Eidgenössische
Technische Hochschule
Zürich

Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Federal Institute of Technology at Zurich

Departement of Computer Science
Markus Püschel, David Steurer
François Hublet, Goran Zuzic, Tommaso d'Orsi, Jingqiu Ding

12 December 2022

Algorithms & Data Structures — Exercise sheet 12 — HS 22

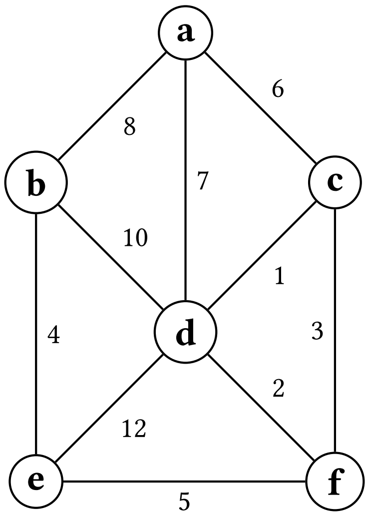
The solutions for this sheet are submitted at the beginning of the exercise class on 19 December 2022.

Exercises that are marked by * are "challenge exercises". They do not count towards bonus points.

You can use results from previous parts without solving those parts.

Exercise 12.1 *MST practice.*

Consider the following graph



a) Compute the minimum spanning tree (MST) using Boruvka's algorithm. For each step, provide the set of edges that are added to the MST.

b) Provide the order in which Kruskal's algorithm adds the edges to the MST.

c) Provide the order in which Prim's algorithm (starting at vertex **d**) adds the edges to the MST.

Exercise 12.2 *Maximum Spanning Trees and Trucking* (**2 points**).

We start with a few questions about **maximum spanning trees**.

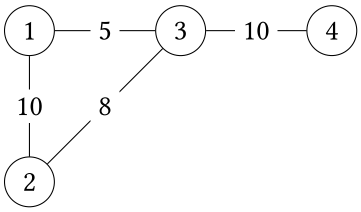
(a) How would you find the **maximum** spanning tree in a weighted graph G? Briefly explain an algorithm with runtime $O((|V| + |E|) \log |V|)$.

(b) Given a weighted graph $G = (V, E)$ with weights $w : E \to \mathbb{R}$, let $G_{\geq x} = (V, \{e \in E \mid w(e) \geq x\})$ be the subgraph where we only preserve edges of weight x or more. Prove that for every $s \in V, t \in V, x \in \mathbb{R}$, if s and t are connected in $G_{\geq x}$ then they will also be connected in $T_{\geq x}$, where T is the maximum spanning tree of G.

Hint: Use Kruskal's algorithm as inspiration for the proof.
Hint: If it helps, you can assume all edges have distinct weight and only prove the claim for that case.

Problem: You are starting a truck company in a graph $G = (V, E)$ with $V = \{1, 2, \ldots, n\}$. Your headquarters are in vertex 1 and your goal is to deliver the maximum amount of cargo to a destination $t \in V$ in a single trip. Due to local laws, each road $e \in E$ has a maximum amount of cargo your truck can be loaded with while traversing e. Find the maximum amount of cargo you can deliver for each $t \in V$ with an algorithm that runs in $O((|V| + |E|) \log |V|)$ time.

Example:



Output:
```
Max cargo to 1 is ∞
Max cargo to 2 is 10
Max cargo to 3 is 8
Max cargo to 4 is 8
```

Explanation:
The best path from the headquarters to 4 is $1 \to 2 \to 3 \to 4$, and the maximum cargo the truck can carry is $\min(10, 8, 10) = 8$.

(c) Prove that for every $t \in V$, the optimal route is to take the unique path in the **maximum** spanning tree of G.

Hint: Suppose that the largest amount of cargo we can carry from 1 to t in G (i.e., the correct result) is OPT and let ALG be the largest amount of cargo from 1 to t in the maximum spanning tree. We need to prove two directions: $OPT \leq ALG$ and $OPT \geq ALG$.
Hint: One direction holds trivially as any spanning tree is a subgraph. For the other direction, use part (b).

(d) Write the pseudocode of the algorithm that computes the output for all $t \in V$ and runs in $O((|V| + |E|) \log |V|)$. You can assume that you have access to a function that computes the maximum spanning tree from G and outputs it in any standard format. Briefly explain why the runtime bound holds.

Exercise 12.3 *Counting Minimum Spanning Trees With Identical Edge Weights* **(1 point)**.

Let $G = (V, E)$ be an undirected, weighted graph with weight function w.

It can be proven that, if G is connected and all its edge weights are pairwise distinct[1], then its Minimum Spanning Tree is unique. You can use this fact without proof in the rest of this exercise.

For $k \geq 0$, we say that G is *k-redundant* if k of G's edge weights are non-unique, e.g.

$$|\{e \in E \mid \exists e' \in E.\ e \neq e' \wedge w(e) = w(e')\}| = k.$$

In particular, if G's edge weights are all distinct, then G is 0-redundant, and if its edge weights are all identical, it is $|E|$-redundant.

(a) Given a weighted graph $G = (V, E)$ with weight function c and $e = \{v, w\} \in E$, we say that we *contract* e when we perform the following operations:

(i) Replace v and w by a single vertex vw in V, i.e., $V' \leftarrow V - \{v, w\} \cup \{vw\}$.

[1] I.e., for all $e \neq e' \in E$, $w(e) \neq w(e')$.

(ii) Replace any edge $\{v, x\}$ or $\{w, x\}$ by an edge $\{vw, x\}$ in E, i.e.,

$$E' \leftarrow E - \{\{v, x\} \mid x \in V\} - \{\{w, x\} \mid x \in V\} \cup \{\{vw, x\} \mid \{v, x\} \in E \vee \{w, x\} \in E\}.$$

(iii) Set the weight of the new edges to the weight of the original edges, taking the minimum of the two weights if two edges are merged, i.e.

$$\begin{aligned}
c'(\{x, y\}) &= c(\{x, y\}) && x, y \notin \{v, w\} \\
c'(\{vw, x\}) &= c(\{v, x\}) && \{v, x\} \in E, \{w, x\} \notin E \\
c'(\{vw, x\}) &= c(\{w, x\}) && \{v, x\} \notin E, \{w, x\} \in E \\
c'(\{vw, x\}) &= \min(c(\{v, x\}), c(\{w, x\})) && \{v, x\} \in E, \{w, x\} \in E.
\end{aligned}$$

For all $G = (V, E)$ and $e \in E$, we denote by G_e the graph obtained by contracting e in G. Explain why if T is an MST of G and $e \in T$, then T_e must be an MST of G_e.

(b) Let $k > 0$. Show that for all k-redundant $G = (V, E)$ and $e \neq e' \in E$ with $w(e) = w(e')$, then G_e is k'-redundant for some $k' \leq k - 1$.

(c) Show that if G is connected and k-redundant, it has at most 2^k distinct MSTs.

Hint: *By induction over k, using (a) and (b).*

(d) Show that for all large enough n, there exists a graph G such that G is n-redundant and has at least $2^{\frac{n}{2}}$ distinct MSTs.

Hint: *First assume that $n = 3k$ for some k. Consider graphs of the following form, where all unmarked edges have weight 0. When $n = 3k + 1$ or $n = 3k + 2$, you can add one or two edges with cost 0 at either end.*