Karl Bringmann and Vasileios Nakos **Summer 2020**

Sublinear Algorithms, Exercise Sheet 4

www.mpi-inf.mpg.de/departments/algorithms-complexity/teaching/summer20/sublinear-algorithms/

Total Points: **40** Due: 12:00 (noon), Monday, **July 13**, 2020

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 to be admitted to the exam.

Exercise 1 — 10 points

Fill in the missing details for the lecture:

1. You are given blackbox access to a coin which is either (1) biased with probability δ (that is, the coin shows heads with probability δ and tails otherwise), or (2) deterministic (that is, it *always* shows tails). Come up with an algorithm distinguishing these two cases with probability $2/3$ using $O(\delta^{-1})$ coin throws.
2. Recall the definitions of convolutions ($*$) and cyclic convolutions ($*_c$) from the lecture. Prove that these two problems are equivalent. More precisely, prove that if the convolution of two length-d vectors can be computed in time $T(d)$, then the cyclic convolution of two length-d vectors can be computed in time $O(T(d))$, and vice versa.

Exercise 2 — 10 points

Consider the following property testing problem: Given two length-n strings x, y and an integer k, your task is to distinguish whether x and y are similar (that is, they differ in at most k positions) or dissimilar (that is, they differ in at least $(1+\varepsilon)k$ positions). Devise a randomized algorithm for this problem with success probability $1-\delta$ in time $O(nk^{-1}\varepsilon^{-2}\log\delta^{-1})$.

Exercise 3 — 10 points

Consider the following property testing problem: Given blackbox access to two linear functions $f, g : \{0,1\}^n \to \{0,1\}$, your task is to check whether $f = g$. Design a randomized algorithm with success probability $3/4$ and constant running time.

Exercise 4 — 10 points

The *diameter* D in an undirected graph is the maximum distance between any pair of nodes. In this exercise we will design a property tester for the following problem: Given an undirected n-vertex graph G, distinguish whether the diameter of G is at most D, or whether G is ε-far from a graph of diameter $4D+2$ (that is, it is necessary to add at least εn edges to G to obtain a diameter of at most $4D+2$). We are always assuming that G is given in the sparse graph model (aka the adjacency list model). Prove correctness of the following algorithm (with probability $2/3$) and analyze its running time:

- Select a subset S of $4\varepsilon^{-1}$ start vertices (uniformly at random).
- Starting from every start vertex, run a breadth-first search. Stop a breadth-first search as soon as (A) $\min\{2\varepsilon^{-1}, n\}$ vertices have been explored, or (B) all vertices up to distance D have been explored.

- Accept (and claim that the diameter is $\leq D$) if and only if case (B) never occured.

*Hint: The proof goes along the following lines. For a vertex v, the D-*neighborhood *of v contains all vertices of distance at most D to v. Prove that:*

1. *If the D-neighborhood of each vertex contains at least $2\varepsilon^{-1}$ vertices, then G can be transformed into a graph with diameter at most $4D + 2$ by adding at most $\frac{1}{2}\varepsilon n$ edges. To prove this, greedily pick disjoint balls of radius D as long as possible. Somehow connect their centers.*
2. *If the D-neighborhood of at least $(1 - \frac{1}{2}\varepsilon)n$ vertices contains at least $2\varepsilon^{-1}$ vertices, then G can be transformed into a graph with diameter at most $4D + 2$ by adding at most εn edges.*
3. *Use the previous claim to show completeness of the property tester.*