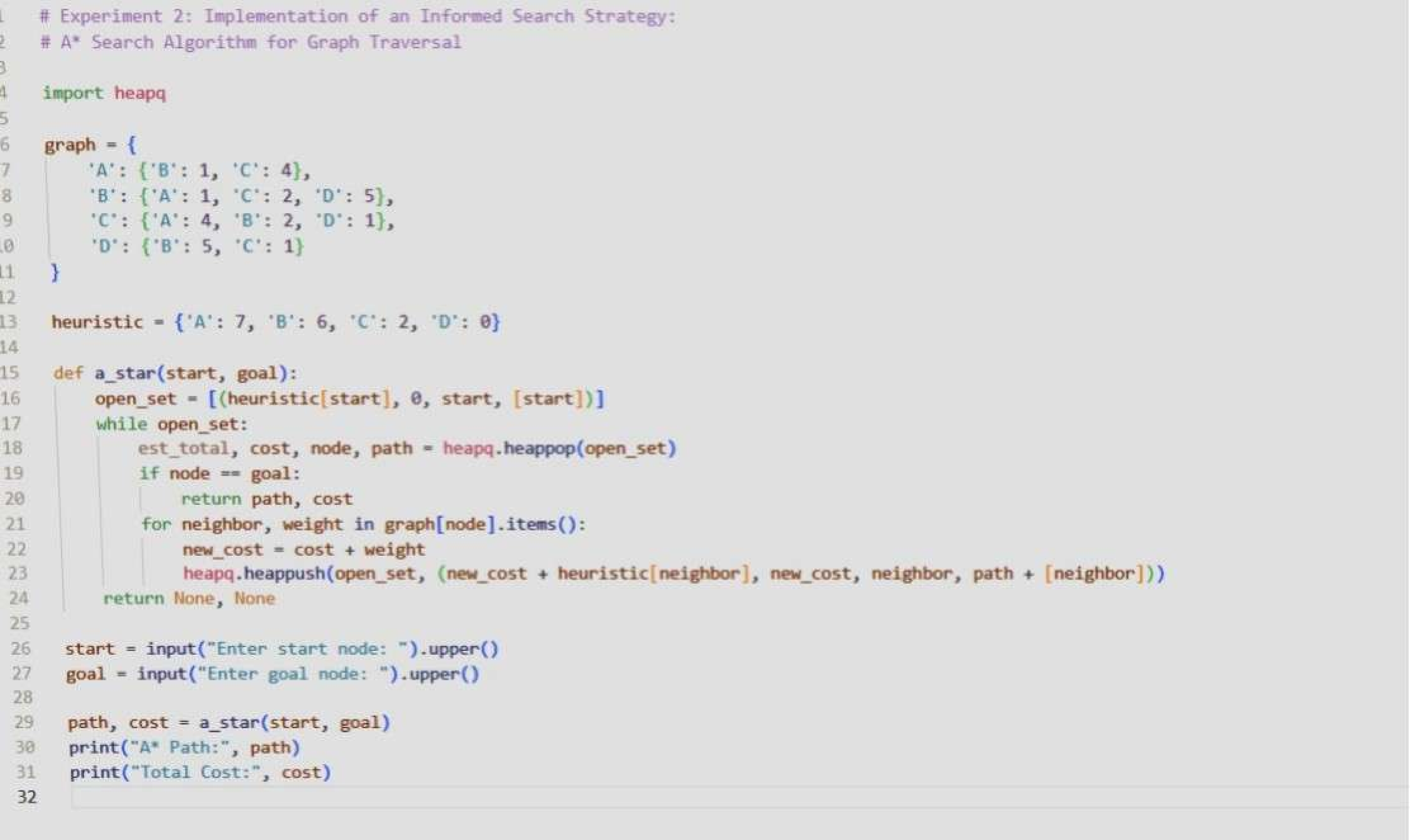
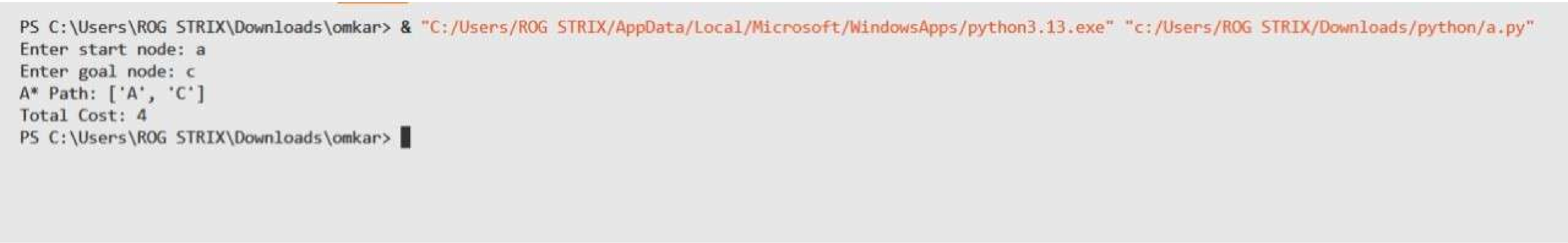


```python
# Experiment 2: Implementation of an Informed Search Strategy:
# A* Search Algorithm for Graph Traversal

import heapq

graph = {
    'A': {'B': 1, 'C': 4},
    'B': {'A': 1, 'C': 2, 'D': 5},
    'C': {'A': 4, 'B': 2, 'D': 1},
    'D': {'B': 5, 'C': 1}
}

heuristic = {'A': 7, 'B': 6, 'C': 2, 'D': 0}

def a_star(start, goal):
    open_set = [(heuristic[start], 0, start, [start])]
    while open_set:
        est_total, cost, node, path = heapq.heappop(open_set)
        if node == goal:
            return path, cost
        for neighbor, weight in graph[node].items():
            new_cost = cost + weight
            heapq.heappush(open_set, (new_cost + heuristic[neighbor], new_cost, neighbor, path + [neighbor]))
    return None, None

start = input("Enter start node: ").upper()
goal = input("Enter goal node: ").upper()

path, cost = a_star(start, goal)
print("A* Path:", path)
print("Total Cost:", cost)

```



```
PS C:\Users\ROG STRIX\Downloads\omkar> & "C:/Users/ROG STRIX/AppData/Local/Microsoft/WindowsApps/python3.13.exe" "c:/Users/ROG STRIX/Downloads/python/a.py"
Enter start node: a
Enter goal node: c
A* Path: ['A', 'C']
Total Cost: 4
PS C:\Users\ROG STRIX\Downloads\omkar> █
```