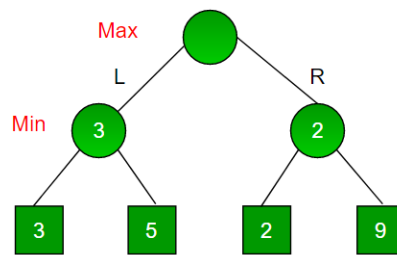


Lab 8**1. Write code for Min max Algorithm**

```
# maximizing player can get
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, 3, 5, 2, 9]
treeDepth = math.log(len(scores), 2)
print("The optimal value is : ", end = "")
print(minimax(0, 0, True, scores, treeDepth))
```

Lab 8 Task

- 1. Write code for Min Max Algorithm**