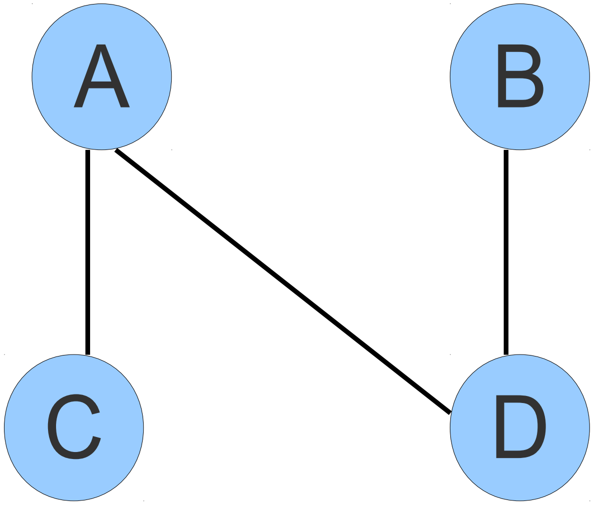
Minimum Spanning Tree

~~Minimum~~ Spanning Tree

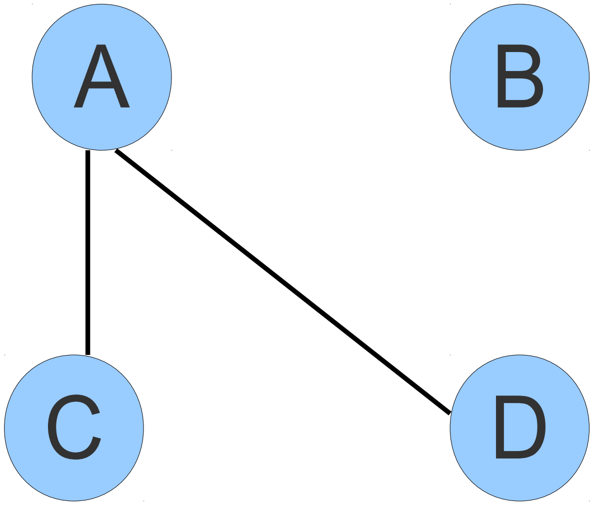
Spanning tree?

- Set of edges of a graph that connects all of its vertices and **is a tree**.



