**NAME:SAKCHI CHOUDHARY**

**REGISTRATION NO:RA1911042010035**

**BATCH:CSBS-R1**

**EXPERIMENT NO: 7**

# TOPIC: MIN MAX ALGORITHM WITH APPLICATION

# CODE FOR MIN MAX ALGORITHM IN ALPHA BETA PRUNING

# #include <iostream>

# #include <algorithm>

# #include <cmath>

# #include <climits>

# #define SIZE(arr) (sizeof(arr) / sizeof(arr[0]))

# using namespace std;

# int getHeight(int n) {

# return (n == 1) ? 0 : 1 + log2(n / 2);

# }

# int minmax(int height, int depth, int nodeIndex,

# bool maxPayer, int values[], int alpha,

# int beta) {

# if (depth == height) {

# return values[nodeIndex];

# }

# if (maxPayer) {

# int bestValue = INT\_MIN;

# for (int i = 0; i < height - 1; i++) {

# int val = minmax(height, depth + 1, nodeIndex \* 2 + i, false, values, alpha, beta);

# bestValue = max(bestValue, val);

# alpha = max(alpha, bestValue);

# if (beta <= alpha)

# break;

# }

# return bestValue;

# }

# else {

# int bestValue = INT\_MAX;

# for (int i = 0; i < height - 1; i++) {

# int val = minmax(height, depth + 1, nodeIndex \* 2 + i, true, values, alpha, beta);

# bestValue = min(bestValue, val);

# beta = min(beta, bestValue);

# if (beta <= alpha)

# break;

# }

# return bestValue;

# }

# }

# OUTPUT:

# OUTPUT:

# Text Description automatically generated