta$

return v

$\alpha \leftarrow$ MAX(α, v)

return v

function min-value (state, α, β):

if TERMINAL-TEST (state, α, β)

return UTILITY(STATE)

$v \leftarrow +\infty$

For each action in ACTION(state)

$v \leftarrow$ MIN(v, MAX-VALUE(RESULT(state, α, β))

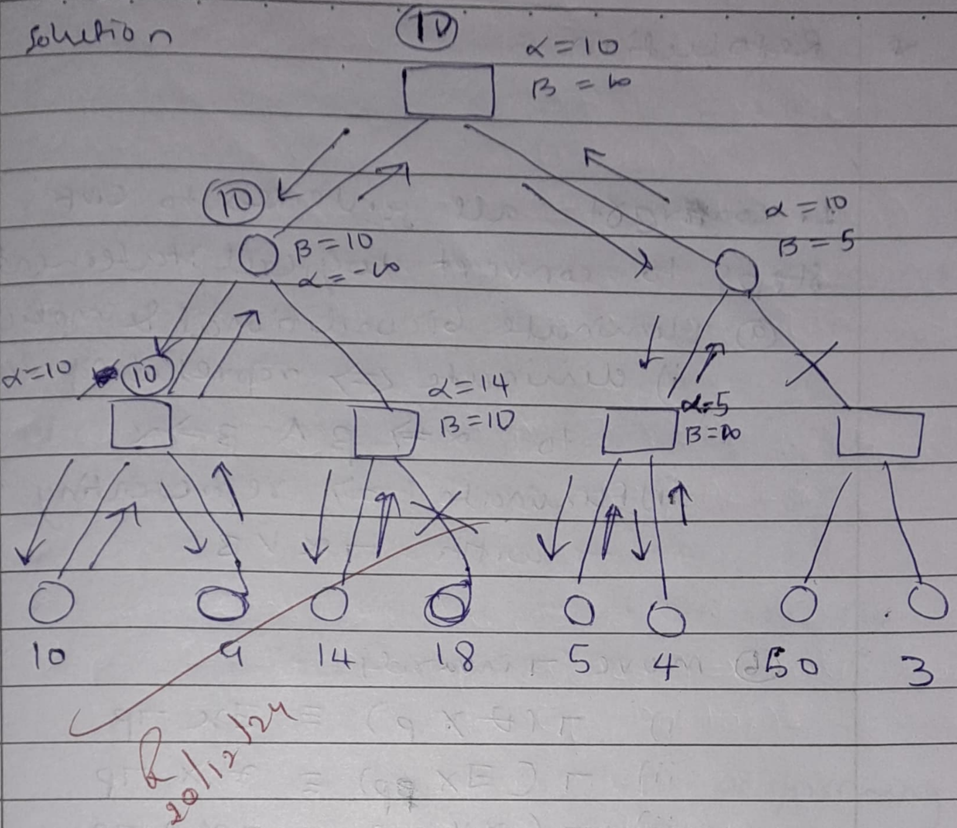
if $v \leq \alpha$

return v

$\beta \leftarrow$ MIN(β, v)

return v

Solution



```python
print("Name:Sudarshan Komar","USN:1BM22CS291",sep="\n")

def alpha_beta_pruning(depth, node_index, maximizing_player, values,
alpha, beta):
    # Base case: leaf node
    if depth == 3:
        return values[node_index]

    if maximizing_player:
        max_eval = float('-inf')
        # Explore two children
        for i in range(2):
            eval = alpha_beta_pruning(depth + 1, node_index * 2 + i,
False, values, alpha, beta)
            max_eval = max(max_eval, eval)
            alpha = max(alpha, eval)
            if beta <= alpha:
                print(f"Pruning at depth {depth}, node {node_index}
(maximizing)")
                break  # Beta cut-off
        return max_eval
    else:
        min_eval = float('inf')
        # Explore two children
        for i in range(2):
            eval = alpha_beta_pruning(depth + 1, node_index * 2 + i,
True, values, alpha, beta)
            min_eval = min(min_eval, eval)
            beta = min(beta, eval)
            if beta <= alpha:
                print(f"Pruning at depth {depth}, node {node_index}
(minimizing)")
                break  # Alpha cut-off
        return min_eval


# Example usage
# Tree with 8 leaf nodes
values = [3, 5, 6, 9, 1, 2, 0, -1]
print("Optimal value is:", alpha_beta_pruning(0, 0, True, values,
float('-inf'), float('inf')))
```

```
Name:Sudarshan Komar
USN:1BM22CS291
Pruning at depth 2, node 1 (maximizing)
Pruning at depth 1, node 1 (minimizing)
Optimal value is: 5
```