Write down on each answer sheet:

- Your name, degree programme, student number, and the date
- The course title: CS-E4530 Computational Complexity Theory

1. Order the complexity classes L, NP, EXP, PSPACE, P, Σ_2^p, Π_2^p, and ZPP by set inclusion (that is, write enough set inclusion statements of the form

$$X \subseteq Y$$

where X and Y are complexity classes given above such that all known set inclusions follow from the statements).

2. (a) Let $L_1, L_2 \in \mathsf{NP} \cap \mathsf{coNP}$. Prove that the language

$$L_1 \oplus L_2 = \{x \in \{0,1\}^* \colon \ x \text{ is in exactly one of } L_1 \text{ and } L_2\}$$

is in $\mathsf{NP} \cap \mathsf{coNP}$.

(b) A Boolean circuit C with n inputs and n outputs can be viewed as a *succinct encoding* of a specific kind of directed graph; specifically, we can view a circuit C describing a graph G_C on vertex set $V_C = \{0,1\}^n$ with edge set

$$E_C = \{(x,y) \colon C(x) = y\}.$$

That is, each vertex $x \in \{0,1\}^n$ has exactly one outgoing edge, with the other endpoint being $C(x) \in \{0,1\}^n$.

Consider the following problem about succinct graphs:

SUCCINCT CYCLE:

- **Instance:** A Boolean circuit C with n inputs and n outputs.
- **Question:** Is the graph G_C described by C a directed cycle on 2^n vertices? That is, is there a permutation $\sigma \colon \{1,2,\dots,2^n\} \to \{0,1\}^n$ such that $C(\sigma(i)) = \sigma(i+1)$ for all $i = 1,2,\dots,2^n - 1$ and $C(\sigma(2^n)) = \sigma(1)$?

Prove that the SUCCINCT CYCLE problem is in PSPACE.

(*You may assume it is known that evaluating the value of a Boolean circuit on input x can be done in polynomial time and space.*)

3. Let $\alpha > 1$ be a constant. Prove that if there is an α-approximation algorithm for the minimum dominating set problem, then there is an α-approximation algorithm for the minimum set cover problem.

Recall that the optimisation versions of these problems are defined as follows:

MINIMUM SET COVER:

- **Instance:** A finite set U and a family $\mathcal{S} = \{S_1, S_2, \dots, S_m\}$ of subsets of U.
- **Feasible solution:** A subfamily $\mathcal{T} \subseteq \mathcal{S}$ such that any element $u \in U$ is contained in at least one set $S \in \mathcal{T}$.
- **Objective:** Minimise $c(\mathcal{T}) = |\mathcal{T}|$.

(*Continued on the next page.*)

MINIMUM DOMINATING SET:

- **Instance:** A graph $G = (V, E)$.
- **Feasible solution:** A subset $D \subseteq V$ such that for all $v \in V$, either $v \in D$ or at least one neighbour of v is in D.
- **Objective:** Minimise $c(D) = |D|$.

Grading: Each problem 8pt, total 24pt.
Passing grade: 12pt.