| CSE 431/531: Algorithm Analysis and Design | Spring 2022 |
|---|---|

# Homework 5

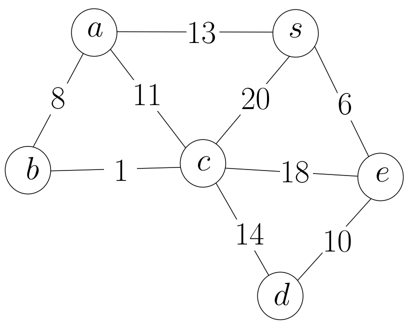
*Instructor: Shi Li* **Deadline: 5/4/2022**

Your Name: ______________________ Your Student ID: ________________

| Problems | 1 | 2 | 3 | 4 | Total |
|---|---|---|---|---|---|
| Max. Score | 25 | 25 | 25 | 25 | 80 |
| Your Score | | | | | |

The total score for the 4 problems is 100, but your score will be truncated at 80.

**Problem 1** Consider the following graph $G$ with non-negative edge weights.




Use Prim's algorithm to compute the minimum spanning tree of $G$. You need to give the minimum spanning tree and its weight.

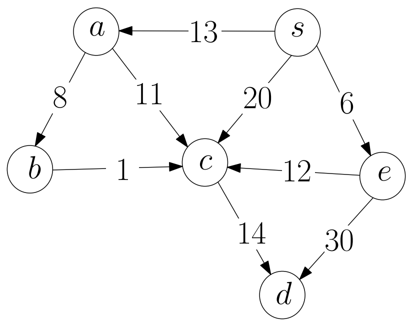
You need to use the following table to describe the execution of the algorithm. If $d[v] = \infty$, then $\pi[v] =$ "$\perp$". Also, when a vertex $v$ has been added to $S$, you can leave its $d$ and $\pi$ values empty, to make the table clean. But it is not required to do so.

| iteration | vertex added to $S$ | $a$: $d$ | $a$: $\pi$ | $b$: $d$ | $b$: $\pi$ | $c$: $d$ | $c$: $\pi$ | $d$: $d$ | $d$: $\pi$ | $e$: $d$ | $e$: $\pi$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | $s$ | 13 | $s$ | $\infty$ | $\perp$ | 20 | $s$ | $\infty$ | $\perp$ | 6 | $s$ |
| 2 | $e$ | 13 | $s$ | $\infty$ | $\perp$ | 18 | $e$ | 10 | $e$ | | |
| 3 | $d$ | 13 | $s$ | $\infty$ | $\perp$ | 14 | $d$ | | | | |
| 4 | $a$ | | | 8 | $a$ | 11 | $a$ | | | | |
| 5 | $b$ | | | | | 1 | $b$ | | | | |
| 6 | $c$ | | | | | | | | | | |

Table 1: Prim's Algorithm for Minimum Spanning Tree

The MST contains the edges $(s, a), (a, b), (b, c), (e, d)$ and $(s, e)$. It has weight $13 + 8 + 1 + 10 + 6 = 38$.

**Problem 2** Consider the following directed graph $G$ with non-negative edge weights. Use Dijkstra's algorithm to compute the shortest paths from $s$ to all other vertices in $G$.




You need to fill the following table. When $d[v] = \infty$, we set $\pi[v] =$"$\perp$". Also, when a vertex $v$ has been added to $S$, you can leave its $d$ and $\pi$ values empty, to make the table clean. But it is not required to do so.

