Assignment 8

Post Date: 12 Dec 2016  Due Date: 19 Dec 2016, 1 pm
You are permitted and encouraged to work in groups of two.

Problem 1: Dating Problem 4 Points

An online dating-service has received personal information from a set $V$ of men and a set $W$ of women. For each man $v_i \in V$ and each woman $w_j \in W$, a compatibility score $s_{i,j} > 0$ is calculated. The task is to arrange dates such that the total compatibility of the partners is maximized.

Translate the task into a min-cost flow problem.

Problem 2: Successive Shortest-Paths I 7 Points

Apply the Successive Shortest-Path algorithm to the min-cost flow problem below.

[Diagram of the min-cost flow problem with vertices and edges labeled with capacities and costs.]
Problem 3: All-Pairs Shortest Paths 9 Points

On a directed graph $G = (V, E)$ with $n$ vertices, $m$ edges and edge weights $\omega : E \rightarrow \mathbb{R}$, the Single-Source Shortest Paths (SSSP) problem is to find shortest paths from a single source vertex to all other vertices. If $(G, \omega)$ does not have any negative-weight edges, the algorithm of Dijkstra solves the SSSP problem on $(G, \omega)$ in $O(m + n \log n)$ time. On graphs with negative-weight edges, but no negative-weight cycles, however, the algorithm of Dijkstra cannot be used. Instead, the algorithm of Bellman and Ford computes the shortest paths from a single source vertex to all other vertices on such a graph $(G, \omega)$ in $O(nm)$ time.

In the context of a software project you work on, you require an algorithm that finds shortest paths between all pairs of vertices in a graph. This problem is known as the All-Pairs Shortest Paths (APSP) problem.

(a) Give algorithms that solve the APSP problem

i. on directed graphs $(G, \omega)$ without any negative-weight edges in $O(nm + n^2 \log n)$ time and

ii. on directed graphs $(G, \omega)$ with negative-weight edges, but no negative-weight cycles, in $O(n^2m)$ time.

In your software project, the graph $(G, \omega)$ contains negative-weight edges, but no negative-weight cycles. You wonder if the APSP problem can actually be solved on such a graph $(G, \omega)$ in $o(n^2m)$ time. While brainstorming various ideas, you realize that you might be able to solve the APSP problem faster if you can find some suitable vertex potentials $\pi : V \rightarrow \mathbb{R}$.

(b) Find a way to compute potentials $\pi : V \rightarrow \mathbb{R}$ on $(G, \omega)$ in $O(nm)$ time such that the reduced weights $\omega_\pi(e) = \pi(v)+\omega(e)−\pi(w)$ are non-negative for any edge $e = (v, w) \in E$.

(c) Show that a $v$-$w$-path $P$ is a shortest path on $G$ with weights $\omega$ if and only if it is a shortest path on $G$ with reduced weights $\omega_\pi$.

(d) Conclude that the APSP problem can be solved in $O(nm + n^2 \log n)$ time on directed graphs $(G, \omega)$ with negative-weight edges, but no negative-weight cycles.

Note: You may use the algorithms of Dijkstra and of Bellman and Ford as building blocks as you need them. It is not necessary to understand or argue about the internals of these SSSP algorithms to solve the exercise.