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

**DIJKSTRA**$(G, w, s)$

1. **INITIALIZE-SINGLE-SOURCE**$(G, s)$
2. $S = \emptyset$
3. $Q = G.V$
4. while $Q \neq \emptyset$
   5. $u = \text{EXTRACT-MIN}(Q)$
   6. $S = S \cup \{u\}$
   7. for each vertex $v \in G.Adj[u]$
      8. $\text{RELAX}(u, v, w)$

**Theorem 24.6 (Correctness of Dijkstra’s algorithm)**
Dijkstra’s algorithm, run on a weighted, directed graph $G = (V, E)$ with non-negative weight function $w$ and source $s$, terminates with $u.d = \delta(s, u)$ for all vertices $u \in V$. 
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($G, w, s$)**

1. **INITIALIZE-SINGLE-SOURCE($G, s$)**
2. $S = \emptyset$
3. $Q = G.V$
4. while $Q \neq \emptyset$
   
   5. $u = \text{EXTRACT-MIN}(Q)$
   6. $S = S \cup \{u\}$
   7. for each vertex $v \in G.Adj[u]$
      
      8. $\text{RELAX}(u, v, w)$

**Theorem 24.6 (Correctness of Dijkstra’s algorithm)**

Dijkstra’s algorithm, run on a weighted, directed graph $G = (V, E)$ with nonnegative weight function $w$ and source $s$, terminates with $u.d = \delta(s, u)$ for all vertices $u \in V$. 

**Running Time**

- $\Theta(V)$
- $\Theta(1)$
- $\Theta(V)$
- $V + 1$
- $V \times O(lg V)$
- $V \times O(C)$
- $E \times O(lg V)$

worst-case $O(E lg V)$ maybe better $O(E+V) lg V$
C is the source vertex.
Q = \{ b, c, a \}

10 \text{ not changed, } 16 \text{ is worse than } 14