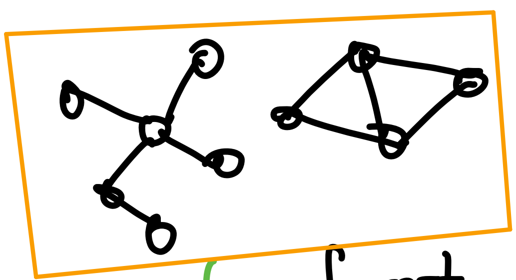
Trees (Chapter 2)

These are the simplest graphs to understand, but also form the backbone for understanding all graphs.

DEFINITION: A multigraph $G=(V,E)$

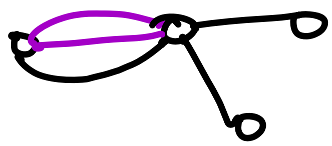
- with no cycles is called **acyclic** or a **forest**,
- a **tree** if it's a connected forest

NON-EXAMPLES:

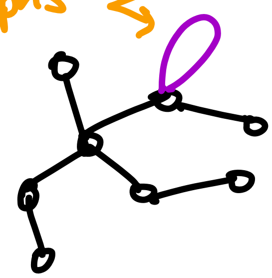


not a forest
(one of its components has a cycle)

forests are always simple graphs

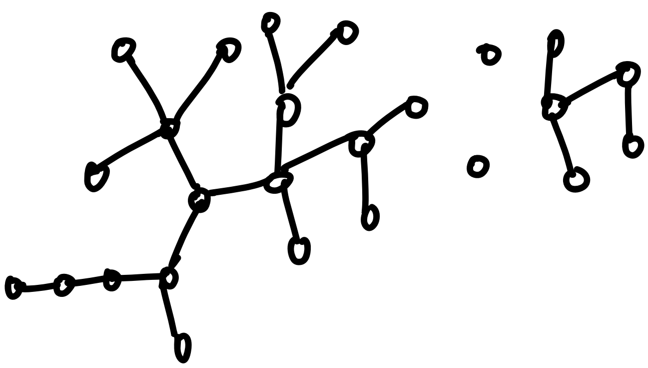


not a forest

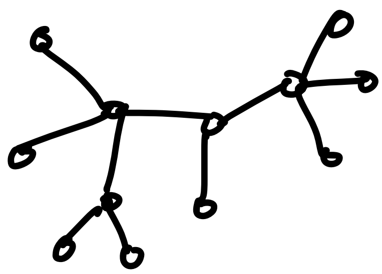


not a forest

EXAMPLES:



a **forest**



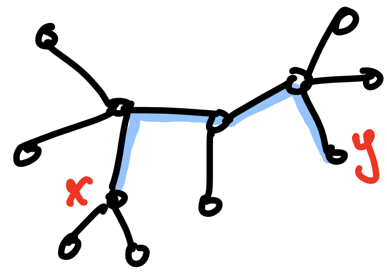
a **forest**,
and also a **tree**

There are other useful ways to define trees, forests.

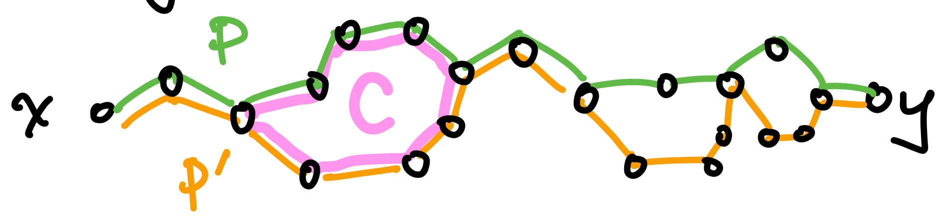
PROPOSITION: A multigraph $G=(V,E)$

(a) is a forest $\Leftrightarrow$ $\forall x,y \in V$, $\exists \leq 1$ path from x to y

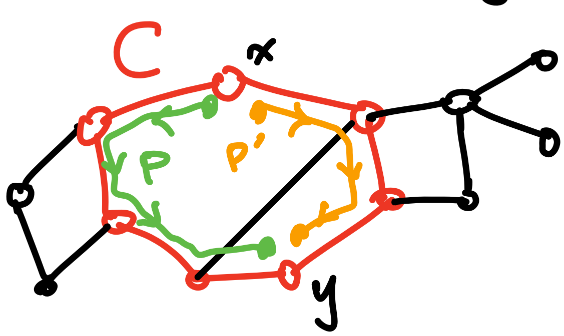
(b) is a tree $\Leftrightarrow$ $\forall x,y \in V$, $\exists$ exactly 1 path from x to y.



proof:
(a) ($\Longrightarrow$): Two distinct paths $P \neq P'$ from x to y will contain a cycle in $P \cup P'$, namely between their first divergence to their next re-convergence:



($\Longleftarrow$): A cycle C gives two paths $P \neq P'$ between any two of its vertices x,y:



(b) follows from (a) since the definition of G connected is having ≥ 1 path from x to y $\forall x,y \in V$ ▨

Two more useful characterizations of trees:

PROPOSITION: For a multigraph $G=(V,E)$

(a) G is a tree $\iff$ G is minimally connected:
G is connected, but $\forall e \in E$
the deletion $G-e=(V, E-\{e\})$
is disconnected

(b) G is a tree $\iff$ G is maximally acyclic/forest:
G is acyclic, but $\forall x,y \in V$
the addition $G \cup \{x,y\} = (V, E \sqcup \{x,y\})$
contains a cycle.

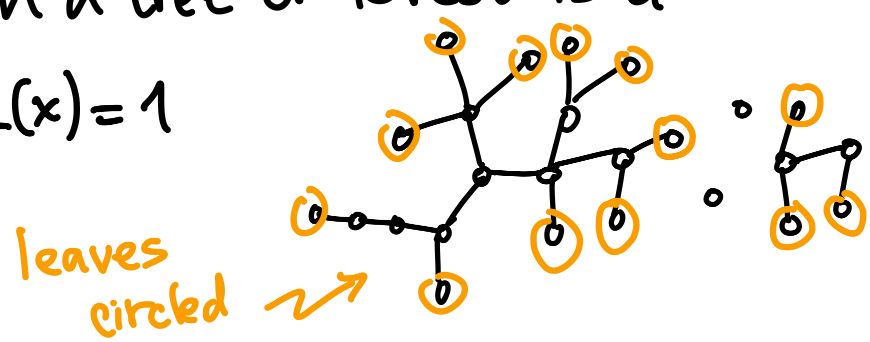
ACTIVE LEARNING

Prove the above PROPOSITION, which really means proving 4 implications:

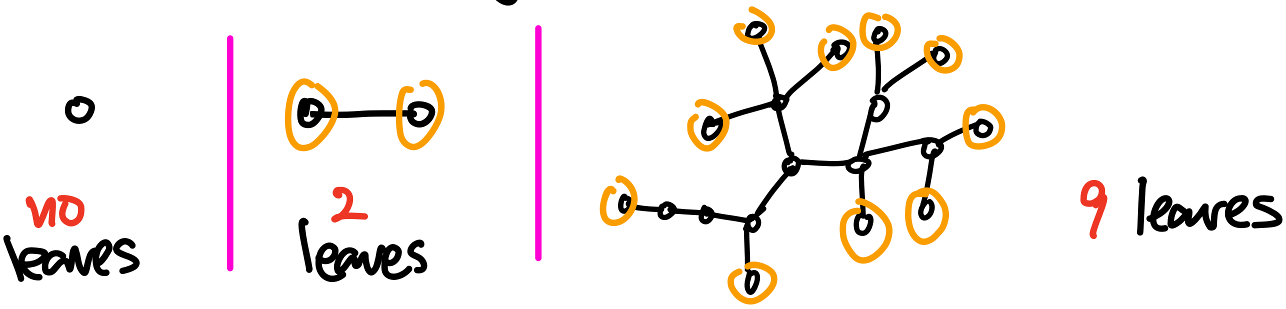
(a) $(\Rightarrow)$, $(\Leftarrow)$
(b) $(\Rightarrow)$, $(\Leftarrow)$

A common proof technique for trees is leaf induction.

