MTH6105 – Algorithmic Graph Theory **Spring 2024**

Assessed Coursework 2 **F. Fischer**

This assessment consists of three exercises, which carry equal weight and together contribute **10% of your mark for the module**. Please **upload your answers before the deadline**.

Any work you submit must be your own. You may discuss the exercises with other students, but you must write up your solution yourself. Copying a solution or submitting someone else's solution constitutes an assessment offence.

---

1. Consider the network (G, w) with

$$V(G) = \{a, b, c, d, e, f\},$$
$$E(G) = \{ab, ac, ad, ae, af, bc, bd, be, bf, cd, ce, cf, de, df, ef\},$$

and

$$\begin{array}{lllll} w(ab) = 9, & w(ac) = 8, & w(ad) = 12, & w(ae) = 3, & w(af) = 15, \\ w(bc) = 5, & w(bd) = 6, & w(be) = 13, & w(bf) = 10, & w(cd) = 4, \\ w(ce) = 14, & w(cf) = 2, & w(de) = 16, & w(df) = 11, & w(ef) = 7. \end{array}$$

(a) Use Kruskal's algorithm to find a minimum spanning tree of (G, w). List the edges of the tree in the order in which they are added, and draw the tree.

(b) Show that the minimum spanning tree found by Kruskal's algorithms is in fact the unique minimum spanning tree of (G, w).

Solution:

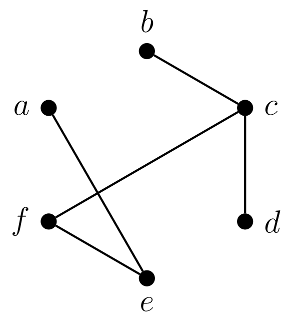
(a) The following ordering of the edges is non-decreasing in weight, and in fact is the only ordering with this property:

$$cf,\ ae,\ cd,\ bc,\ bd,\ ef,\ ac,\ ab,\ bf,\ df,\ ad,\ be,\ ce,\ af,\ de.$$

Kruskal's algorithm adds edges to the spanning tree in this order, unless adding an edge would create a cycle. It thus adds the edges

$$cf,\ ae,\ cd,\ bc,\ ef$$

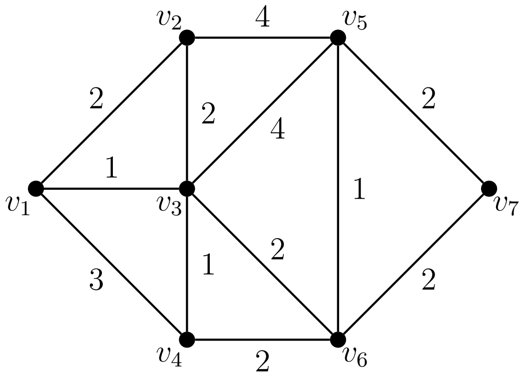
and obtains the following spanning tree:



(b) The spanning tree T obtained by the two algorithms is unique if and only if every edge $e \in E(G) \setminus E(T)$ is the unique edge of maximum weight in the unique cycle formed by e together with the edges of T. This condition is clearly satisfied for all edges in $(E(G) \setminus E(T)) \setminus \{bd\}$, because their weight is greater than that of all edges in $E(T)$. It is also satisfied for edge bd because $6 = w(bd) > \max\{w(bc), w(cd)\} = \max\{5, 4\}$.

---

2. Consider the following network (G, w).



(a) Use Dijkstra's algorithm to find a shortest v_1–v_7-path. Give $V(T)$ and $E(T)$ after each iteration of the algorithm.

(b) Explain the effect each of the following changes of the length of a single edge would have on the length of a shortest v_1–v_7-path:

(i) an increase of $w(v_3v_4)$ from 1 to 2;

(ii) a decrease of $w(v_6v_7)$ from 2 to 1;

(iii) a decrease of $w(v_3v_5)$ from 4 to 1.

Solution:

(a) The algorithm may for example construct T as follows.

