

Lecture 3 (Trees)

1 Tree

$T = (V, E)$ is a connected acyclic undirected graph.

2 Minimum Spanning Tree (MST)

Given: A connected undirected graph with n vertices and m edges. A weight function $w : E \rightarrow \mathbb{R}$ is also given.

Task: Find a spanning tree, $T = (V, E_T \subseteq E)$ of G such that, $\sum_{e \in E_T} w(e)$ is minimum.

2.1 Algorithm

```
MST(G) {
    e0 = (x, y) be edge with least weight
    H = G
    H.remove(x)
    H.remove(y)
    H.add(z)
    for v in adj(x) or adj(y) {
        H.addEdge(v, z, min(w(v, x), w(v, y)))
        map(v, z) = (v, w(v, x) <= w(v, y) ? x : y)
    }
    TH = MST(H)
    TG = (V, {e0})
    for e in TH.edges() {
        if e not on z then TG.add(e)
        else TG.add(map(e))
    }
    return TG
}
```