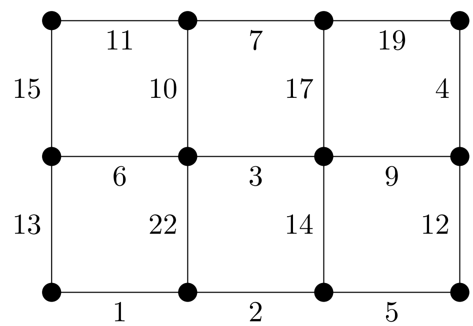
Chapters 4.2 - Trees and 4.3 - Minimum Spanning Tree

Spanning tree of a connected graph G is a spanning subgraph that is a tree.

Theorem 4.10 Every connected graph has a spanning tree.

Assume a function w assigning weight (cost) to edges of a graph G, that is $w : E(G) \to \mathbb{R}$.
Minimum Spanning Tree Problem: Find a spanning tree T of G minimizing $\sum_{e \in E(T)} w(e)$.

1: Find a minimum spanning tree of following graph.



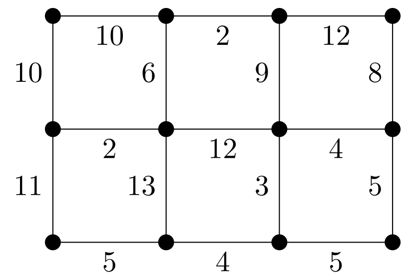
Kruskal's greedy algorithm [1956]: Order $E(G)$ increasingly according to w. Start with the empty spanning subgraph T of G, take edges according to the ordering one by one and add if T remains acyclic.

Jarník's [1930] – **Prim**'s [1957] algorithm: Start with T that is a single vertex of G. Find an edge e of the smallest cost that has only one endpoint in T and add e to T.

Borůvka's algorithm [1926]: Start with the empty spanning subgraph T of G, note T is a forest. For every connected component C of T, add an edge e of the smallest cost that has only one endpoint in C. Note the algorithm can run in parallel.

Theorem 4.11 Kruskal's algorithm produces minimum spanning tree.

2: Find minimum spanning tree of the following graph by running Kruskal's, Jarník's and Borůvka's algorithms.



3: Let C be a cycle in G and let T be a minimum spanning tree of G. Let e be the edge of maximum weight in C. Show that $e \notin E(T)$.

Corollary 4.6 Every forest on n vertices with k components has $n - k$ edges.

Theorem 4.7 Every connected graph on n vertices has at least $n - 1$ edges.

Theorem 4.8 If G is connected graph on n vertices with $n - 1$ edges, then G is a tree.

Theorem 4.8 If G is acyclic graph on n vertices with $n - 1$ edges, then G is a tree.

Theorem 4.9 Let T be a tree on k vertices. If G is a graph with $\delta(G) \geq k - 1$ then G contains a subgraph isomorphic to T.

Hints: 3 assume for contradiction $e \in E(T)$. 4.6 count edges in components; 4.7 Induction on n and find a leaf; 4.8 use 4.7 and show no cycles; 4.8 ; 4.9 induction on k; 4.11 take MST with as many edges in common as output of the algorithm.

Reading for next time: All up to 4.3 (skip 2.5, 3.3, 3.4) - midterm on Feb 11 (Thursday).