**Experiment 3: MINIMAX Algorithm**

**Aim**

To implement the MINIMAX algorithm for decision making in game trees.

**Procedure**

1. Generate the game tree.  
2. Assign a score to each leaf node.  
3. Propagate scores back using MIN and MAX at alternate levels.  
4. Choose the move with the optimal score.

**Code**

def minimax(depth, nodeIndex, isMax, scores, h):  
 if depth == h:  
 return scores[nodeIndex]  
 if isMax:  
 return max(minimax(depth+1, nodeIndex\*2, False, scores, h),  
 minimax(depth+1, nodeIndex\*2 + 1, False, scores, h))  
 else:  
 return min(minimax(depth+1, nodeIndex\*2, True, scores, h),  
 minimax(depth+1, nodeIndex\*2 + 1, True, scores, h))  
  
scores = [3, 5, 6, 9, 1, 2, 0, -1]  
h = 3  
print("The optimal value is :", minimax(0, 0, True, scores, h))

**Output**

