PROGRAM-16

ALPHA & BETA PRUNING PROBLEM

AIM:-

To write and execute the python program for the ALPHA & BETA pruning program.

PROCEDURE:-

Imports and Setup:

- Import the required library: random

Hill Climbing Function:

- Define the hill climbing algorithm function.
- Initialize the current state with the initial state.
- Loop indefinitely until the goal state is reached:
    - If the current state is the goal state, return it.
    - Generate neighboring states.
    - Sort the neighboring states based on their heuristic values.
    - If the heuristic value of the best neighboring state is not better than the current state, return the current state.
    - Update the current state with the best neighboring state.

Execution and Output:

- Execute the hill climbing algorithm with the defined initial and goal states, heuristic, and neighbors function.
- Print the result.

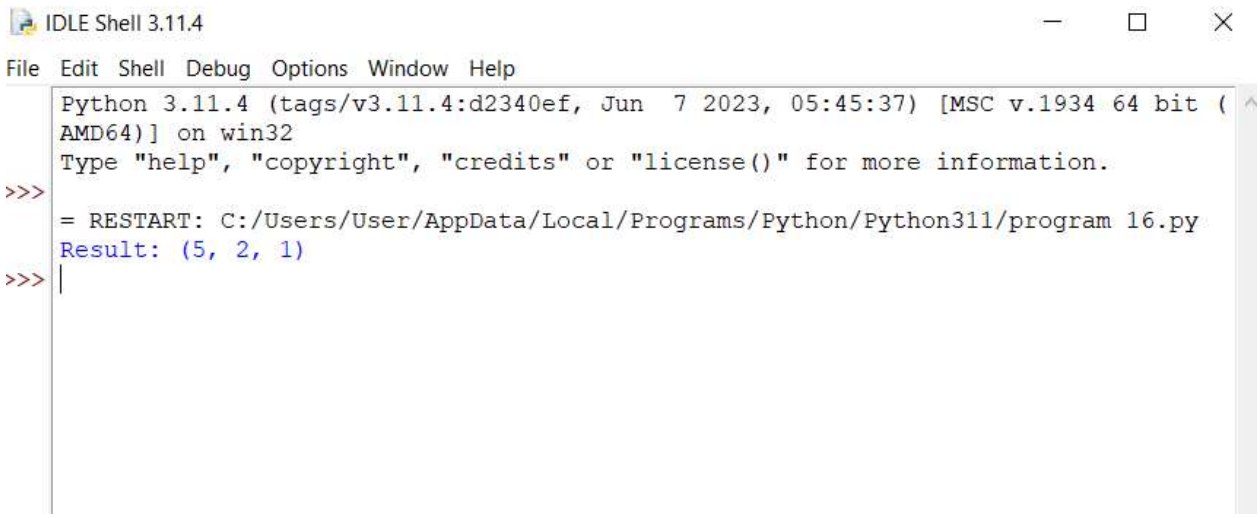
.

CODING:-

```
import random

def hill_climbing(initial_state, goal_state, heuristic, neighbors):

    current_state = initial_state

    while True:

        if current_state == goal_state:

            return current_state
```

```
        next_states = neighbors(current_state)
        next_states.sort(key=lambda state: heuristic(state, goal_state))
        if heuristic(next_states[0], goal_state) >= heuristic(current_state, goal_state):
            return current_state
        current_state = next_states[0]
def heuristic(state, goal_state):
    return sum(abs(state[i] - goal_state[i]) for i in range(len(state)))
def neighbors(state):
    neighbors = []
    for i in range(len(state)):
        neighbor = list(state)
        neighbor[i] += random.choice([-1, 1])
        neighbors.append(tuple(neighbor))
    return neighbors

initial_state = (0, 0, 0)
goal_state = (5, 5, 5)
result = hill_climbing(initial_state, goal_state, heuristic, neighbors)
print("Result:", result)
```

OUTPUT:-



RESULT:-

Hence the program has been successfully executed and verified.