Minimum Spanning Trees

Given an undirected graph **G**=(**V**,**E**), find a graph **G'=(V, E')** such that:

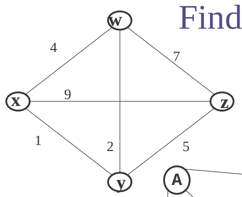
- E' is a subset of E
- |E'| = |V| - 1
- G' is connected
- $\sum_{(u,v)\in E'} c_{uv}$ is minimal

G' is a **minimum spanning tree**.

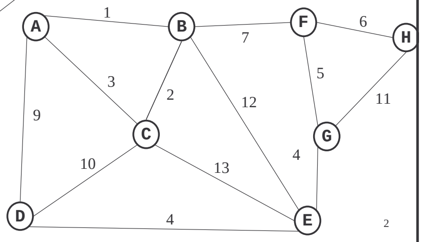
Applications: wiring a house, power grids, Internet connections

1

Find the MST



How many edges in a MST?

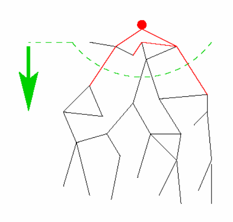


What is the total cost of each MST?

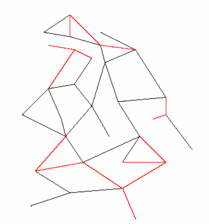
What is a good algorithm?

2

Two Different Approaches



Prim's Algorithm
Almost identical to Dijkstra's

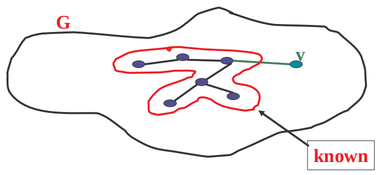


Kruskals's Algorithm
Completely different!

3

Prim's algorithm

Idea: Grow a tree by adding an edge from the "known" vertices to the "unknown" vertices. Pick the ***edge with the smallest weight.***



4

Prim's Algorithm for MST

A *node-based* greedy algorithm

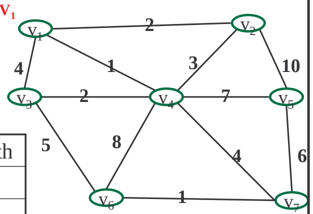
Builds MST by greedily adding nodes

1. Select a node to be the "root"
   - mark it as known
   - Update cost of all its neighbors
2. While there are unknown nodes left in the graph
   a. Select an unknown node *b* with the smallest cost from some *known* node *a*
   b. Mark *b* as known
   c. Add (*a*, *b*) to MST
   d. Update cost of all nodes adjacent to *b*

5

Find MST using Prim's

Start with V_1



V	Kwn	Distance	path
v1			
v2			
v3			
v4			
v5			
v6			
v7			

Order Declared Known:

V_1

6

Prim's Algorithm Analysis

Running time:

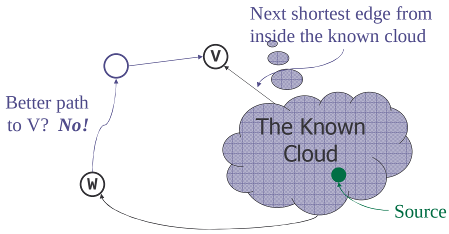
Same as Dijkstra's: O(|E| log |V|)

7

Prim's Algorithm Analysis

Correctness:

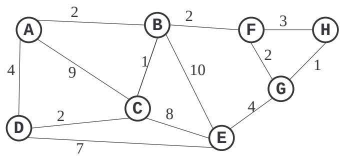
Proof is similar to Dijkstra's



8

Prep For Alternate Algorithm

- Will study alternate MST algorithm (Kruskals)



9