DEFINITION: A leaf in a tree or forest is a vertex $x \in V$ with $\deg_T(x)=1$

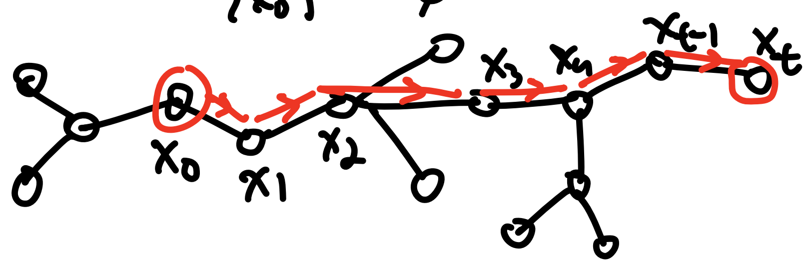


PROPOSITION: Every tree $T=(V,E)$ with at least one edge has at least 2 leaves.



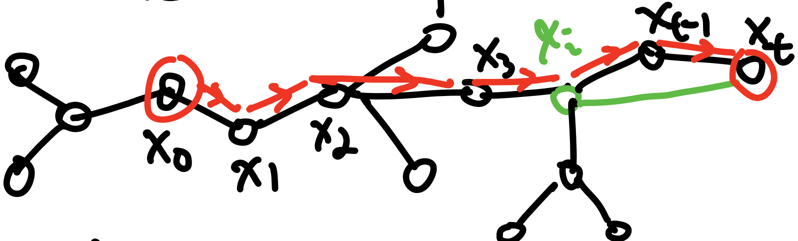
Proof: Start at any vertex $x_0 \in V$ and walk along edges to new (unvisited) vertices $x_1, x_2, \ldots$ until you get stuck at some vertex x_t.

(Note that $t \geq 1$, i.e. $x_t \neq x_0$, else you got stuck at x_0, which forces $T = (\underset{\{x_0\}}{\underset{\shortparallel}{V}}, \underset{\emptyset}{\underset{\shortparallel}{E}})$ with no edges.)



CLAIM: x_t is a leaf in T, otherwise it has another neighbor $y \neq x_{t-1}$ as no parallel edge to x_{t-1}, no loop on x_t.

So y was previously visited, say $y = x_i$ for some $0 \leq i \leq t-2$, and one has two paths from x_i to x_t:



Having found one leaf, use it as x_0 to repeat and find a 2nd ∎

Here's an example of a proof by leaf induction.

COROLLARY:
Trees $T=(V,E)$ have $|E|=|V|-1$.

proof: Induct on $|V|$.

Base case: $|V|=1$. Then $E=\emptyset$
so $|E|=0=|V|-1$ ✓.

Inductive step:
Given T with $|V|\geq 2$, connectivity implies it has at least one edge, and hence it has a leaf vertex x, say with unique neighbor y in T. **CLAIM:** $\hat{T}=(V-\{x\}, E-\{x,y\})$

This is because $\hat{T}$ is still acyclic, and still connected because the unique path in T from y to any $z\in V-\{x\}$ could not pass through x (else its next step is y), so it persists in $\hat{T}$.

Hence induction applies to $\hat{T}$,
and $|E(\hat{T})|=|V(\hat{T})|-1$
so $|E(T)|=1+|E(\hat{T})|=|V(\hat{T})|=|V(T)|-1$. ∎

REMARK: Here's another useful characterization of trees, that is not too hard to prove, but we won't prove it here.

PROPOSITION: For a multigraph $G = (V, E)$, any two of these three properties together implies the third (and hence implies that G is a tree):

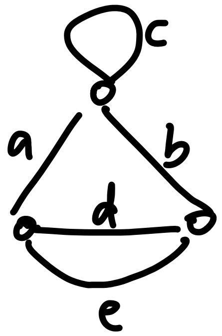
(i) G is connected.

(ii) G is acyclic / forest.

