```
**Data:** Graph: $G(V, E)$, Dictionary: $weights$, Vertex: $root$
**Result:** Distance dictionary: $dist$, Previous node dictionary: $prev$
**for** $v \in V$ **do**
    $dist[v] \leftarrow \infty$
    $prev[v] \leftarrow \emptyset$
    $Q$.enqueue(v)
**end**
dist[$root$] $\leftarrow 0$
**while** $Q$ *is not empty* **do**
    $v \leftarrow$ the element in $Q$ which has the smallest distance to $root$
    $Q$.remove($v$)
     // Forall neighboors of $v$
    **for** $u \in \mathcal{N}(v)$ **do**
        $alt \leftarrow$ dist[$u$] $+ weights(u, v)$
        **if** $alt <$ *dist[u]* **then**
            $dist[u] \leftarrow$ alt
            $prev[u] \leftarrow v$
        **end**
    **end**
**end**
```