Minimum Spanning Tree

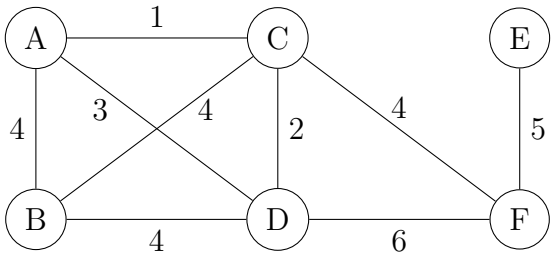

Figure 1: A weighted, connected undirected graph G_1

(a) Prim's algorithm

MST-PRIM(G, w, r)

```
 1  Q = EMPTY
 2  for each vertex v ∈ V
 3        v.key = +INFTY
 4        v.pi = NIL
 5        INSERT(Q, v)
 6  DECREASE-KEY(Q, r, 0)
 7  while Q not EMPTY
 8        u = EXTRACT-MIN(Q)
 9        for each v ∈ Adj[u]
10              if v ∈ Q and w(u, v) < v.key
11                    v.pi = u
12                    v.key = w(u, v)
13  return {(v, v.pi) : v ∈ V - {r}}
```

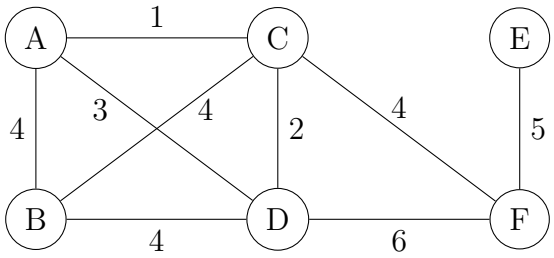

Figure 2: A weighted, connected undirected graph G_1

(b) Kruskal's Algorithm

MST-KRUSKAL(G, w)

```
1  A = EMPTY
2  for each vertex u ∈ V
3       MAKE-SET(u)
4  Sort the edges E in nondecreasing order by w
5  for each edge (u, v) ∈ E, taken in nondecreasing order by w
6       if FIND-SET(u) ≠ FIND-SET(v)
7            A = A ∪ {(u, v)}
8            UNION(u, v)
9  return A
```