bipartite

trees

SYSM 6302

CLASS 3

Bipartite Network

$G(W, E)$ $\quad = \quad$ $B(U, V, E)$

nodes, edges $\qquad$ nodes, edges

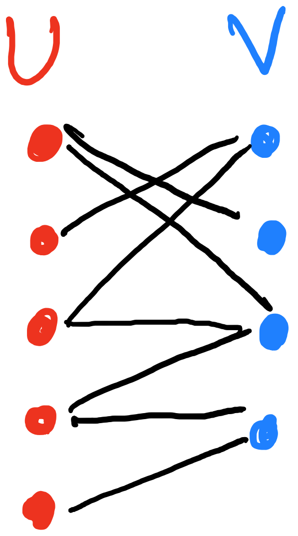
$e \in W \times W$

Nodes can be partitioned into two types: "U" and "V" (these are disjoint)

$W = U \cup V \qquad e \in U \times V$

and

$U \cap V = \emptyset$

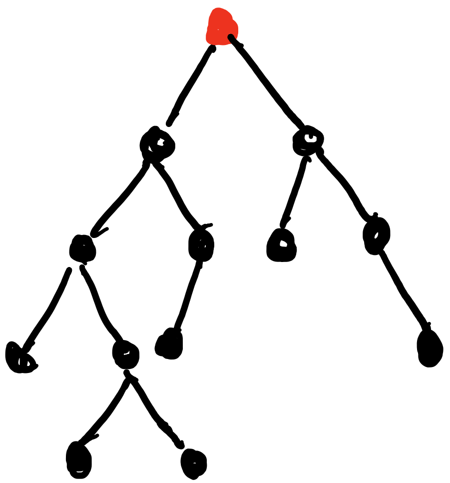


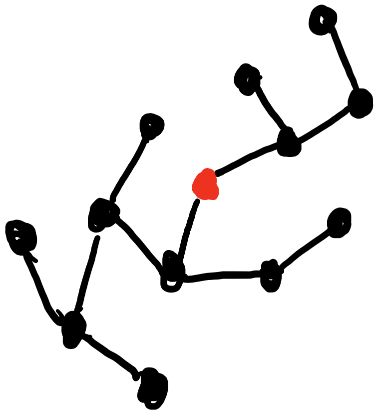
Trees

A tree is a connected, undirected network with no cycles.

every vertex can be reached from every other vertex by a path

"rooted"

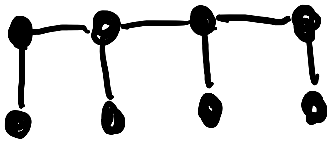




No cycles ⇒ there is only one path between any two nodes

Forest is a network made up of a collection of trees

(disjoint union) ← they don't overlap



⇒ there is at most one path between pairs of nodes (there may be no path)

A (finite) connected graph G with n nodes is a tree if and only if it has $n-1$ edges.

$\Rightarrow$ Suppose G is a tree

By induction

$n=1$

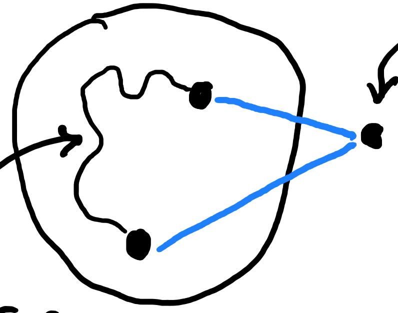
this is a (lone) tree

1 node, $\emptyset$ edges

Suppose Theorem is true for n

$\therefore$ G has n nodes
$n-1$ edges

$\rightarrow$ To make a graph with $n+1$ nodes, need to add one node and some # of edges



new node

0 edges $\Rightarrow$ disconnected

2^+ edges $\Rightarrow$ cycle

$\Rightarrow$ 1 edge!

Since G is a tree always one path b/t pairs of nodes

A (finite) connected graph G with n nodes is a tree if and only if it has $n-1$ edges.

$\Leftarrow$ Suppose G has $n-1$ edges

→ if it is a tree, we are done

→ if it is not a tree, it must contain a cycle

↳ so delete an edge in the cycle & repeat until a tree remains

↳ Call the new graph $\tilde{G}$, which has $n-1$ edges by previous slide

↳ So we started with G with $n-1$ edges, then removed edges until we got $\tilde{G}$ with $n-1$ edges .... Contradiction! G is a tree!

Directed Trees

A directed graph whose underlying graph is a tree.
remove orientation

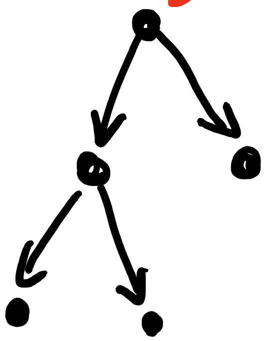
Not quite the same as an acyclic graph (the lab has you think about this)

directed acyclic graphs $\overset{?}{\subseteq}$ directed trees

OR

directed trees $\overset{?}{\subseteq}$ directed acyclic graphs

root has a path to every node



rooted directed tree