Dijkstra's Algorithm

Dijkstra's algorithm solves the single-source shortest-paths problem on a weighted, directed graph $G = (V, E)$ for the case in which all edge weights are nonnegative.

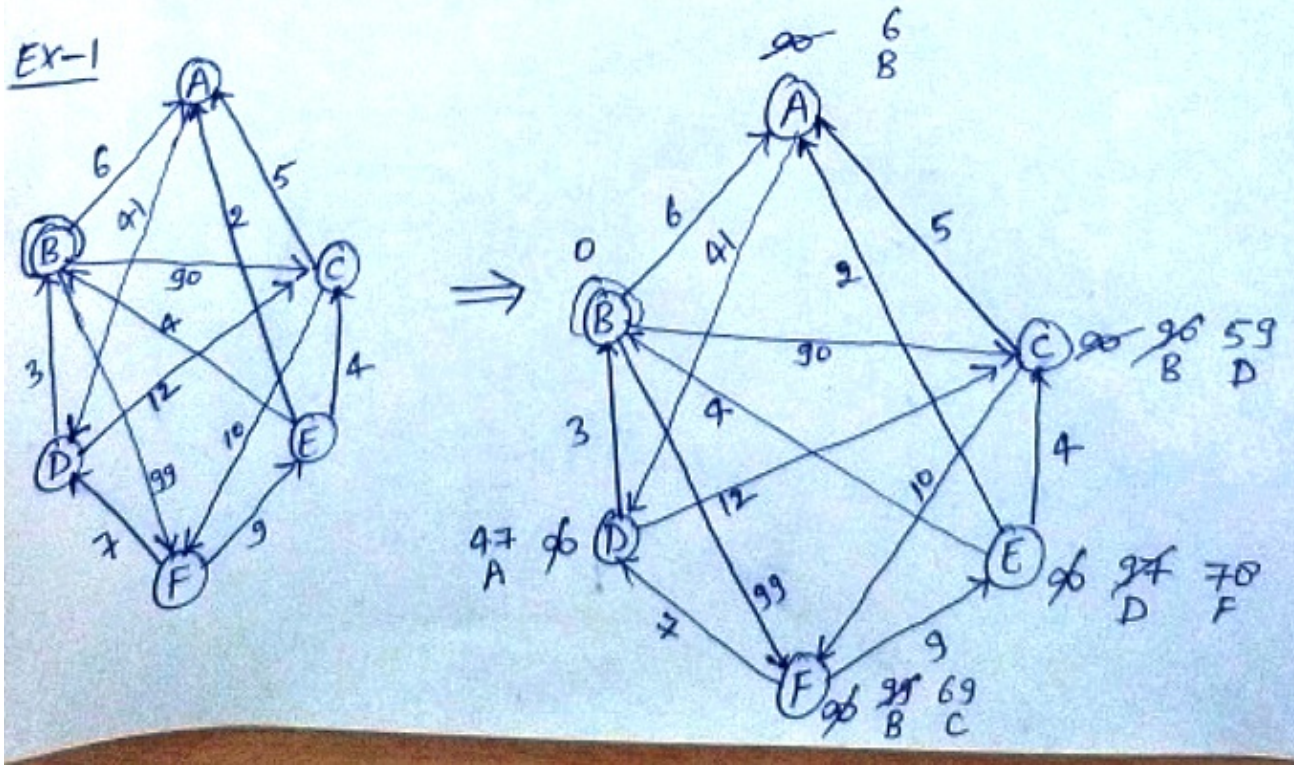
Dijkstra's algorithm maintains a set S of vertices whose final shortest-path weights from the source s have already been determined.

```
Dijkstra(G, w, s)
1. Initialize-Single-Source(G, s)
2. S = ∅
3. Q = G.V
4. while Q ≠ ∅
5.     u = Extract-Min(Q)
6.     S = S ∪ {u}
7.     for each vertex v ∈ G.Adj[u]
8.         Relax(u, v, w)
```