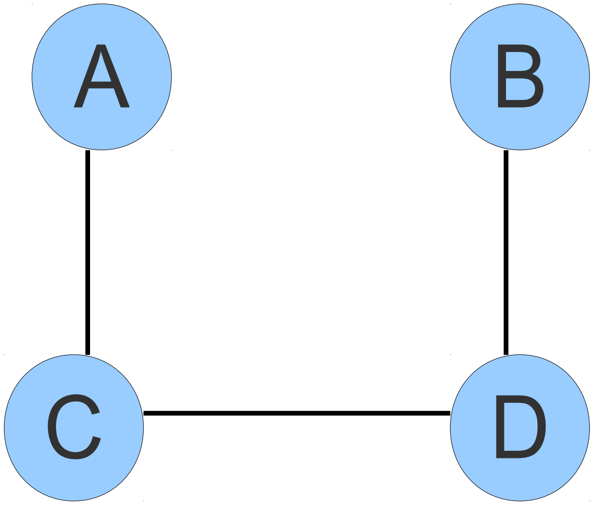
Connected

✓



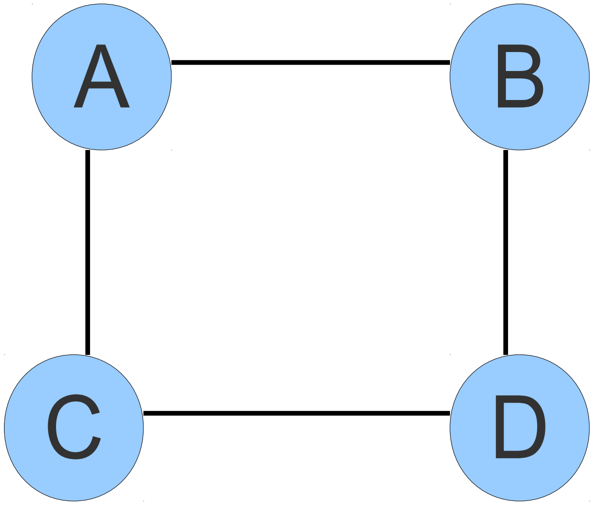
Disconnected

✗



Acyclic

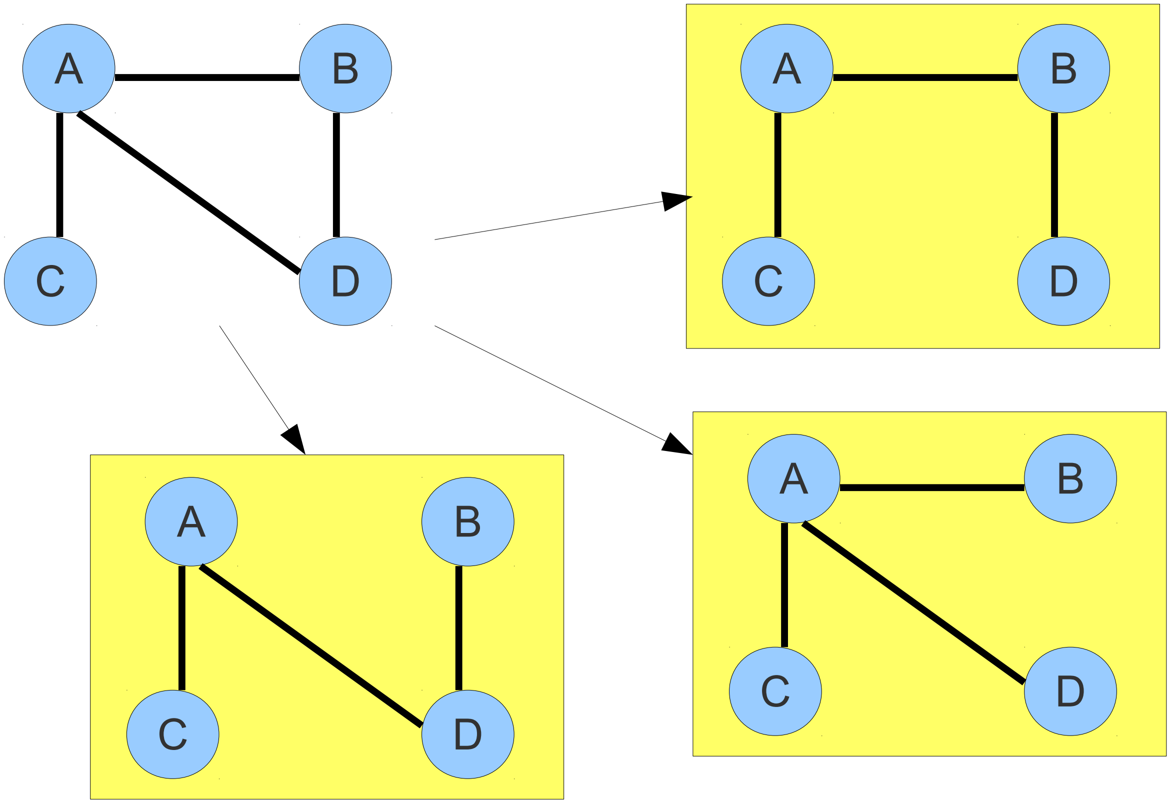
✓



