CSCI3160: Regular Exercise Set 9

Prepared by Yufei Tao

Problem 1*. Let $G = (V, E)$ be a weighted directed acyclic graph. Given a source vertex $s \in V$, design an algorithm to find the shortest path distances from s to the vertices in V. Your algorithm should terminate in $O(|V| + |E|)$ time.

Problem 2. Let $G = (V, E)$ be a weighted directed graph where the weight of an edge (u, v) is $w(u, v)$. It is guaranteed that G has no negative cycles. Prove: the following is a correct implementation of Bellman-Ford's algorithm:

algorithm Bellman-Ford
1. pick an arbitrary vertex $s \in V$
2. set λ to the sum of all the positive edge weights in G
3. initialize $dist(s) = 0$ and $dist(v) = \lambda$ for every other vertex $v \in V$
4. **for** $i = 1$ **to** $|V| - 1$
5.   relax all the edges in E
6. **return** $dist(v)$ for all $v \in V$

Remark: Compared to the description in our lecture notes, the key difference here is that, at Line 3, we initialize $dist(v)$ as λ, instead of ∞.

Problem 3*. Let $G = (V, E)$ be a weighted directed graph where the weight of an edge (u, v) is $w(u, v)$. Prove: the following algorithm correctly decides whether G has a negative cycle:

algorithm negative-cycle-detection
pick an arbitrary vertex $s \in V$
set λ to the sum of all the positive edge weights in G
initialize $dist(s) = 0$ and $dist(v) = \lambda$ for every other vertex $v \in V$
for $i = 1$ **to** $|V| - 1$
    relax all the edges in E
for each edge $(u, v) \in E$
    if $dist(v) > dist(u) + w(u, v)$ **then**
        return "there is a negative cycle"
return "no negative cycles"

Problem 4. In our lecture about the Floyd-Warshall algorithm, we have given the following recursive function:

$$spdist(i, j \mid\leq k) = \min \begin{cases} spdist(i, j \mid\leq k-1) \\ spdist(i, k \mid\leq k-1) + spdist(k, j \mid\leq k-1) \end{cases}$$

Give the details of computing $spdist(i, j)$ for all $i, j \in [1, n]$ in $O(n^3)$ time.

Problem 5. Augment your algorithm for the previous problem to compute the shortest path between vertex i and vertex j, for all $i, j \in [1, n]$.