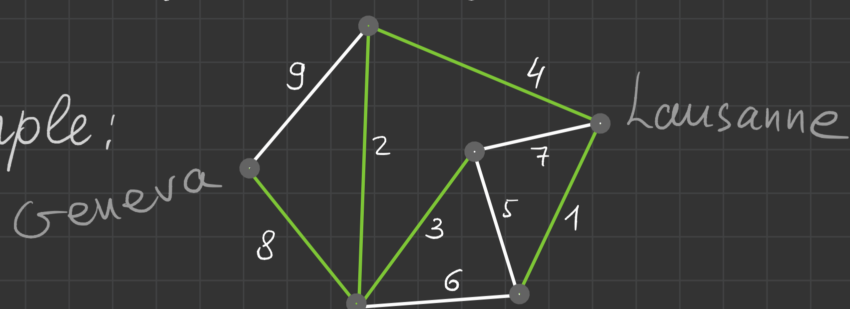
Minimal spanning tree problem

Definition: A *weighted graph* is a graph in which each edge is assigned a numerical weight.

We define the *weight of a graph* as the sum of weights of all its edges

Problem: Find a minimum weight spanning tree T for a given weighted connected graph G

Example:



Greedy algorithm (or Kruskal's algorithm)

Input: Connected weighted graph

Step 1: Start with an empty graph

Step 2: Take all the edges that have not been selected and that would not create a cycle with the already selected edges. Add the one with the smallest weight.

Step 3: Repeat until the graph is connected and contains all vertices

Correctness of Kruskal's algorithm

Let T be the graph obtained as the output of Kruskal's algorithm to a weighted graph connected G.

We observe:

- T contains all vertices of G
  (otherwise the algorithm would not stop)
- T is connected
  (otherwise we could add one more edge without creating a cycle)
- T contains no cycles (by step 2)

$\Rightarrow$ T is a spanning tree.

Now we show that T has minimal weight.

Let F be another spanning tree of G.
We want to show that $wt(F) \geq wt(T)$

We number the edges of T according to the order we have added them while running Kruskal's algo.
⚠ This number is not the same as the weight

Let e be the edge with the smallest number such that $e \in E(T)$ and $e \notin E(F)$.

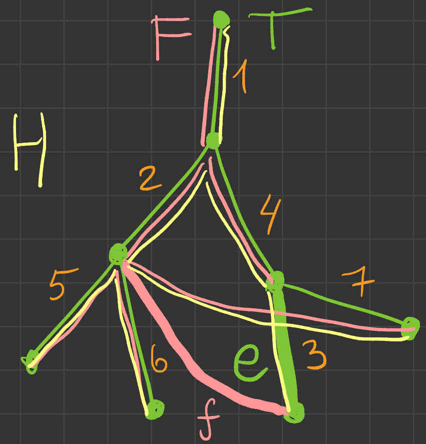
Add e to F => new graph will contain a cycle C

Let e be the edge with the smallest number
such that $e \in E(T)$ and $e \notin E(F)$.

Add e to F => new graph will contain a cycle C

C is not fully contained in T =>
C has an edge f that is not an edge of T

If we add the edge e to F and delete f
we get a (third) tree H.
Why is H a tree?
We want to show $wt(H) \leq wt(F)$
This is equivalent to $wt(f) \geq wt(e)$.



Want to show: $wt(f) \geq wt(e)$

Suppose $wt(f) < wt(e)$

Why in the algorithm we selected e and not f?

Only reason: f would form a cycle with edges of T already selected.

All previously selected edges of T are edges of F, f is an edge of F $\Rightarrow$ F contains a cycle.

Impossible since F is a tree. ↯

Therefore F can not be cheaper then H.

$wt(H) \leq wt(F)$ and H coinsides with T after more steps of algo

Repeat the procedure. After several iterations end up with T ▨