**AIM: IMPLEMENTING THE MINMAX ALGORITHM FOR AN APPLICATION**

**CODE:**

**import** math

**def** minimax (curDepth, nodeIndex,

maxTurn, scores,

targetDepth):

*# base case : targetDepth reached*

**if** (curDepth **==** targetDepth):

**return** scores[nodeIndex]

**if** (maxTurn):

**return** max(minimax(curDepth **+** 1, nodeIndex **\*** 2,

**False**, scores, targetDepth),

minimax(curDepth **+** 1, nodeIndex **\*** 2 **+** 1,

**False**, scores, targetDepth))

**else**:

**return** min(minimax(curDepth **+** 1, nodeIndex **\*** 2,

**True**, scores, targetDepth),

minimax(curDepth **+** 1, nodeIndex **\*** 2 **+** 1,

**True**, scores, targetDepth))

*# Driver code*

scores **=** [3, 5, 2, 9, 12, 5, 23, 23]

treeDepth **=** math**.**log(len(scores), 2)

print("The optimal value is : ", end **=** "")

print(minimax(0, 0, **True**, scores, treeDepth))

OUTPUT:

