Math 4990: Trees

11/10

Ch. 10

Reminders:
- HW #4 is due today.
- Midterm #2 posted, due in a week (11/17)

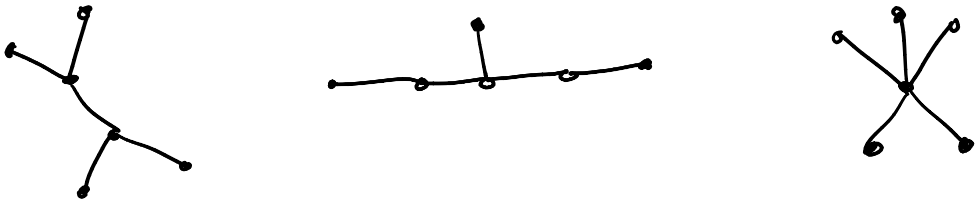
Last class we started graph theory. We considered various problems about walking around on a graph. Central to these problems was the notion of connectivity. We will study connectivity in more detail today by investigating minimally connected graphs, which are called trees.

Thm Let G be a graph. TFAE:

1) G is minimally connected, i.e., G is connected but the removal of any edge would disconnect G.

2) G is connected and contains no cycles.

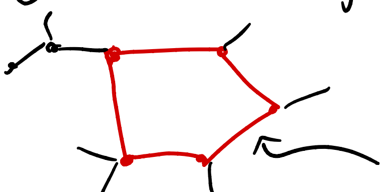
A graph satisfying either of these equiv. conditions is called a **tree**. Some trees on 6 vertices are:



Pf of thm: Let G be a connected graph. We need to show:

G has an edge we can remove + stay connected $\iff$ G has a cycle.

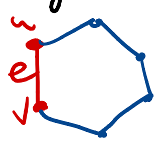
[$\Leftarrow$] Suppose G has a cycle:



(O + D are cycles)

Then we can remove any edge of the cycle without changing connectivity of graph.

[$\Rightarrow$] Suppose G has an edge $e = \{u, v\}$ we can remove + stay connected:



Then there has to be another path from u to v not using e, which forms a cycle with e. $\blacksquare$

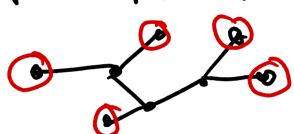
Feels like:

- if G has too few edges, it can't be connected
- if G has too many edges, it will have a cycle

So trees are "goldilocks graphs" that have just the right # of edges. In fact:

Thm A tree with n vertices has $n-1$ edges.

In order to prove this theorem, we need a lemma. A leaf of a tree is a vertex of degree $= 1$. ↳



Lemma Any tree ($w \geq 2$ vertices) has a leaf.

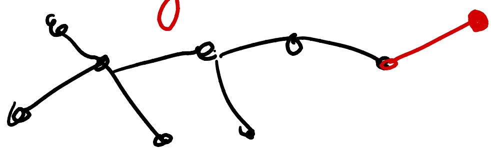
Pf: Start at any vertex of our tree T and keep walking to new vertices along edges we haven't used:

$$v_0 \to v_1 \to \cdots\cdots \to v_k$$

We don't have a cycle, so can never revisit a vertex. Eventually we get stuck: at a leaf. ☒

Remark: Can show that actually there must be at least two leaves.

Pf of thm: By lemma any tree on n vertices can be obtained from a tree on $n-1$ vertices by appending a leaf: (Think about this)



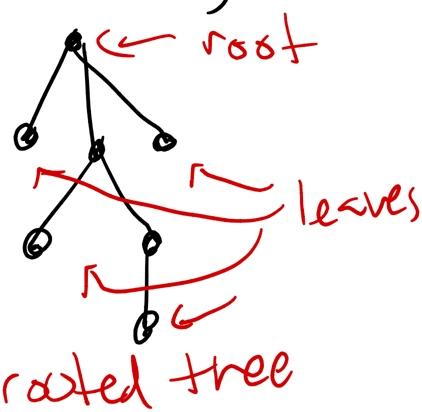
Thus, the theorem follows by induction, with the base case being tree w/ 1 vertex and zero edges: • $\boxtimes$

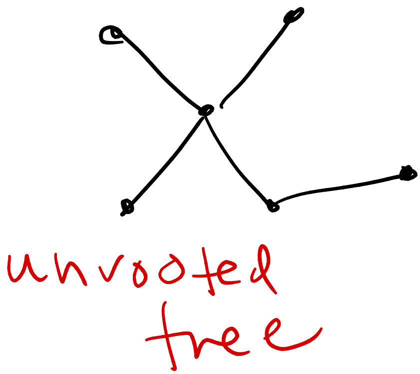
In fact, can show:

Thm Let G be a graph on n vertices. Then any 2 of these implies the 3rd:

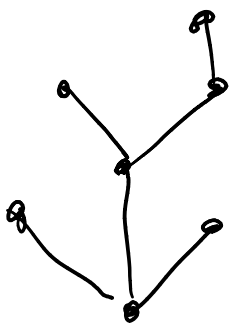
- G is connected.
- G has no cycles.
- G has $n-1$ edges.

