Exercise I, Computational Complexity 2024

These exercises are for your own benefit. Feel free to collaborate and share your answers with other students. Solve as many problems as you can and ask for help if you get stuck for too long. Problems marked * are more difficult but also more fun :).

1 Prove that the following problems are in NP. Which of them are also in P?

(a) A Boolean formula is said to be in Disjunctive Normal Form (DNF) if it is an OR ($\vee$) of a number of terms where each term is an AND ($\wedge$) of the literals present in it. For instance, the following is a DNF formula:

$$(x \wedge y \wedge \bar{z}) \vee (\bar{y} \wedge z) \vee (\bar{x} \wedge \bar{y})$$

The DNF-SAT problem asks, given a DNF formula, if it is satisfiable.

(b) Given n positive integers $a_1, a_2, \ldots, a_n$ in their binary representation, is there a set $S \subseteq \{1, 2, \ldots, n\}$ such that

$$\sum_{i \in S} a_i = \sum_{i \in \bar{S}} a_i \,.$$

Here, $\bar{S}$ denotes the complement of the set S. That is $\bar{S} = \{1, 2, \ldots, n\} \setminus S$.

(c) Given a pair of integers (n, m) in binary, does there exist $2 \leq k \leq m$ such that k divides n?

(d) A k-CNF formula is a CNF formula where each clause has at most k literals. The 2-SAT problem asks if a 2-CNF formula given as input is satisfiable.

2 Is NP is closed under **union**? That is, if $L, L' \in \mathsf{NP}$ then do we always have $L \cup L' \in \mathsf{NP}$? What about **intersection** $L \cap L'$, **complement** $\bar{L} = \Sigma^* \smallsetminus L$, and **Kleene closure**, which is defined for a language $L \subseteq \Sigma^*$ by (here uv is the concatenation of strings u and v)

$$L^* = \{u_1 u_2 \ldots u_k : u_1, u_2, \ldots, u_k \in L \text{ and } k \geq 0\} \,.$$

(Hint: For one of the four operations mentioned above, it is still **unknown** *whether* NP *is closed under that operation—do not spend too much time trying to prove closure for all four!)*

3 Define coNP as the class of languages whose complements are in NP, that is, $\mathsf{coNP} = \{L : \bar{L} \in \mathsf{NP}\}$. Show that if $\mathsf{P} = \mathsf{NP}$ then $\mathsf{NP} = \mathsf{coNP}$. Is the converse necessarily true?

4 Suppose $\mathsf{P} = \mathsf{NP}$ and let A be a polynomial-time algorithm for SAT. Show that, by using A as a subroutine, we can find in polynomial time a satisfying assignment to a CNF formula if one exists. *(Hint: Call A repeatedly to build up the satisfying assignment one variable at a time.)*