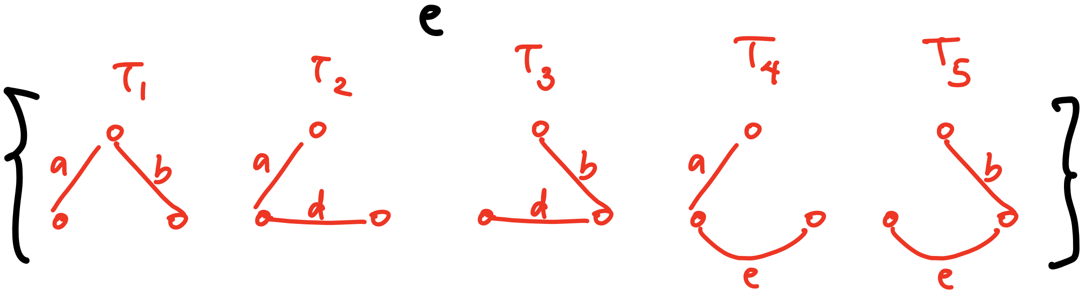
(iii) $|E| = |V| - 1$.

Minimum cost spanning trees

DEFINITION: In a multigraph $G=(V,E)$, a spanning tree for G is a subset $T\subset E$ for which (V,T) is a tree.

EXAMPLE: $G =$ 

has 5 spanning trees:

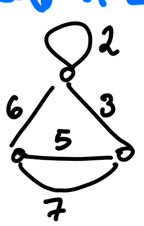


A spanning tree is a way to minimally connect V, and one can even find a cheapest tree quickly if one has a cost function

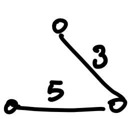
$$c: E \to \mathbb{R}_{\geq 0}, \qquad e \mapsto c(e)$$

where the cost of T is $c(T) := \sum_{e\in T} c(e)$. This is called a minimum cost spanning tree (MST) for G, c.

EXAMPLE

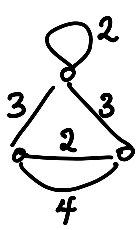


has a unique MST:

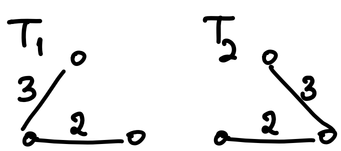


$= T$

$c(T)=3+5=8$



has two MST's:



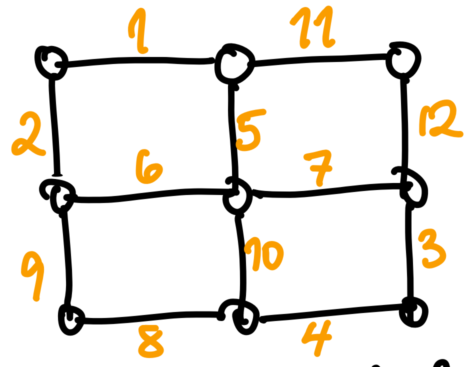
$c(T_1)=2+3=5=c(T_2)$

There are several fast algorithms to find an MST for G, c.

Two greedy ones are Kruskal's (1956) and Prim's (1957) algorithms:

Given $G=(V,E)$ and $c: E \to \mathbb{R}_{\geq 0}$,

e.g. $(G, c) =$



both algorithms build a sequence of forests $F_i \subset E$

$$\phi = F_0, F_1, F_2, F_3, \ldots, F_{|V|-2}, F_{|V|-1} \leftarrow \text{an MST}$$

where $|F_i| = i$, by adding in one edge at a time,

so $F_i = F_{i-1} \sqcup \{e_i\}$ (so taking $|V|$ steps).

