

Problem – 2 : Print Shortest Path – Dijkstra's Algorithm

```
import heapq
```

```
def dijkstra_shortest_path(graph, n):  
    # Initialize distance and visited arrays  
    distance = [float('inf')] * (n + 1)  
    distance[1] = 0  
    visited = [False] * (n + 1)  
  
    # Priority queue to keep track of nodes to visit  
    priority_queue = [(0, 1)]  
  
    while priority_queue:  
        dist, node = heapq.heappop(priority_queue)  
  
        if visited[node]:  
            continue  
  
        visited[node] = True  
  
        for neighbor, weight in graph[node]:  
            if distance[neighbor] > distance[node] + weight:  
                distance[neighbor] = distance[node] + weight  
                heapq.heappush(priority_queue, (distance[neighbor], neighbor))  
  
    if distance[n] == float('inf'):  
        return [-1]  
    else:  
        path = []  
        current = n  
        while current != 0:
```

```
path.append(current)
for neighbor, weight in graph[current]:
    if distance[current] == distance[neighbor] + weight:
        current = neighbor
        break

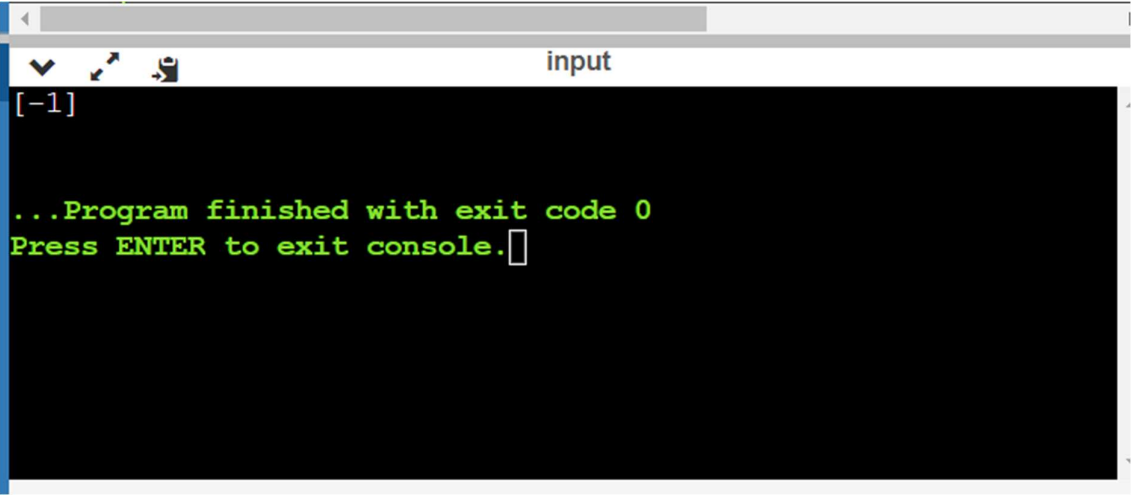
return path[::-1]
```

n = 4

m = 6

```
graph = {
    0: [(1, 2), (2, 4)],
    1: [(0, 2), (2, 1), (3, 7)],
    2: [(0, 4), (1, 1), (3, 3)],
    3: [(1, 7), (2, 3)]
}

shortest_path = dijkstra_shortest_path(graph, n)
print(shortest_path)
```



```
[-1]

...Program finished with exit code 0
Press ENTER to exit console.█
```