



UNIVERSITY OF PETROLEUM AND ENERGY STUDIES  
DEHRADUN

## **DAA LAB - 6**

### **Dijkstra's Algorithm**

MTECH-COMPUTER SCIENCE  
ENGINEERING  
CYBER SECURITY AND FORENSICS

Name: Jigesh Sheoran  
SAP ID: 590025428

**JIGESH SHEORAN**  
590025428  
M.Tech CSE

## Title : Implementation of Dijkstra's algorithm

**Implementation:** Language used : Python

🔗 dijkstra's algo.py ●

Users > sheoraninfosec > Documents > Masters UPES > DAA > Lab 7 > 🔗 dijkstra's algo.py > 📁 dijkstra

```
1  # Author: Jigesh Sheoran
2  # SAP ID: 590025428
3
4  import heapq
5
6  def dijkstra(graph, start):
7      # all other distance is infinity
8      distances = {node: float('inf') for node in graph}
9      distances[start] = 0          # Distance to itself = 0
10
11     algo = [(0, start)]
12
13     while algo:
14         current_distance, current_node = heapq.heappop(algo)
15
16         # skip if shorter path is already found
17         if current_distance > distances[current_node]:
18             continue
19
20         # explore neighbors
21         for neighbor, weight in graph[current_node].items():
22             distance = current_distance + weight
23
24             # if found a shorter path, update
25             if distance < distances[neighbor]:
26                 distances[neighbor] = distance
27                 heapq.heappush(algo, (distance, neighbor))
28
29     return distances
30
```

```

30
31 graph = {
32     'A': {'B': 3, 'C': 6},
33     'B': {'A': 3, 'C': 2, 'D': 1},
34     'C': {'A': 6, 'B': 2, 'D': 4, 'E': 2},
35     'D': {'B': 1, 'C': 4, 'E': 3, 'F': 5},
36     'E': {'C': 2, 'D': 3, 'F': 1},
37     'F': {'D': 5, 'E': 1}
38 }
39
40
41 start_node = input("Enter starting node (A-F): ").upper()
42
43 if start_node not in graph:
44     print("Invalid node!")
45 else:
46     shortest_paths = dijkstra(graph, start_node)
47     print(f"\nShortest distances from {start_node}:")
48     for node, dist in shortest_paths.items():
49         print(f"{start_node} → {node} = {dist}")
50

```

## Output:

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

● sheoraninfosec@Jigeshs-MacBook-Air / % /usr/local/bin/python3 "/Users/sheoraninfosec/Documents/M  
Enter starting node (A-F): a

Shortest distances from A:

```

A → A = 0
A → B = 3
A → C = 5
A → D = 4
A → E = 7
A → F = 8

```

○ sheoraninfosec@Jigeshs-MacBook-Air / % □

-----END OF DOCUMENT -----