Draft Due Thursday

- Main purpose – feedback
- A weak draft is better than no draft
- Peer review
  - There will be something for you to fill out
  - What works well? What can be improved?
  - You will be graded on your reviewing, not based on the feedback you are given

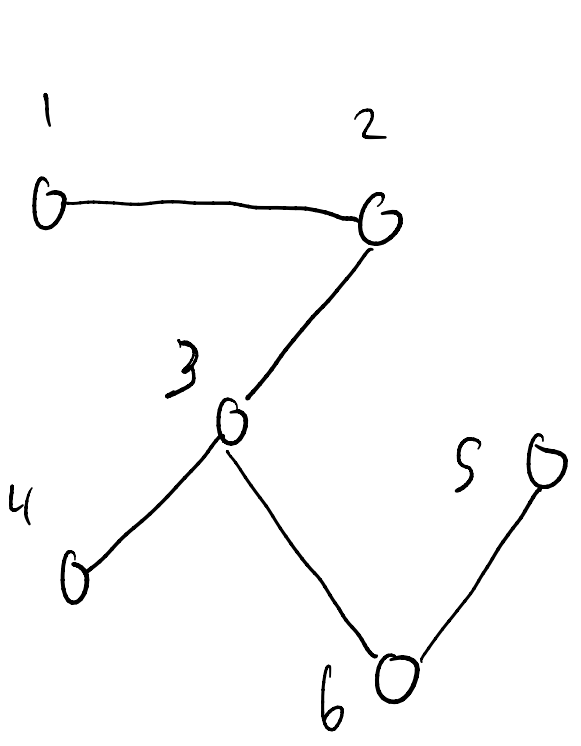
- Abstract - not required now

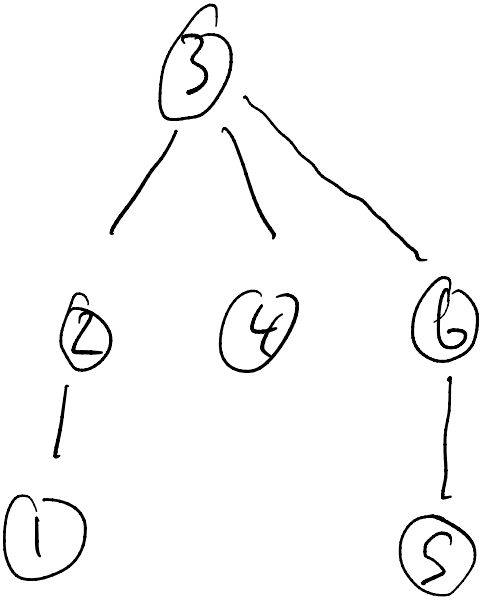
- The more pages the better, but I expect at least 10

- Cite things!

Free Tree (more in appendix B)

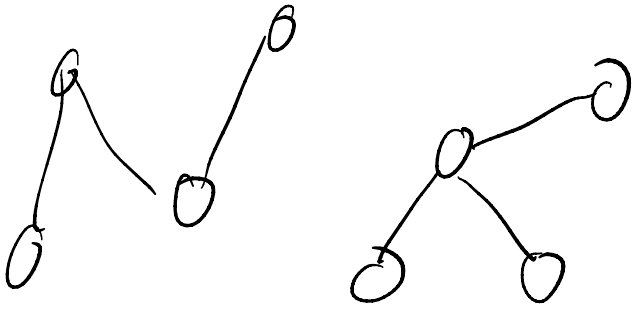
- A connected, acyclic, undirected graph
- Any two vertices are connected by a single simple path
- Removing any edge results in a disconnected graph
- If any edge is added, a cycle is added
- The other trees we have seen so far are "rooted trees"


1
2
3
4
5
6


3
2
4
6
1
5

- Forest

  - A disconnected graph where every connected component is a tree



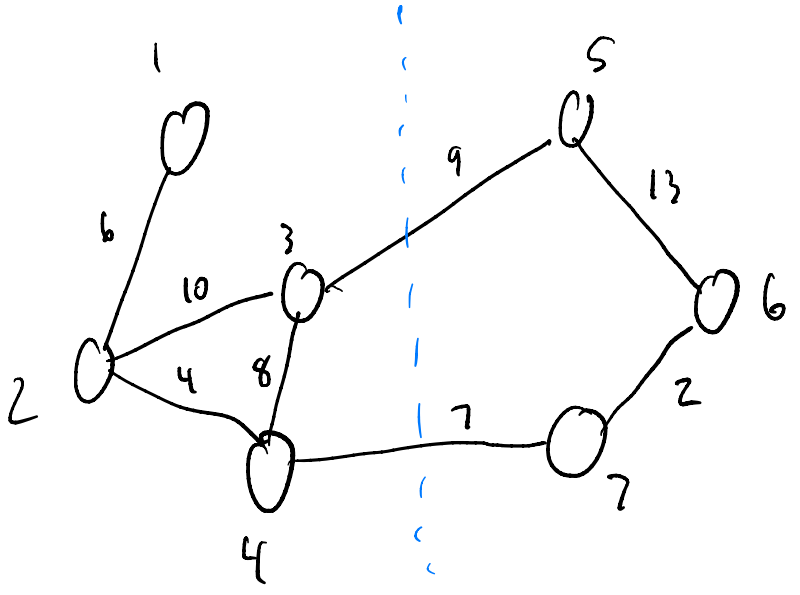
- Graph cut

  - A partition of a graph's vertices into two disjoint subsets

    Cut $(S, S - V)$ partitions V into S and $V - S$

  - Cut set - the set of edges that have one endpoint in each subset

- Edge (u, v) crosses cut $(S, V-S)$ if (u, v) is in the cut set
- A cut respects $A \subset E$ if and only if no edge in A crosses the cut
- An edge is a light edge if its weight is minimum over all edges crossing the cut



