University of Freiburg
Dept. of Computer Science
Prof. Dr. F. Kuhn
S. Faour, M. Fuchs, A. Malyusz

Algorithm Theory
Exercise Sheet 13

Due: Friday, 2nd of February 2024, 10:00 am

Exercise 1: Miscellaneous Approximations *(8 Points)*

Let $G = (V, E)$ be an undirected connected graph .

(a) The *minimum dominating set* problem asks to find a dominating set $D \subseteq V$ of minimum size. Show that for $c \geq 2$, the `domset` algorithm from the previous sheet (c.f. sheet 12, exercise 2) computes an $\mathcal{O}(\ln n)$-approximation of a minimum dominating set with probability at least $1 - \frac{2}{n}$. *(3 Points)*

(b) 1. An *independent set* is a set $I \subseteq V$ such that no two nodes in I share an edge in E. The *maximum independent set* problem asks to find an independent set of maximum size. Recall that the *minimum vertex cover* problem asks to find a vertex cover of minimum size. Now, show that both optimization problems are equivalent i.e. finding the minimum-size vertex cover is equivalent to finding the maximum-size independent set . *(1 Point)*

2. Show that the two problems are not equivalent in an approximation-preserving way, i.e it is not true that for all positive integer α, finding an α-approximate minimum vertex cover is equivalent to finding a $1/\alpha$-approximate maximum independent set.

Hint: Give a counterexample by finding a family of graphs where one can easily obtain a 2-approximate minimum vertex cover, but this will equivalently find a very bad approximate maximum independent set. *(4 Points)*

Exercise 2: A Set Cover Variant *(12 Points)*

We consider the following variant of the set cover problem discussed in the lecture. We are given a set of elements X and a collection $\mathcal{S} \subseteq 2^X$ of subsets of X such that $\bigcup_{S \in \mathcal{S}} S = X$. In addition, we are given an integer parameter $k \geq 2$.

Instead of finding a collection $\mathcal{C} \subseteq \mathcal{S}$ of the sets which covers all elements, the goal is to find a set of **at most** k sets $S_1, \ldots, S_k \in \mathcal{S}$ such that the number of covered elements $|S_1 \cup \cdots \cup S_k|$ is maximized.

We consider the greedy set cover algorithm from the lecture, but we stop the algorithm after adding k sets.

(a) Show that for $k = 2$, the described greedy algorithm has approximation ratio at most 4/3. *(5 Points)*

(b) Let us now consider a general parameter $k \geq 2$. Show that if an optimal choice of k sets $S_1, \ldots, S_k$ covers ℓ elements, after adding t sets, the greedy algorithm covers at least $\frac{\ell}{k} \cdot \sum_{i=1}^{t} \left(1 - \frac{1}{k}\right)^{i-1}$ elements. *(5 Points)*

(c) Prove that the approximation ratio of the greedy algorithm is at most $\frac{e}{e-1}$. You can use that $(1 - 1/k)^k < e^{-1}$. *(2 Points)*