Computational Complexity

Exercise Session 4

Note: These exercises are (likely) too much work to solve all during the exercise session.

Exercise 1. Prove that $\mathsf{L} \subseteq \mathsf{P}$.

Definition 1. We define DP to be the following complexity class:

$$\mathsf{DP} = \{\ A \cap B \mid A \in \mathsf{NP}, B \in \mathsf{coNP}\ \}.$$

Exercise 2.

(a) Explain the difference between DP and the class $\mathsf{NP} \cap \mathsf{coNP}$.

(b) Prove that $\mathsf{NP} \cup \mathsf{coNP} \subseteq \mathsf{DP}$.

(c) Prove that $\mathsf{P} = \mathsf{DP}$ if and only if $\mathsf{P} = \mathsf{NP}$.

(d) Prove that the following problem MAX-SAT is in DP:

> **Instance:** A propositional formula φ in CNF and a positive integer $k \in \mathbb{N}$.
>
> **Question:** Is the maximum number of clauses of φ that can be (simultaneously) satisfied by a truth assignment α exactly k?

Exercise 3. Consider the problem SINGLE-CYCLE, where the input consists of an undirected graph $G = (V, E)$, and the question is to decide if G consists of one single cycle. Show that SINGLE-CYCLE is in L.