Why are these called "trees"? Arboreal terminology makes most sense for rooted trees: a rooted tree is a tree where we've chosen a special root vertex, which we draw at the top, w/ other vertices branching down from it:

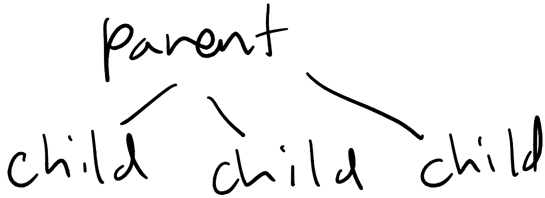




The picture makes most sense if we draw it upside-down:

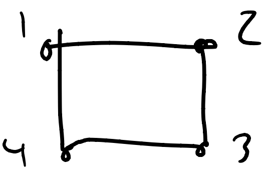


But traditional to draw it with root at top and use family tree terminology:

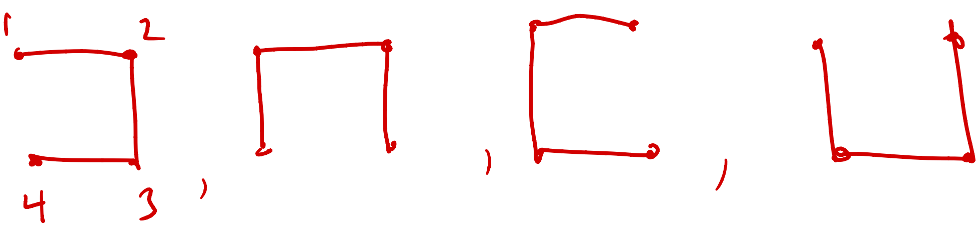


Also descendant + ancestor, etc. This rooted tree perspective is very useful for studying trees...

Let G be a graph. A spanning tree of G is a subgraph that's a tree containing all the vertices of G.

e.g. $G =$ 

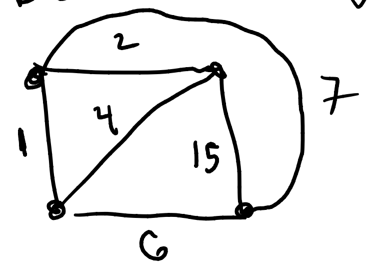
spanning trees:



Prop. G has a spanning tree $\iff$ G is connected.

If G represents a map, then reasonable to think it comes with an edge-weight function $w: E \to \mathbb{R}$ representing cost or distance between vertices:

e.g.

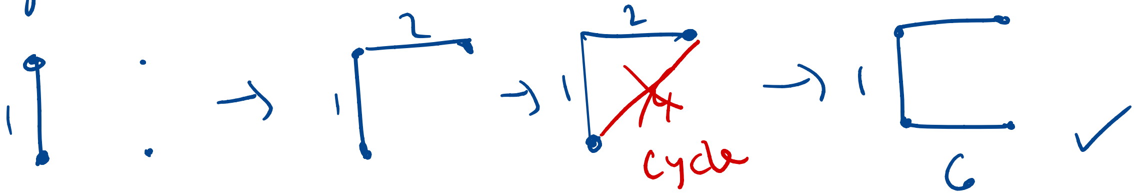
$G =$ 

Problem: How to find a minimum cost spanning tree of G? (Think of a telecommunications or airlines network that wants to be connected)

Answer: Be greedy! Use Kruskal's algorithm:

- Keep adding minimum cost edge we haven't added, unless it creates a cycle! (then skip it...)

e.g.:

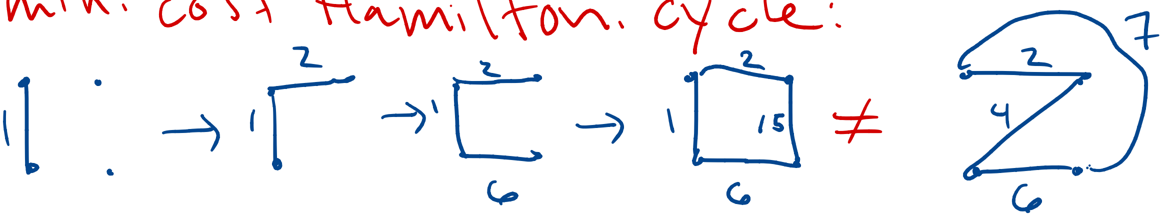


Thm Kruskal's algorithm works, i.e., finds the min. cost spanning tree.

