Exp-14 alpha beta pruning for gaming

Code

def alpha\_beta(node, depth, alpha, beta, maximizingPlayer):

if depth == 0 or is\_terminal(node):

return evaluate(node)

if maximizingPlayer:

maxEval = float('-inf')

for child in get\_children(node):

eval = alpha\_beta(child, depth - 1, alpha, beta, False)

maxEval = max(maxEval, eval)

alpha = max(alpha, eval)

if beta <= alpha:

break

return maxEval

else:

minEval = float('inf')

for child in get\_children(node):

eval = alpha\_beta(child, depth - 1, alpha, beta, True)

minEval = min(minEval, eval)

beta = min(beta, eval)

if beta <= alpha:

break

return minEval

def is\_terminal(node):

return len(get\_children(node)) == 0

def evaluate(node):

return node

def get\_children(node):

if node <= 0:

return []

return [node - 1, node - 2]

start\_node = 5

best\_score = alpha\_beta(start\_node, depth=4, alpha=float('-inf'), beta=float('inf'), maximizingPlayer=True)

print("Best score:", best\_score)

ouput

