## **Tracing the Minimax Algorithm on the Given Game Tree**

### **1. Game Tree Structure**

We are given:

MAX  
 / \  
 MIN MIN  
 / \ / \  
 5 6 4 7

**Leaf nodes (utility values) left to right:** 5, 6, 4, 7

* MAX node is the root (player trying to maximize score)
* MIN nodes are the children (opponent trying to minimize score)

### **2. Calculating Minimax Values**

**Step 1: Evaluate the MIN nodes**

* **Left MIN node:** Children values = 5, 6 MIN chooses **minimum** → min(5, 6) = 5
* **Right MIN node:** Children values = 4, 7 MIN chooses **minimum** → min(4, 7) = 4

**Step 2: Evaluate the root MAX node**

* Children values from MIN nodes = 5, 4
* MAX chooses **maximum** → max(5, 4) = 5

### **3. Updated Game Tree with Minimax Values**

MAX (5)  
 / \  
 MIN (5) MIN (4)  
 / \ / \  
 5 6 4 7

### **4. Optimal Move for MAX Player**

* Compare values of left and right MIN nodes: 5 (left) vs 4 (right)
* MAX player should choose the branch with the **highest value**, which is the **left branch**.

### **✅ Summary of Results**

1. **Minimax value of left MIN node:** 5
2. **Minimax value of right MIN node:** 4
3. **Minimax value of root MAX node:** 5
4. **Optimal first move for MAX player:** Move **left**