
Input: Graph $G = \langle V, E \rangle$, source node $source$

Output: Two lists of distances and predecessors for each node

```
procedure DIJKSTRA( $G, source$ )
   $dist \leftarrow$  initialize to array of size  $|G.V|$ 
   $prev \leftarrow$  initialize to array of size  $|G.V|$ 
   $Q \leftarrow$  empty priority queue with vertex as value and weight as key

  for  $v \in G.V$  do
    if  $v \neq source$  then
       $dist[v] \leftarrow \infty$ 
       $prev[v] \leftarrow$  UNDEFINED

   $dist[source] \leftarrow 0$ 
   $Q.push(source, 0)$ 

  while  $Q \neq \emptyset$  do
     $u \leftarrow Q.remove\_min()$ 

    for  $e = (u, v) \in u.outgoing$  do
       $w \leftarrow dist[u] + e.weight$ 

      if  $w < dist[v]$  then
         $dist[v] \leftarrow w$ 
        if  $prev[v] =$  UNDEFINED then
           $Q.add(v, w)$ 
        else
           $Q.update\_priority(v, w)$ 
         $prev[v] \leftarrow u$ 

  return  $dist, prev$ 
```