Subsets $\{1, 2, 3, 4\}$

and $\{5, 6, 7\}$

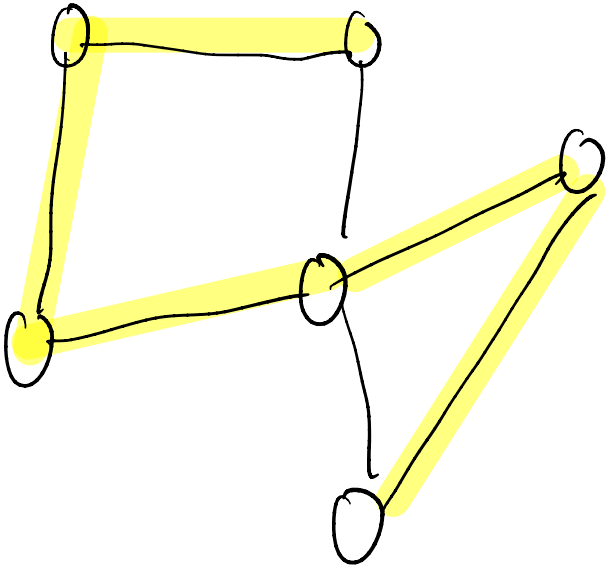
define the cut

Cut set is

$\{(3,5), (4,7)\}$

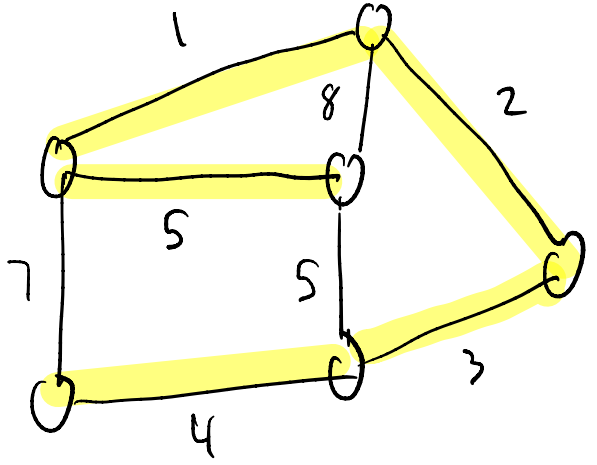
$(4,7)$ is the only light edge

- Spanning tree of a connected undirected graph G
  - A subgraph of G that is a tree and that contains every vertex in G
  - If G' is a spanning tree of G, the edge set of G' is a subset of the edge set of G
  - Contains $|V| - 1$ edges



- Minimum Spanning tree

  - A spanning tree with the minimum sum of the weights of the edges in the spanning tree



2 minimum spanning trees in this graph

- Generic MST algorithm

  - A "safe edge"

    is an edge that

    is guaranteed to

    in some MST

GENERIC-MST(G, w)

$A = \emptyset$

while A is not a spanning tree

  find an edge (u, v) that is safe for A

  $A = A \cup \{(u, v)\}$

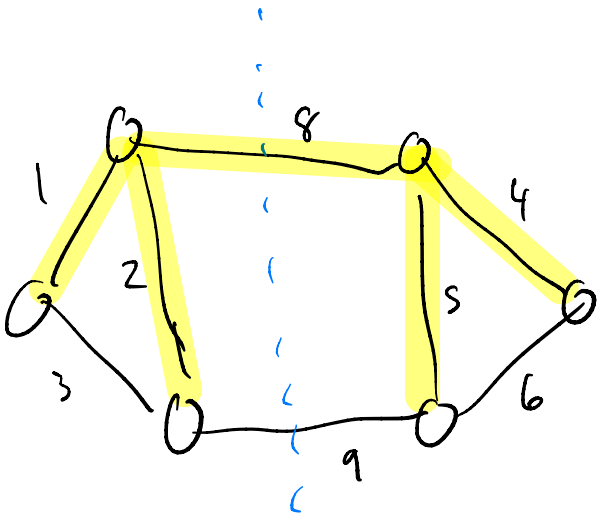
return A

- Loop invariant - A is a subset of a MST

- Theorem 23.1

  - Let A be a subset of some MST, let $(S, V-S)$ be a cut that respects A, and let (u, v) be a light edge that crosses the cut. Then (u, v) is a safe edge for A

- Proof idea
  - An MST must contain a single unique path p from u to v
  - p must cross $(S, V-S)$ at least once
  - $w(u,v) \leq w(x,y)$ for every (x,y) that crosses the cut
  - If a spanning tree contained (x,y), it could be improved by replacing it with (u,v)


1
2
3
4
5
6
8
9

- Kruskal's Algorithm
  - A is a forest
  - Maintain disjoint sets for each component of the growing forest
  - Initially each vertex is its own set
  - Sort the edges by weight
  - For each edge (u, v) in nondecreasing order of weight
    - If u and v are not in the same set (u, v) is safe

```
KRUSKAL(G, w)
  A = ∅
  for each vertex v ∈ G.V
      MAKE-SET(v)
  sort the edges of G.E into nondecreasing order by weight w
  for each (u, v) taken from the sorted list
      if FIND-SET(u) ≠ FIND-SET(v)
          A = A ∪ {(u, v)}
          UNION(u, v)
  return A
```