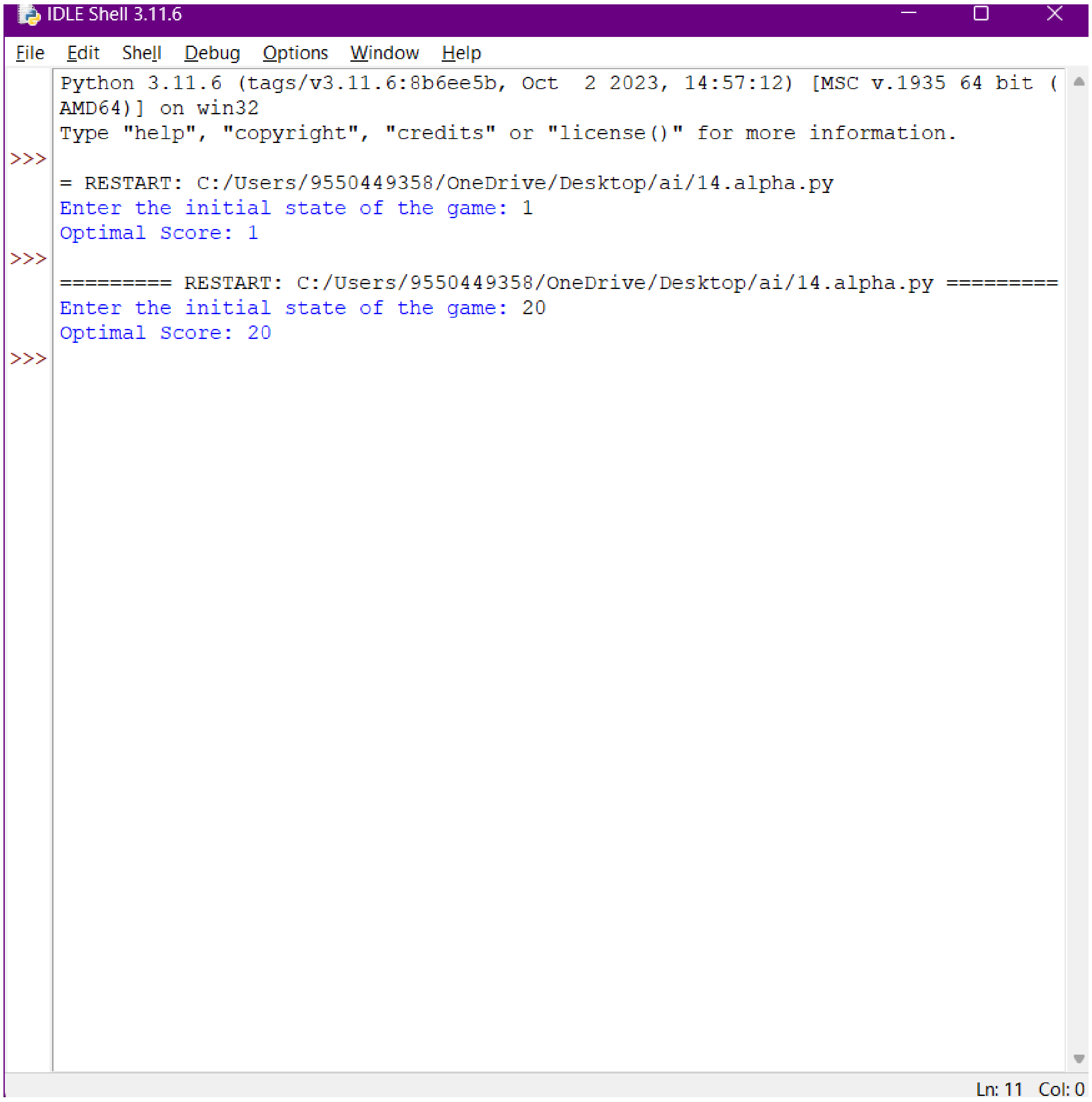
Date 04/03/2024

7.Aim: Write the python program to implement Apha & Beta pruning algorithm for gaming.

Program: import math

```
# Function to simulate the game. This is a placeholder and can be replaced with the actual game logic.
def evaluate_game_state(state):
    # For demonstration purposes, we return a simple evaluation score.
    return len(state)


# Alpha-Beta Pruning Algorithm
def alpha_beta_pruning(state, depth, alpha, beta, maximizing_player):
    if depth == 0 or evaluate_game_state(state) != 0:  # Depth limit or terminal node
        return evaluate_game_state(state)


    if maximizing_player:
        max_eval = -math.inf
        for child_state in generate_child_states(state):
            eval_score = alpha_beta_pruning(child_state, depth - 1, alpha, beta, False)
            max_eval = max(max_eval, eval_score)
            alpha = max(alpha, eval_score)
            if beta <= alpha:
                break
        return max_eval
    else:
        min_eval = math.inf
        for child_state in generate_child_states(state):
            eval_score = alpha_beta_pruning(child_state, depth - 1, alpha, beta, True)
            min_eval = min(min_eval, eval_score)
            beta = min(beta, eval_score)
            if beta <= alpha:
                break
        return min_eval

# Function to generate child states. This is a placeholder and should be replaced with the actual function generating child states.
def generate_child_states(state):
    # Placeholder logic to generate child states. Replace this with actual implementation.
    return [state]

# Main function to run the game
def main():
    # Initial state of the game
    initial_state = "Initial state of the game"

    # Define alpha and beta values
    alpha = -math.inf
    beta = math.inf

    # Perform Alpha-Beta Pruning
    optimal_score = alpha_beta_pruning(initial_state, 5, alpha, beta, True)  # Example depth 5

    # Print the optimal score
    print("Optimal Score:", optimal_score)

if __name__ == "__main__":
    main()
```

Output:



```
Python 3.11.6 (tags/v3.11.6:8b6ee5b, Oct  2 2023, 14:57:12) [MSC v.1935 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:/Users/9550449358/OneDrive/Desktop/ai/14.alpha.py
Enter the initial state of the game: 1
Optimal Score: 1
>>> 
========= RESTART: C:/Users/9550449358/OneDrive/Desktop/ai/14.alpha.py =========
Enter the initial state of the game: 20
Optimal Score: 20
>>> 
```

Result: The given program has been executed successfully