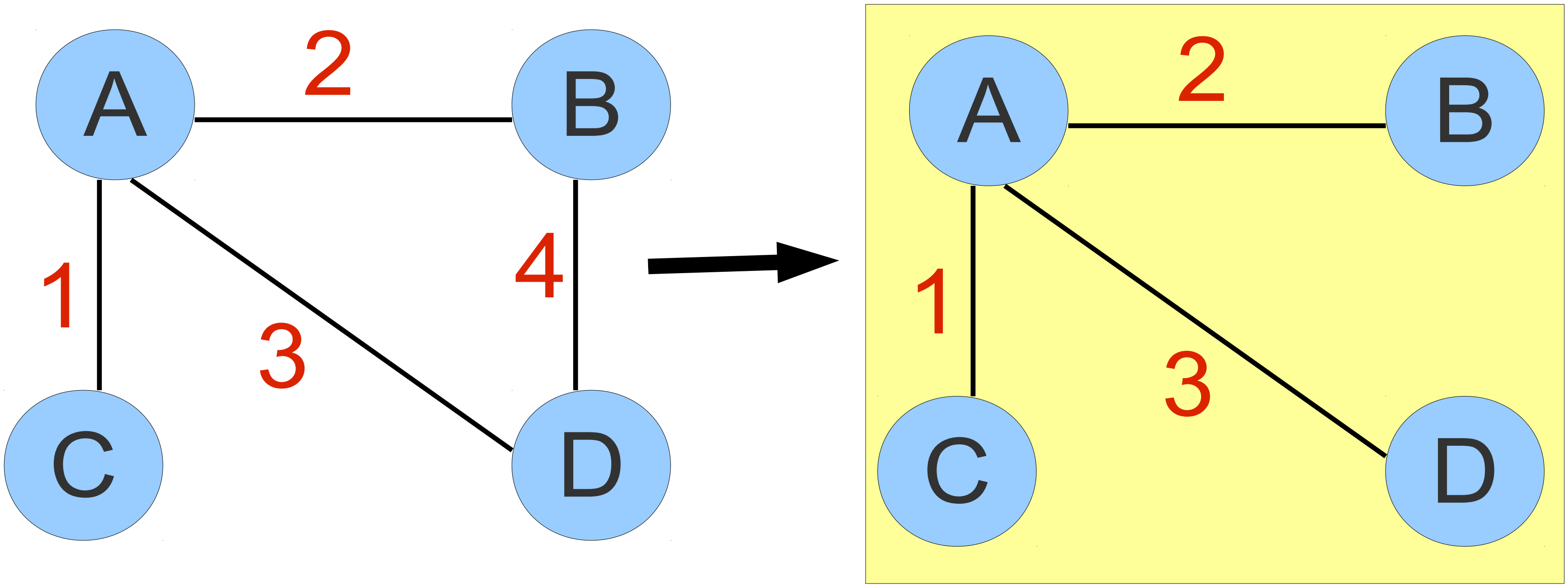
Cyclic

✗

Spanning trees of this graph?



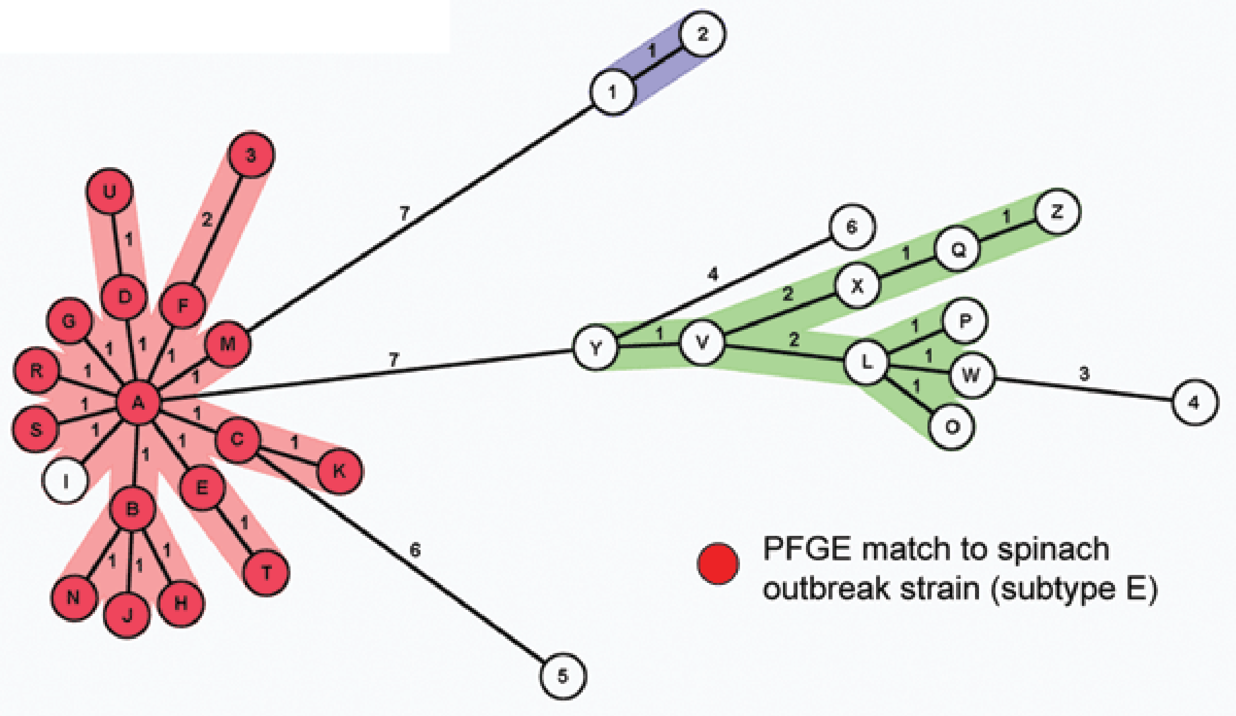
Minimum Spanning Tree (MST)



Total weight = 6

Who cares about MST?

