Karl Bringmann and Sebastian Krinninger **Summer 2016**

Exercises for Complexity Theory of Polynomial-Time Problems

https://www.mpi-inf.mpg.de/departments/algorithms-complexity/teaching/summer16/poly-complexity/

Exercise sheet 6 Due: **Monday, July 18, 2016**

Total points: 40

Either email an electronic version of your assignment submission to gjindal@mpi-inf.mpg.de or give it to Gorav in his office at Room 425, Building E1.3. If Gorav is not in his office then you can just slide your submission under the door of his office.

You are allowed to collaborate on the exercise sheets, but you have to write down a solution on your own, using your own words. Please indicate the names of your collaborators for each exercise you solve. Further, cite all external sources that you use (books, websites, research papers, etc.).

You need to collect at least 50% of all points on exercise sheets.

Exercise 1 (*10 points*) Given a unweighted directed graph $G = (V, E)$ and a source vertex $s \in V$, **SSR** (single source reachability) is the problem of checking for every node t if it is reachable from s, i.e, if there is a path from s to t. Consider the partially dynamic version of SSR.

a) (4 points) Show an *incremental* algorithm with total update time $O(|E|)$ and query time $O(1)$.

b) (6 points) When G is a DAG (directed acyclic graph), demonstrate a *decremental* algorithm with total update time $O(|E|)$ and query time $O(1)$.

Exercise 2 (*10 points*) In the lecture, we saw that there is a decremental algorithm for maintaining APSP in unweighted directed graphs with total update time $O(mn \log^2 n)$ and query time $O(n)$. Modify this algorithm (or create your own algorithm) to demonstrate a decremental algorithm for maintaining APSP in unweighted directed graphs with total update time $O(mn(t + \log^2 n))$ and query time $O(\frac{n \log n}{t})$ for any $1 \leq t \leq n$.

Note : Putting $t = 1$ gives us the algorithm we showed in the lecture.

Exercise 3 (*10 points*) Consider the dynamic version of APSP called APSPM where we want the distance matrix to be stored explicitly (which immediately gives a query time of $O(1)$ for answering distance queries between any pair of nodes).

a) (6 points) Show an $\Omega(n^2)$ time lower bound per update for fully dynamic APSPM.

b) (4 points) For incremental APSPM, show an $\Omega(n^3)$ time lower bound on the total update time with $O(n)$ insertions starting from the empty graph (with n nodes but no edges).

Exercise 4 (*10 points*) In lecture we presented the Even/Shiloach algorithm, which has a total update time time of $O(mn)$ for both incremental and decremental SSSP in unweighted graphs. In this exercise we consider barriers for improving this algorithm in dense graphs.

a) (6 points) Show that there is no combinatorial incremental SSSP algorithm with $O(n^{3-\epsilon})$ total update time and $O(1)$ query time unless BMM has a $O(n^{3-\delta})$ time combinatorial algorithm.

b) (4 points) Show that there is no incremental SSSP algorithm with $O(n^{3-\epsilon})$ total update time and $O(1)$ query time unless the OMv conjecture fails.