**TASK 4 :**

**Implementation of Mini-Max algorithm using recursion to search through the Game tree using python.**

**Program :**

MAX, MIN = 1000, -1000

def minimax(depth, nodeIndex, maximizingPlayer, values, alpha, beta):

if depth == 3:

return values[nodeIndex]

if maximizingPlayer:

best = MIN

for i in range(0, 2):

val = minimax(depth + 1, nodeIndex \* 2 + i, False, values, alpha, beta)

best = max(best, val)

alpha = max(alpha, best)

if beta <= alpha:

break

return best

else:

best = MAX

for i in range(0, 2):

val = minimax(depth + 1, nodeIndex \* 2 + i, True, values, alpha, beta)

best = min(best, val)

beta = min(beta, best)

if beta <= alpha:

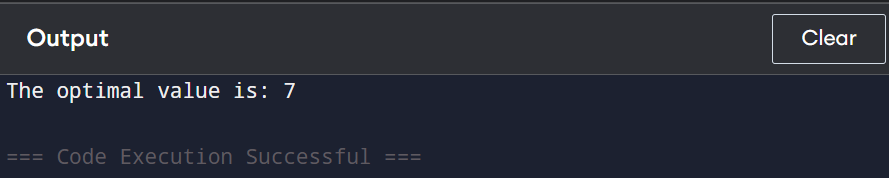
break

return best

if \_\_name\_\_ == "\_\_main\_\_":

values = [10, -5, 3, 7, 8, -2, 4, 7]

print("The optimal value is:", minimax(0, 0, True, values, MIN, MAX))

**Output :**