They choose e_i to be any one of the edges e that achieve the minimum cost $c(e)$ among these sets of edges:

Kruskal: $\{ e \in E : F_{i-1} \sqcup \{e\} \text{ is acyclic} \}$

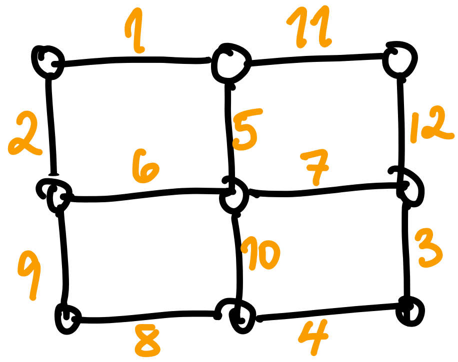
Prim: $\{ e \in E :$ $\{$ $F_{i-1} \sqcup \{e\}$ is acyclic

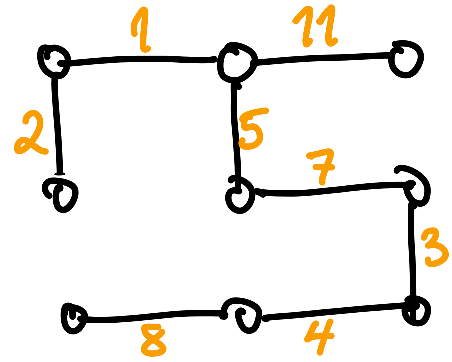
AND

(V, F_{i-1}) has only isolated vertices and one tree as connected components $\}$ $\}$

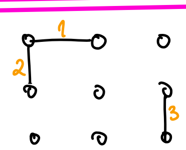
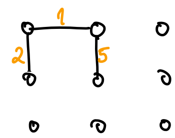
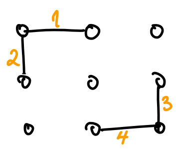
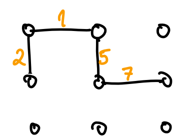
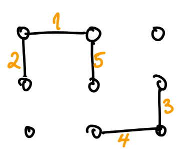
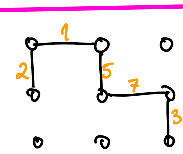
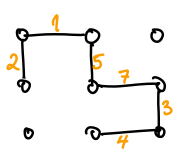
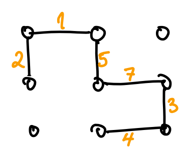
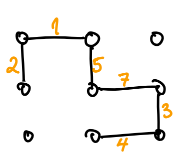
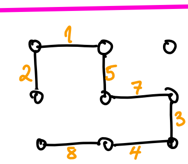
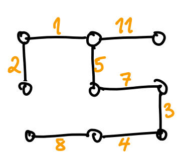
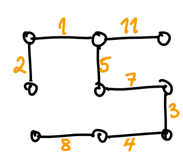
equivalently: $e = \{x,y\} \notin F_{i-1}$ and has at most one of x, y isolated in F_{i-1} when $i \geq 2$.

EXAMPLE:

For 

Kruskal, Prim both produce $T = F_8 =$ 

and both start with $F_0 = \emptyset$, $F_1 = \{1\}$, $F_2 = \{1,2\}$, but then ...

Kruskal:		Prim:
	F_3	
	F_4	
	F_5	
	F_6	
	F_7	
	$F_8 = T$	

THEOREM:

(a) Both Kruskal's and Prim's algorithms find an MST.

(b) If $c: E \to \mathbb{R}_{\geq 0}$ has $c(e) \neq c(e')$ $\forall e \neq e'$ in E, then $\exists$ a unique MST (found by both algorithms).

proof: Let T_{min} be any MST for (G, c)
T_{greedy} be either the tree produced by Kruskal or by Prim.

We will show the following:

> If $T_{min} \neq T_{greedy}$, then
>
> $\exists$ two edges $\begin{cases} e \in T_{min} \setminus T_{greedy} \\ e_i \in T_{greedy} \setminus T_{min} \end{cases}$
>
> for which $T = (T_{min} \setminus \{e\}) \sqcup \{e_i\}$ is an MST sharing more edges with T_{greedy},
>
> that is, $|T \cap T_{greedy}| > |T_{min} \cap T_{greedy}|$.

ACTIVE LEARNING:

We claim that this boxed assertion would prove both (a) and (b). Explain why.