$V(T) = \{v_1\}$	$E(T) = \emptyset$	$\delta(v_1) = 0$
$V(T) = \{v_1, v_3\}$	$E(T) = \{v_1v_3\}$	$\delta(v_3) = 1$
$V(T) = \{v_1, v_3, v_2\}$	$E(T) = \{v_1v_3, v_1v_2\}$	$\delta(v_2) = 2$
$V(T) = \{v_1, v_3, v_2, v_4\}$	$E(T) = \{v_1v_3, v_1v_2, v_3v_4\}$	$\delta(v_4) = 2$
$V(T) = \{v_1, v_3, v_2, v_4, v_6\}$	$E(T) = \{v_1v_3, v_1v_2, v_3v_4, v_3v_6\}$	$\delta(v_6) = 3$
$V(T) = \{v_1, v_3, v_2, v_4, v_6, v_5\}$	$E(T) = \{v_1v_3, v_1v_2, v_3v_4, v_4v_6, v_6v_5\}$	$\delta(v_4) = 4$
$V(T) = \{v_1, v_3, v_2, v_4, v_6, v_5, v_7\}$	$E(T) = \{v_1v_3, v_1v_2, v_3v_4, v_4v_6, v_6v_5, v_6v_7\}$	$\delta(v_7) = 5$

Note that the algorithm has a choice whether to first include v_1v_2 or v_3v_4, which it can make in an arbitrary way. The unique v_1–v_7-path in T, the path $P = v_1, v_3, v_6, v_7$, has length $\delta(v_7) = 5$ and is a shortest v_1–v_7-path in (G, w).

(b) (i) The shortest v_1–v_7-path P determined by Dijkstra's algorithm does not contain the edge v_3v_4. An increase of $w(v_3v_4)$ thus does not affect the length of P, and it cannot decrease the length of any other path. This implies that P remains a shortest v_1–v_7-path.

(ii) The shortest v_1–v_7-path P determined by Dijkstra's contains the edge v_6v_7. A decrease of $w(v_6v_7)$ thus decreases the length of P by the same amount, and cannot decrease the length of any other path by more than this amount. This implies that P remains a shortest v_1–v_7-path, and its length decreases by 1.

(iii) The shortest v_1–v_7-path P determined by Dijkstra's algorithm does not contain the edge v_3v_5. A decrease of $w(v_3v_5)$ thus does not affect the length of P, but it may decrease the length of other paths and thus create a new shortest path. Indeed, we can apply Dijkstra's algorithm to the modified network to find a new shortest path v_1, v_3, v_5, v_7 of length 4.

---

3. Consider a network (G, w). Let $c \in \mathbb{R}$, and let $m : E(G) \to \mathbb{R}$ such that $m(e) = w(e) + c$ for all $e \in E(G)$.

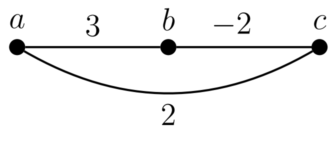
(a) Show that T is a minimum spanning tree of (G, w) if and only if it is a minimum spanning tree of (G, m).

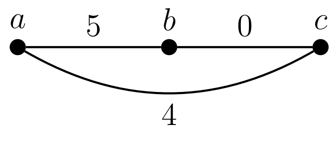
(b) What is the analogous claim for shortest paths in (G, w) and (G, m)? Prove this claim or provide a counterexample showing that it is not true.

Solution:

(a) Both networks have the same set of spanning trees, namely the set of spanning trees of G. Consider two arbitrary spanning trees S and T of G, and let $s = \sum_{e \in E(S)} w(e)$ and $t = \sum_{e \in E(T)} w(e)$. Since S and T are spanning trees of G, $|E(S)| = |E(T)| = |V(G)| - 1$. Thus $\sum_{e \in E(S)} m(e) = s + (|V(G)| - 1)c$ and $\sum_{e \in E(T)} m(e) = t + (|V(G)| - 1)c$. A spanning tree thus has minimum weight in (G, w) if and only if it has minimum weight in (G, m).

(b) The claim is that a path in G is a shortest s–t-path in (G, w) if and only if it is a shortest s–t-path in (G, m). To see that the claim is false, consider the following two networks.





The network on the right has been obtained from the network on the left by increasing the weight of every edge by 2. The unique shortest a–c-path in the network on the left is a, b, c, whereas the unique shortest a–c-path in the network on the right is a, c.