| iteration $i$ | vertex added to $S$ in iteration $i$ | $a$ | | $b$ | | $c$ | | $d$ | | $e$ | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | $d$ | $\pi$ | $d$ | $\pi$ | $d$ | $\pi$ | $d$ | $\pi$ | $d$ | $\pi$ |
| 1 | $s$ | 13 | $s$ | $\infty$ | $\perp$ | 20 | $s$ | $\infty$ | $\perp$ | 6 | $s$ |
| 2 | $e$ | 13 | $s$ | $\infty$ | $\perp$ | 18 | $e$ | 36 | $e$ | | |
| 3 | $a$ | | | 21 | $a$ | 18 | $e$ | 36 | $e$ | | |
| 4 | $c$ | | | 21 | $a$ | | | 32 | $c$ | | |
| 5 | $b$ | | | | | | | 32 | $c$ | | |
| 6 | $d$ | | | | | | | | | | |

Table 2: Dijkstra's algorithm for Shortest Paths

**Problem 3** We are given an undirected graph $G = (V, E)$ with non-negative edge weights $(w_e)_{e \in E}$. Assume all the weights are different.

(3a) Let $T$ be the unique minimum spanning tree of $G$. Is the following statement true or false? If we change the weight of every edge $e$ from $w_e$ to $w_e^2$, then $T$ is still the unique minimum spanning tree of $G$. Justify your answer.

True. The execution of Prim's algorithm (or Kruskal's algorithm) only depends on the relative order of the weights. As squaring the weights do not change the order, the execution of the Prim's algorithm will not be affected. It will produced the same MST for both scenarios.

(3b) Let $s$ and $t$ be two distinct vertices in $V$. Let $P$ be the unique shortest path from $s$ to $t$. Is the following statement true or false? If we change the weight of every edge $e$ from $w_e$ to $w_e^2$, then $P$ is still the unique shortest path from $s$ to $t$. Justify your answer.

False. Consider a graph with 4 vertices $s, a, b$ and $t$, and 4 edges, $(s, a)$ with weight 3, $(a, t)$ with weight 4, $(s, b)$ with weight 1 and $(b, t)$ with weight 5. There are only two simple paths from $s$ to $t$: $s$-$a$-$t$, which has weight 7, and $s$-$b$-$t$, which has weight 6. Thus the shortest $s$-$t$ path is $s$-$b$-$t$. However if we square the weights, the weights of $s$-$a$-$t$ and $s$-$b$-$t$ becomes $3^2 + 4^2 = 25$ and $5^2 + 1^2 = 26$ respectively. The shortest $s$-$t$ path becomes $s$-$a$-$t$.

By justifying your answer, we mean the following: If the answer is yes, you need to give a proof. If your answer is no, you need to give a counter-example.

**Problem 4** Suppose there is a negative cycle over a directed graph $G = (V, E)$ with edge weights $w : \mathbb{E} \to \mathbb{R}$. Then show that for any array $d : V \to \mathbb{R}$ over vertices, there exists some edge $(u, v) \in E$ such that $d[u] + w(u, v) < d[v]$.

(So if we do not put an upper bound on the number of iterations the Bellman-Ford algorithm runs, it will run forever when there is a negative cycle.)

Let $C$ be the negative cycle and $a \geq 2$ be the size of $C$, i.e, the number of vertices in $C$. Let $v_1, v_2, \cdots, v_a$ be the set of vertices in $C$, so that $(v_1, v_2), (v_2, v_3), \cdots, (v_{a-1}, v_a), (v_a, v_1) \in C$.

Assuming towards to the contradiction that there exists $d : V \to \mathbb{R}$ such that for every edge $(u, v) \in E$ we have $d[u] + w(u, v) \geq d[v]$, which is equivalent to $d[v] - d[u] \leq w(u, v)$. Then, we have

$$\begin{aligned}\text{the weight of } C &= w(v_1, v_2) + w(v_2, v_3) + \cdots + w(v_{a-1}, v_a) + w(v_a, v_1) \\ &\geq (d[v_2] - d[v_1]) + (d[v_3] - d[v_2]) + \cdots + (d[v_a] - d[v_{a-1}]) + (d[v_1] - d[v_a]) \\ &= 0.\end{aligned}$$

This contradicts the fact that $C$ is a negative cycle. So, the vector $d$ does not exist.