Pf: See the book... ∎

Note: Greedy algorithm does not work for all problems (something special abt trees).

E.g., greedily choosing will not produce the min. cost Hamilton. cycle:



Actually, this is the famous Travelling Salesman Problem for which no good algorithm is known (big problem in comp. sci.).

Q: How many trees on n vertices are there?

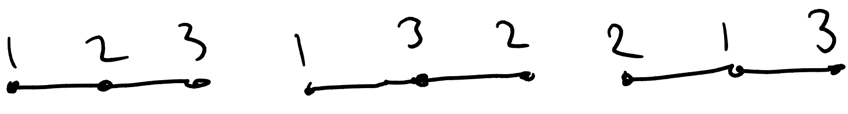
e.g. $n = 1$

1

$n = 2$

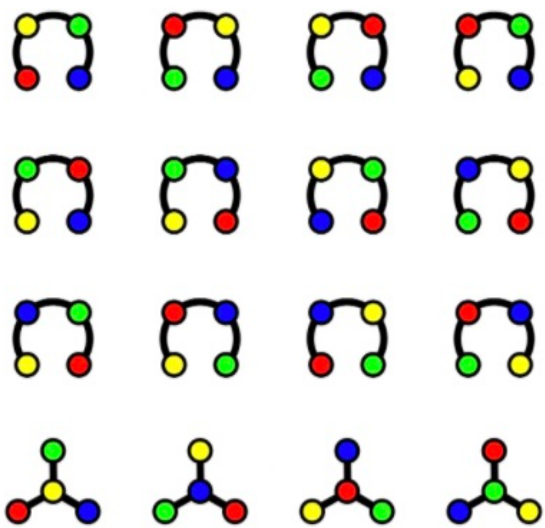
1

$n = 3$



3

$n = 4$



16

$n = 5$? ? ?

Notice any pattern? ...

Thm (Cayley's formula) There are $\boxed{n^{n-2}}$ trees on n (labelled) vertices.

Very beautiful formula! Many proofs:

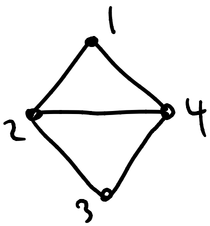
- Generating functions ('Lagrange inversion')
- Bijective proofs:
  - Prüfer code
  - A. Joyal's proof ← (see the book)
  - J. Pitman's proof
- Linear algebra pf: Matrix-Tree Thm

The Matrix-Tree Theorem gives a formula for # spanning trees of any graph G. Set:

$$A_G := \underset{\text{matrix}}{\text{adjacency}} = (a_{ij}) \text{ w/ } a_{ij} = \begin{cases} 1 & \text{if } v_i + v_j \text{ adjacent} \\ 0 & \text{otherwise} \end{cases}$$

$$L_G := \text{Laplacian matrix of } G = \begin{pmatrix} \deg(v_1) & & & 0 \\ & \deg(v_2) & & \\ & & \ddots & \\ 0 & & & \deg(v_n) \end{pmatrix} - A_G$$

$\tilde{L}_G :=$ reduced Laplacian $=$ remove last row + last col. of L_G

e.g. $G =$ 

$$A_G = \begin{array}{c c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

$$L_G = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 3 & -1 & -1 \\ 0 & -1 & 2 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix} \qquad \tilde{L}_G = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Thm (Kirchoff's Matrix Tree Thm)

$\#$ spanning trees $(G) = \det(\tilde{L}_G)$.

e.g. $\det \begin{bmatrix} 2 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 2 \end{bmatrix} = 2\cdot 3\cdot 2 - 2 - 2 = 8.$

Pf: See book ... $\blacksquare$

To get Cayley's formula from M.T. Thm need to evaluate determinant of

$$\tilde{L}_{K_n} = \det \begin{bmatrix} n-1 & -1 & -1 & \cdots & -1 \\ -1 & n-1 & -1 & \cdots & -1 \\ & & n-1 & & \vdots \\ \vdots & & & \ddots & \\ -1 & -1 & \cdots & -1 & n-1 \end{bmatrix}$$

(e.g. $\det \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} = 3$)

Now let's take a break...

and when we come back

we'll do some problems

about thecs on the worksheet

in breakout groups.