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 ← initialize to array of size |G.V|
    prev ← initialize to array of size |G.V|
    Q ← empty priority queue with vertex as value and weight as key

    for v ∈ G.V do
        if v ≠ source then
            dist[v] ← ∞
        prev[v] ← UNDEFINED

    dist[source] ← 0
    Q.push(source, 0)

    while Q ≠ ∅ do
        u ← Q.remove_min()

        for e = (u, v) ∈ n.outgoing do
            w ← dist[u] + e.weight

            if w < dist[v] then
                dist[v] ← w
                if prev[v] = UNDEFINED then
                    Q.add(v, w)
                else
                    Q.update_priority(v, w)
                prev[v] ← u
return dist, prev
```