

**INSTITUTE OF COMPUTER TECHNOLOGY**  
**B-TECH COMPUTER SCIENCE ENGINEERING 2025-26**  
**SUBJECT:-ALGORITHM ANALYSIS & DESIGN**

NAME: Rahul Prajapati

ENRLL NO: 23162171020

BRANCH: CYBER SECURITY

BATCH: 52

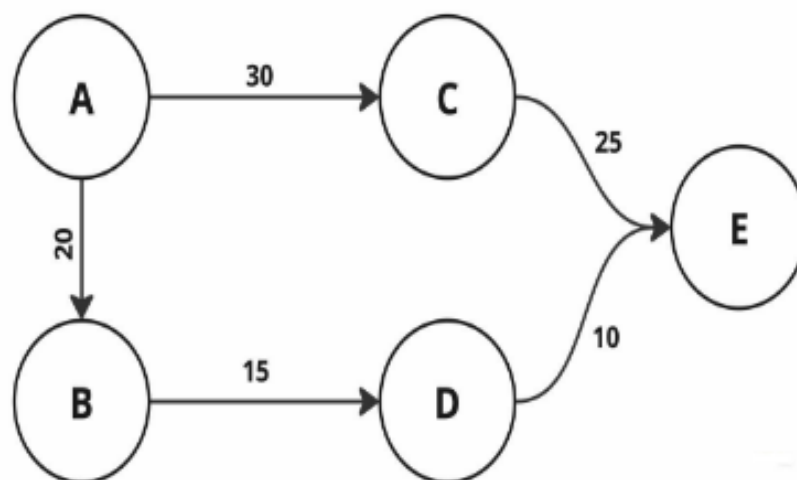
**PRACTICAL\_10**

**AIM:**

A government official needs to visit several cities within a state. To minimize travel costs, they want to find the shortest path between their starting city and each destination city.

**Task:**

Given a graph representing the cities and their connecting roads, determine the minimum cost path from a given starting city to all other cities.



## CODE:

```
Practical10_1.py > ...
1  def dijkstra(graph, start):
2      n = len(graph)
3      visited = [False] * n
4      dist = [float('inf')] * n
5      dist[start] = 0
6
7      for _ in range(n):
8          u = -1
9          for i in range(n):
10             if not visited[i] and (u == -1 or dist[i] < dist[u]):
11                 u = i
12
13             visited[u] = True
14
15             for v in range(n):
16                 if graph[u][v] != float('inf') and dist[u] + graph[u][v] < dist[v]:
17                     dist[v] = dist[u] + graph[u][v]
18
19     return dist
20
21 nodes = ["A", "B", "C", "D", "E"]
22 graph = [
23     [0, 20, 30, float('inf'), float('inf')],
24     [float('inf'), 0, float('inf'), 15, float('inf')],
25     [float('inf'), float('inf'), 0, float('inf'), 25],
26     [float('inf'), float('inf'), float('inf'), 0, 10],
27     [float('inf'), float('inf'), float('inf'), float('inf'), 0]
28 ]
29
30 start_node = "A"
31 start_index = nodes.index(start_node)
32
33 distances = dijkstra(graph, start_index)
34
35 print("Source → Destination → Cost")
36 for i, d in enumerate(distances):
37     print(f"{start_node} → {nodes[i]} = {d}")
38
```

## OUTPUT:

```
(.venv) PS C:\Users\Hp\OneDrive\Desktop\SEM_05\Algorithm Analysis & Design\SOURCE_CODES> python .\Practical10_1.py
Source → Destination → Cost
A → A = 0
A → B = 20
A → C = 30
A → D = 35
A → E = 45
```