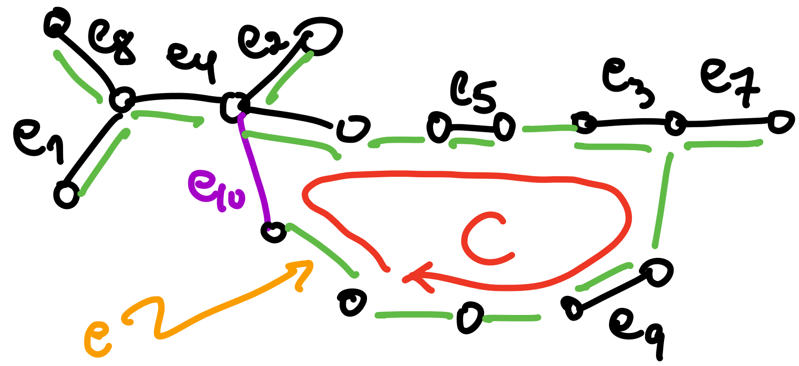
So let's show the boxed assertion.

First choose $i \in \{1,2,\ldots,M-1\}$ to be the earliest stage where the greedy algorithm (Kruskal or Prim) chose $e_i = F_i \setminus F_{i-1}$ with $e_i \notin T_{min}$.

Since T_{min} is a tree on vertex set V, it is maximally acyclic and so $T_{min} \sqcup \{e_i\}$ contains a cycle C.

e.g. $i=10$ $F_{i-1} = F_9 = \{e_1, e_2, \ldots, e_9\} \subset T_{min} \cap T_{greedy}$

but $e_{10} \in F_{10} - T_{min}$



Since T_{greedy} is acyclic, $C \not\subset T_{greedy}$, and so $\exists$ some edge $e \in C \setminus T_{greedy}$.

Choose this as e to define $T = (T_{min} \setminus \{e\}) \sqcup \{e_i\}$

In fact, in Prim's algorithm, since e_i has at least one endvertex non-isolated in F_{i-1}, one can choose e to also have this property.

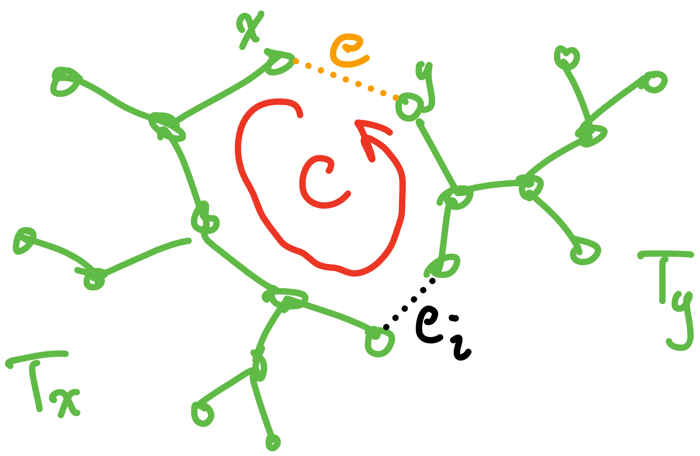
We CLAIM that $T := (T_{min} \setminus \{e\}) \sqcup \{e_i\}$ is again a spanning tree. Here is a (sketch) proof.

$T_{min} \setminus \{e\}$ is a forest with two tree components, $\{e\} = \{x,y\}$

the one (V_x, T_x) containing x,

and the one (V_y, T_y) containing y.

Since $T_{min} \sqcup \{e_i\} \supset C \supset \{e_i\}$, one knows that e_i has one endpoint in T_x and one in T_y,



so that $T = (T_{min} \setminus \{e\}) \sqcup \{e_i\}$ is connected,

and acyclic,

hence a tree.

One also knows that $c(e) \geq c(e_i)$ by the definitions of the greedy algorithms (since e competes with e_i at the i^{th} step).

Hence
$$c(T) = c((T_{min} - \{e\}) \sqcup \{e_i\})$$
$$= c(T_{min}) - c(e) + c(e_i)$$
$$= c(T_{min}) - \underbrace{(c(e) - c(e_i))}_{\geq 0}$$
$$\leq c(T_{min}).$$
So T is another MST. $\blacksquare$