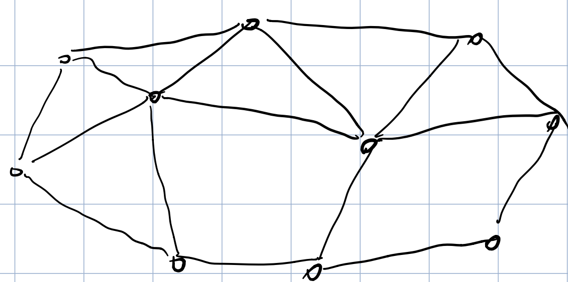
$\boxed{\text{17 Feb 2021}}$ Minimum Spanning Tree (§4.5)

<u>History.</u> Boruvka had to figure out a plan for cheaply connecting the towns of Moravia with electric power lines.

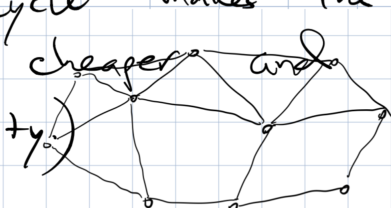


Undirected, connected graph G.

For each edge e, a number $\text{cost}(e) > 0$.

<u>Goal:</u> Find a connected subgraph of minimum total edge cost.

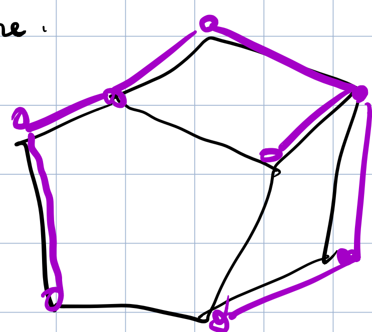
<u>Observation.</u> The optimal solution will not contain a cycle. (Removing any edge from the cycle makes the subgraph strictly cheaper and preserves connectivity.)



Def A connected graph is called a **tree** if it has no cycles.
In general a graph without cycles is called a **forest** whether or not it's connected.

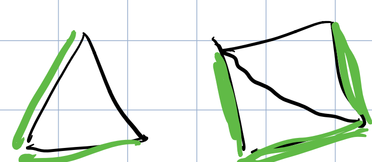
A **spanning tree** or **spanning forest** of a graph G is a subgraph H that is a forest and such that the vertex sets of the connected components of G and H are the same.

G



H = spanning tree of G.

G'



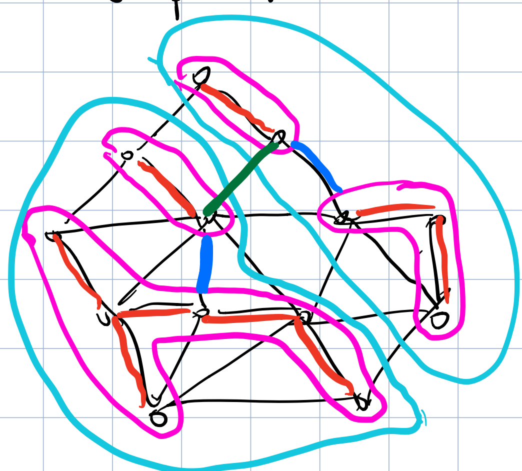
H' = spanning forest of G'.

The problem above can be rephrased as:
given undirected G and $\text{cost}(e) > 0$ $\forall e$, find a spanning tree of min total edge cost.

Def. A graph G is <u>connected</u> if every pair of vertices can be joined by a path in G.

Equivalently: G is connected ~~iff~~ if and only if there is no partition of its vertices into nonempty sets A, B with no edges between them.

We say u and v are in the same <u>connected component</u> of G if there is a path from u to v. The vertex set of every graph can be partitioned into disjoint connected components.

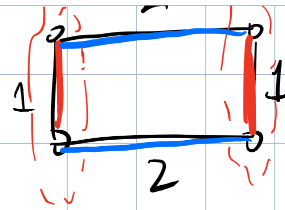


<u>Boruvka's Alg.</u>

For every vertex find its nearest neighbor. Join them together.

i.e., cheapest

While the graph remains disconnected, repeat the operation of joining each connected component to



(Inconsistent tie breaking could lead to a cycle forming.)

another connected comp't using the cheapest available edge.

