Junyan Li

CS325-001

Practice Assignment #3.

1. Pseudocode:

```
Prim ( G = (V, E) , root )
    for each u ∈ V
        u.keyEdge = INF
        u.parent = NULL
    root.keyEdge = 0
    PriorityQueue Q = V
    while Q ≠ ∅
        u = popup(Q)
        for each v ∈ adjacent of u
            if v ∈ Q and w(u,v) < v.keyEdge
                v.keyEdge = w(u,v)
                v.parent = u
```

Assume we use the binary min-heap. So if we want to build a heap, that will cost $O(V)$. For every popup function, the cost is $O(\log V)$ and in total we do it $O(V \cdot \log V)$. For the for-loop we do it for $O(E \log V)$. So the total time is $O((V+E)\log V)$.

2. Pseudocode:

```
Boruka(G = (V, E))
    disjoint sets D = {{v1}, ..., {vn}}
    if |D| > 1
        for each S ∈ D
            for each vertex v ∈ S
                find the cheapest edge, add to S'
            find the cheapest edge in S', S-to-t ed
            union(S,t)
```

Because when using union-find set. we do not need too much for-loop operation. For every union-find set operate, it cost constant time. For the main for-loop it will cost $O(V)$.

3.

(a) According to Kruskal Algorithm. Each time we add the minimum edge to the MST. And due to there is no same value of edge in the graph, that means every time the edge is unique. And the result is unique. So MST is unique.

(b)

4.