Exercises to the lecture
Complexity Theory
Sheet 6

Prof. Dr. Roland Meyer
Thomas Haas Delivery until 24.01.2022 at 15:00

Exercise 6.1 (Maximal satisfiability)

We construct a kernelization for the following problem:

Maximal Satisfiability (MAXSAT)	
Input:	A Boolean formula $\varphi = \bigwedge_{i=1}^{m} C_i$, where the C_i are clauses, and $k \in \mathbb{N}$.
Parameter:	$k \in \mathbb{N}$.
Question:	Is there a variable assignment that satisfies at least k clauses of φ?

Let (φ, k) be an instance of the problem. The first step of the kernelization is to delete all *trivial* clauses. We call a clause *trivial* if it contains a variable and its negation.

a) Show that by removing all trivial clauses, we can reduce (φ, k) to an instance (ψ, k') such that $k' \leq k$ and $(\psi, k') \in$ MAXSAT if and only if $(\varphi, k) \in$ MAXSAT.

In the next step, we delete *long* clauses, clauses that contain more than k' literals.

b) Prove the following: If ψ contains more than k' long clauses, then $(\psi, k') \in$ MAXSAT.

c) Let t denote the number of long clauses in ψ and set $\hat{k} = k' - t$. Show that by removing all long clauses, we can reduce (ψ, k') to an instance $(\rho, \hat{k})$ such that $(\rho, \hat{k}) \in$ MAXSAT if and only if $(\psi, k') \in$ MAXSAT.

Hence, we obtain an instance that only consists of clauses of size at most k'. We argue that we are only interested in such formulas the size of which is bounded by the parameter.

d) Prove the following: If $(\rho, \hat{k})$ has more than $2\hat{k}$ clauses, then $(\rho, \hat{k})$ is in MAXSAT.

e) Summarize the reduction steps in an algorithm and show that the size of the kernel (the size of the obtained instance) is bounded by $\mathcal{O}(k^2)$.

Exercise 6.2 (Set Cover)

Consider the following problem:

Set Cover	
Input:	A family of sets $(S_i)_{i\in[1..m]}$ over a universe $U = \bigcup_{i\in[1..m]} S_i$ with n elements, and an $\ell \in \mathbb{N}$.
Parameter:	$\lvert U\rvert = n \in \mathbb{N}$.
Question:	Are there ℓ sets $S_{i_1}, \ldots, S_{i_\ell}$ from the family such that $U = \bigcup_{j\in[1..\ell]} S_{i_j}$?

Develop an algorithm for Set Cover that relies on the Inclusion/Exclusion principle. Show that it runs in time $\mathcal{O}^*(2^n)$.

Hint: This is quite similar to the algorithm for computing the chromatic number.

Exercise 6.3 (Count TSP)

In this exercise, we want to establish an algorithm for the following problem:

Counting Traveling Salesperson (Count TSP)	
Input:	A complete (each two vertices are connected) graph $G = (V, E)$ and a weight function $w : E \to \{0, \ldots, W\}$.
Parameter:	$\lvert V\rvert = n \in \mathbb{N}$.
Question:	What is the number of Hamiltonian cycles that admit minimal weight?

Let $\pi = v_0 v_1 \ldots v_k$ be a path or cycle in G. Then the weight of π is $w(\pi) = \sum_{i=0}^{k-1} w(v_i v_{i+1})$. The problem asks for the number of Hamiltonian cycles the weight of which is minimal among all Hamiltonian cycles.

Develop an algorithm for Count TSP based on the Inclusion/Exclusion principle, that runs in $\mathcal{O}^*(2^n)$ time.

Hint: It is easier if you fix the weight in the universe. For each weight j of a Hamiltonian cycle, define the universe U_j to be the cycles of length n, starting in v_0, of weight j. Then proceed as in the Inclusion/Exclusion-algorithm for Hamil Cycle. At some point, one has to count the number of cycles that have weight j in a certain graph. Use a dynamic programming approach for this task.

Delivery until 24.01.2022 at 15:00 to `https://cloudstorage.tu-braunschweig.de/preparefilelink?folderID=2UBADHPF6dDgmphBeAHkK`.