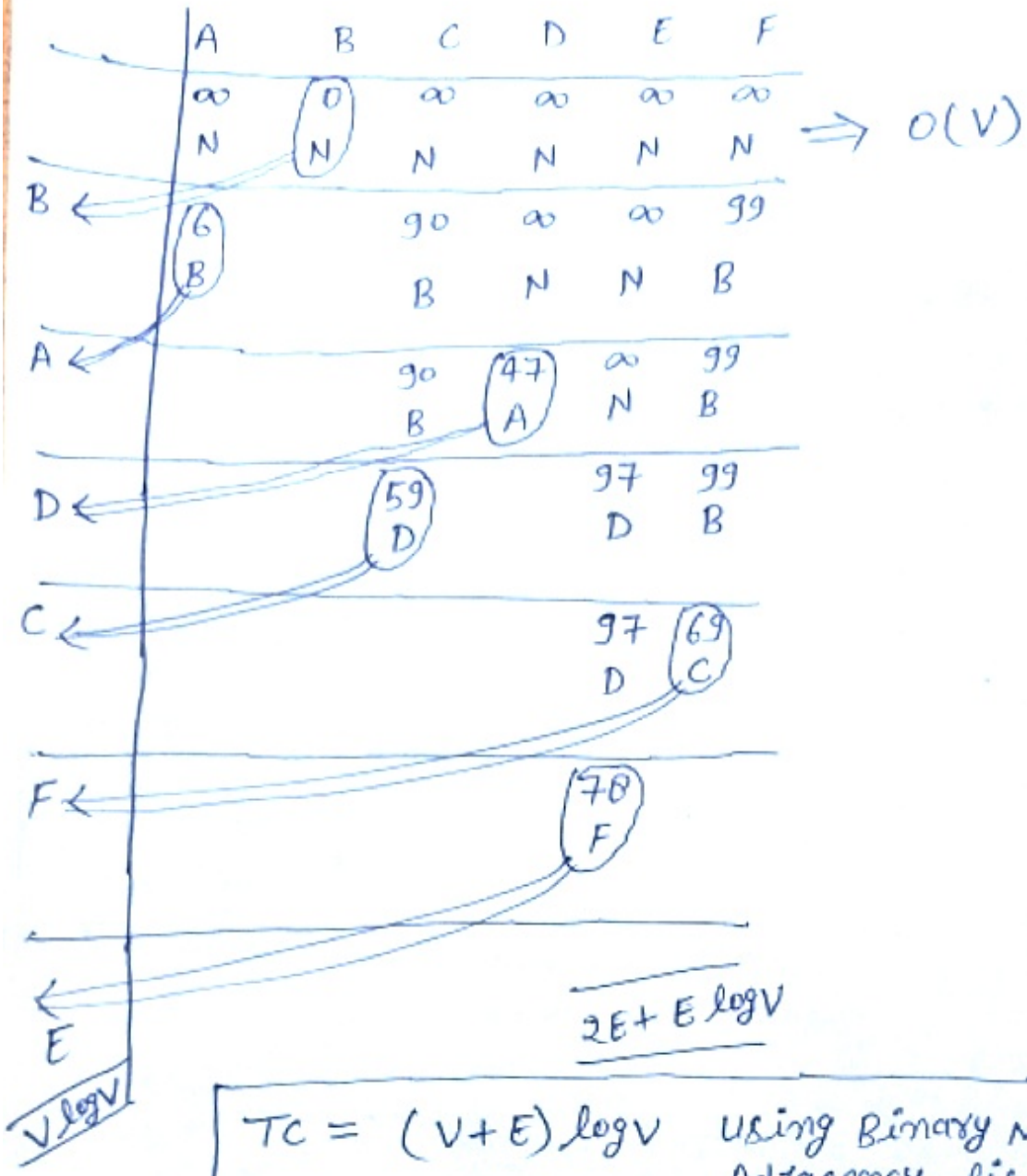
```
Initialize-Single-Source(G, s)
1. for each vertex v ∈ G.V
2.     v.d = ∞
3.     v.π = NIL
4. s.d = 0
```

```
Relax(u, v, w)
1. if v.d > u.d + w(u, v)
2.     v.d = u.d + w(u, v)
3.     v.π = u
```

Ex-1



Sol- Ex-1



TC = (V+E) logV	using Binary Min heap & Adjacency list
TC = O(V²)	using Array and Adjacency list

Q.1 print the sequence of vertices identified Dijkstra's, source is B.

Ans → B — A — D — C — F — E

Q.2 What will be the cost of shortest path from B to E.

Ans → 78

Q.3 What will be the shortest path from B to E.

Ans → B → A → D → C → F → E