Some questions:

1. Does this even produce a tree?
2. Is that tree the min spanning tree?

We'll see the answer to both is yes.

For the remainder of the lecture assume edge costs are all distinct.
(This assumption is essentially without loss of generality because we can break ties consistently by assigning arbitrary distinct priority values to edges before starting to run the alg. See Chapt 4.5 for a lengthier explanation.)

Why does Boruvka's algorithm work?

Lemma. (Cut Property) If G is a graph with distinct edge costs and A is a non-empty proper subset of the vertices (at least one vtx belongs to A, at least one does not) then the min cost edge from A to its complement V-A must belong to <u>every</u> min spanning tree.

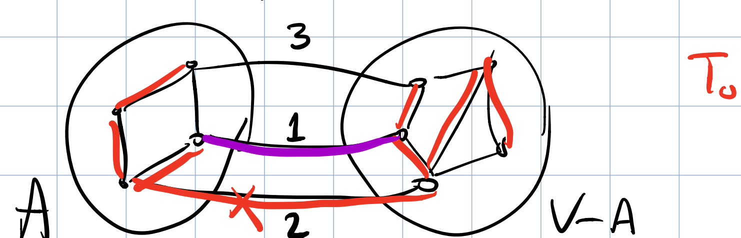
<u>Proof.</u> By "exchange argument." Suppose T_0 is a spanning tree that <u>doesn't</u> contain the min cost edge from A to V-A. We will show how to modify T_0 into another spanning tree T_1 that is cheaper. It will follow that T_0 is not a MST. So

$$\left(\begin{array}{l}T_0 \text{ doesn't contain} \\ \text{min-cost edge in} \\ \text{the cut}\end{array}\right) \Longrightarrow \left(\begin{array}{l}T_0 \text{ is not} \\ \text{a MST}\end{array}\right)$$

Contrapositively

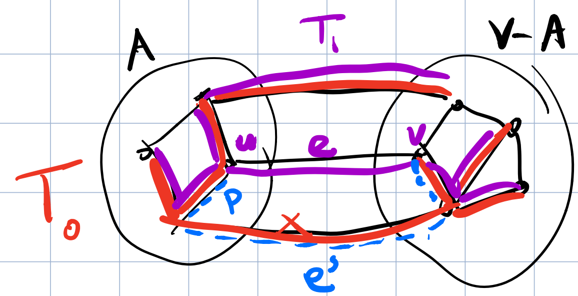
$$\left(T_0 \text{ is a MST}\right) \Longrightarrow \left(\begin{array}{l}T_0 \text{ contains min cost} \\ \text{edge in the cut}\end{array}\right)$$

How to modify T_0 into T_1?



Logical idea: delete the edge of T_0 that goes from A to $V-A$. Replace it with the min cost edge.

Snag: T_0 may contain more than one edge from A to $V-A$.



More careful exchange:

Let $e=(u,v)$ be the min cost edge from A to $V-A$.

If $e \notin E(T_0)$ then T_0 must contain a path P from u to v.

Then P starts in A, ends in $V-A$, so it contains at least one edge e' from A to $V-A$.

Let e' be any such edge.

Delete e' from T_0.

Now u, v are in separate components of $T_0 - e'$.

Rejoin these components by adding e, to form new spanning tree T_1.

$\text{cost}(T_1) < \text{cost}(T_0)$ because we removed e' which costs more than e, and we inserted e, leading to net reduction in cost.

<u>Correctness of Boruvka:</u> The whole alg. performs a sequence of steps each joining a component A to its complement $V-A$ using the cheapest available edge. By the Lemma above, that never introduces an edge that doesn't belong to <u>every</u> MST.

<u>Friday:</u> Continue with MST. (Sections 4.5 and 4.6...)