École Polytechnique Fédérale de Lausanne
Prof. Eisenbrand — September 23, 2016
Assistant: Manuel Aprile

Combinatorial Optimization (Fall 2016)

Assignment 1

Deadline: September 30, noon, into the right box in front of MA C1 563.

Exercises marked with a $\star$ can be handed in for bonus points. Due date is September 30.

Problem 1

Complete the proof of Theorem 1.1 seen in class. In particular, given the conditions:

i) G is a tree.

iii) G is connected, and the removal of any edge disconnects G.

iv) G does not contain a cycle and the addition of one edge creates a cycle.

v) G is connected and $|E| = |V| - 1$.

Prove that iii) implies iv), iv) implies v) and v) implies i).

Solution:
iii) $\implies$ iv): If G contained a cycle, then removing any edge uv from the cycle wouldn't disconnect G as u and v remain connected through the remaining part of the cycle. Moreover, since G is connected, for any two vertices u, v either uv is an edge or there is a path joining u and v. In the latter case, adding the edge uv creates a cycle.
iv) $\implies$ v): for any two non-adjacent vertices u, v, adding the edge uv creates a cycle, hence there is a path joining u and v, i.e. G is connected. To show that $|E| = |V| - 1$, we argue by induction on $|E|$. For $|E| = 1$, $|V| = 2$ necessarily since G is connected. For $|E| > 1$, choose any edge e and remove it. This disconnects the graph (since otherwise e was part of a cycle) in two connected components G_1, G_2. It is easy to see that each G_i satisfies the hypotheses in iv), hence by induction

$$|E| = |E_1| + |E_2| + 1 = |V_1| - 1 + |V_2| - 1 + 1 = |V| - 1.$$

v) $\implies$ i): We only need to prove that G has no cycles. If G had a cycle, then we can remove one edge from it while leaving G connected. We keep removing edges until there is no cycle, obtaining a connected spanning subgraph of G, T. T is a tree, hence it has $|V| - 1$ edges from the previous implications. But this is a contradiction because we removed at least one edge, so $|E(T)| < |E|$.

Problem 2

Find, both with the Dijkstra-Prim algorithm (Algorithm 1.1) and with Kruskal's algorithm (Algorithm 1.2), a spanning tree of minimum length in the graph below.

Solution:
The two algorithms give in general different spanning trees, all of weight 26.

Problem 3

Let $G = (V, E)$ be a graph and $l : E \to \mathbb{R}$ be a length function. Call a forest $F \subset E$ *good* if $l(F') \geq l(F)$ for each forest F' satisfying $|F'| = |F|$.

Let F be a good forest and e be an edge not in F such that $F \cup \{e\}$ is a forest and such that (among all such e) $l(e)$ is as small as possible. Show that $F \cup \{e\}$ is good again.

Solution:

By contradiction, assume $F \cup \{e\}$ is not good, i.e. there is a forest $\bar{F}$ such that $|\bar{F}| = |F \cup \{e\}| = |F|+1$ and $l(\bar{F}) < l(F \cup \{e\}) = l(F) + l(e)$. Consider an edge f of $\bar{F} \setminus F$, since F is good we have $l(\bar{F} \setminus f) \geq l(F)$, hence $l(F) \leq l(\bar{F}) - l(f) < l(F) + l(e) - l(f)$ which implies $l(f) < l(e)$. This is only possible if $F \cup \{f\}$ is not a forest. We now show that there must be an edge $f \in \bar{F} \setminus F$ such that $F \cup \{f\}$ is still a forest, deriving a contradiction. If there is an edge of $\bar{F}$ that connects two components of F, or that has one endpoint that is not an endpoint of any edge of F, adding this edge to F won't create any cycle, and we are done. Suppose that there is no such edge. Then, for any edge $f \in \bar{F}$, f is in a component of F. Hence for each component U of F we must have $|\bar{F} \cap E(U)| \leq |F \cap E(U)| = |U| - 1$ ($U \subset V$ is a tree), but this is a contradiction since $|\bar{F}| > |F|$.

Problem 4

Let $G = (V, E)$ be a complete graph and let $l : E \to \mathbb{R}^+$ be a length function satisfying $l(uw) \geq \min\{l(uv), l(vw)\}$ for all distinct $u, v, w \in V$. Let T be a longest spanning tree in G. Show that for all $u, w \in V$, $l(uw)$ is equal to the minimum length of the edges in the unique $u - w$ path in T.

Solution:

Consider any pair of vertices $u, w \in V$. If $uw \in E(T)$, we are done. Otherwise, consider the path P joining u and w in T. P, together with the edge uw, forms a cycle, and removing any edge of this cycle we obtain a spanning tree of G (prove it!). For any $e \in P$, if $l(e) < l(uw)$, then we could obtain a tree longer than T by removing e and adding uw. Hence $l(uw) \leq \min_{e \in P} l(e)$. We now prove that equality holds by induction on $|P|$. First notice that a direct consequence of the hypothesis on l is that, for any $u, v, w \in V$, two of $l(uv), l(uw), l(vw)$ are equal, and the other one is equal or larger. This already implies the thesis for $|P| = 2$. For $|P| > 2$, let v be the successor of u in P, we have two cases. If $l(uw) = l(uv)$, then this quantity must be the minimum of $l(e)$, $e \in P$, and we are done. Otherwise $l(uw) < l(uv)$ and $l(uw) = l(vw)$. Then applying induction on v, w, (whose corresponding path is $P \setminus uv$) we get that

$$l(uw) = l(vw) = \min_{e \in P \setminus uv} l(e) = \min_{e \in P} l(e).$$

Problem 5

Let $G = (V, E)$ be a graph and $s : E \to \mathbb{R}$ be a *strength* function. For any path P, the *reliability* of P is defined as the minimum strength of the edges occurring in P. The *reliability* $r_G(u, v)$ of two vertices u, v is equal to the maximum reliability of P, where P ranges over all paths from u to v. Let T be a spanning tree of maximum strength, i.e. with $\sum_{e \in E(T)} s(e)$ as large as possible. Prove that T has the same reliability of G for any pair of vertices, that is:

$$r_T(u, v) = r_G(u, v) \;\; \forall\, u, v \in V.$$

Solution:

First notice that for any u, v, $r_T(u, v) \leq r_G(u, v)$ holds. Indeed, every path of T is also a path of G. Assume by contradiction that there are u, v such that $r_T(u, v) < r_G(u, v)$. Let P be the $u - v$ path in T, $e \in P$ such that $l(e) = r_T(u, v)$, and $\bar{P}$ be the $u - v$ path in G with maximum reliability. We have that for any $f \in \bar{P}$, $l(f) \geq l(e)$. If we find an edge $\bar{f}$ in P' that is not in T and such that $T \setminus \{e\} \cup \{\bar{f}\}$ is a spanning tree T', we are done because $s(T')$ would be bigger than $s(T)$, a contradiction. To find such edge, remove e from T: we obtain two connected components, one containing u and one v. There must be an edge $\bar{f}$ in P' that connects the two components: clearly $T \setminus \{e\} \cup \{\bar{f}\}$ is a spanning tree, since it is connected and doesn't contain any cycle.