Example – Cluster analysis



Source:
http://www.cdc.gov/eid/content/13/12/1908-G2.htm

Brute force

1. Find total weight of every possible spanning tree of the graph
2. Return spanning tree with lowest total weight

Prim's algorithm

- Greedy
  - Feasible
  - Locally optimal
  - Irrevocable
- Always yields optimal solution

Primitive operation

- Comparison of array elements

Time complexity

- **Step 1: Initialization**

  for each vertex: Theta(|V|)

      set minimum distance from MST to vertex to infinity
      set parent to null

Time complexity

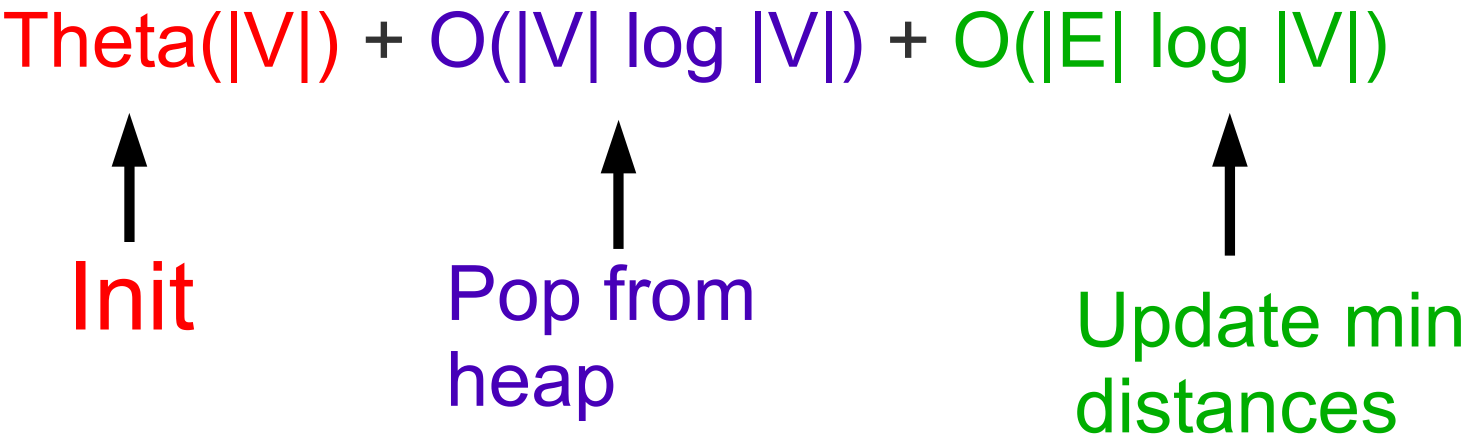
- **Step 2: Building the MST**

  while there are unmarked vertices: Theta(|V|)

  - get next vertex off heap: O(log |V|)
  - and mark it
  - scan each edge adjacent to this vertex: average |E| / |V|

    and MAYBE update the new min distance in heap: O(log |V|)

Time complexity



Theta(|V|) + O(|V| log |V|) + O(|E| log |V|)

Init

Pop from heap

Update min distances

- **O(|E| log |V|)** worst case

Space complexity

- Heap and parents array

  Theta(|V|) extra space

Current state

- Fastest worst case time:

$$O(\ |E|\ \mathbf{a}(\ |E|\ ,\ |V|\ )\ )$$

  - Where **a** is the inverse of Ackermann function

Source:
http://portal.acm.org/citation.cfm?doid=355541.355562

Questions?

Sources

- Our textbook, Section 9.1
- http://www.cs.princeton.edu/algs4/43mst/
- http://en.wikipedia.org/wiki/Prim's_algorithm
- http://video.google.com/videoplay?docid=2755333184412284530
- http://www.cs.ust.hk/~dekai/271/notes/L07/L07.pdf