Albert-Ludwigs-Universität, Inst. für Informatik
Prof. Dr. Fabian Kuhn
H. Ghodselahi, Y. Maus January 19, 2017

Algorithm Theory, Winter Term 2016/17
Problem Set 7

hand in (hard copy or electronically) by 13:55, January 30, 2017,
tutorial session will be on February 02, 2017.
Note that the dates are different than the usual.

Exercise 1: Randomized Independent Set Algorithm (10+10 points)

Let $G = (V, E)$ be a graph with n vertices and m edges. An independent set in a graph G is a subset $S \subseteq V$ of the nodes such that no two nodes in S are connected by an edge. Let $d := \frac{1}{n}\sum_{v \in V} \deg(v) = \frac{2m}{n}$ be the average node degree and consider the following randomized algorithm to compute an independent set S.

(I) Start with an empty set S. Then independently add each node of V with probability $1/d$ to S (you can assume that $d \geqslant 1$).

(II) The subgraph induced by S might still contain some edges and we therefore need to remove at least one of the nodes of each of the remaining edges. For this, we use the following deterministic strategy: As long as S is not an independent set, pick an arbitrary node $u \in S$ which has a neighbor in S and remove u from S.

It is clear that the above algorithm computes an independent set S of G.

a) (10 points) Find a (best possible) lower bound on the expected size of S at the end of the algorithm. Your lower bound should be expressed as a function of n and d.

Hint: First compute the expected numbers of nodes in S and edges in $G[S]$ after Step (I) of the algorithm.

b) (10 points) Assume that the above algorithm has running time $T(n)$ and that it computes an independent set of size $\frac{n}{5d}$ with probability at least $\frac{1}{2}$.

Show how to compute an independent set of size at least $\frac{n}{5d}$ with probability $1 - \frac{1}{n}$. What is the running time of your algorithm?

Exercise 2: Randomized partial 3-coloring (10 points)

The maximum 3-coloring problem asks for assigning one of the colors $\{1, 2, 3\}$ to each node $v \in V$ of a graph $G = (V, E)$ such that the number of edges $\{u, v\} \in E$ for which u and v get different colors is maximized. A simple randomized algorithm for the problem would be to (independently) assign a uniform random color to each node.

What is the expected approximation ratio of this algorithm?

Exercise 3: Random Max Cut Computation (10 points)

In the lecture, we discussed the random contraction algorithm to obtain a minimum edge cut. One could try to use the same algorithm to also find a maximum edge cut (partition $A \subset V, B = V \setminus A$ of the nodes so that the number of edges connecting nodes in A and B is maximized).

Show that for some graphs, the probability that the contraction algorithm returns a maximum cut is 0.