Homework 2, Computational Complexity 2024

The deadline is 23:59 on Wednesday 6 November. Please submit your solutions on Moodle. Typing your solutions using LaTeX is strongly encouraged. The problems are meant to be worked on in groups of 2–3 students. Please submit only one writeup per team. You are strongly encouraged to solve these problems by yourself. If you must, you may use books or online resources to help solve homework problems, but you must credit all such sources in your writeup and you must never copy material verbatim.

1 Prove that $\mathsf{NP}^{\mathrm{SAT}} = \Sigma_2\mathsf{P}$.

2 Construct an oracle A such that $\mathsf{RP}^A \neq \mathsf{coRP}^A$. Namely, consider the class of oracles

$$\mathcal{A} := \Big\{ A \subseteq \{0,1\}^* : \forall n,\ \ |A \cap \{0,1\}^n|/2^n \in \{\tfrac{1}{2}, 0\} \Big\}$$

and the associated language $L_A = \{1^n : |A \cap \{0,1\}^n|/2^n = \frac{1}{2}\}$. Show that

(i) $L_A \in \mathsf{RP}^A$ for every $A \in \mathcal{A}$.
(ii) $L_A \notin \mathsf{coRP}^A$ for some $A \in \mathcal{A}$.

3 A *3-colouring* of a graph $G = (V, E)$ is an assignment $c\colon V \to \{1,2,3\}$ such that no edge $\{u, v\} \in E$ is *monochromatic*, that is, $c(u) = c(v)$. Consider the following *3-colouring game* played by Alice and Bob on a graph G where $V = \{1, \ldots, n\}$. In each round of the game, we have a partial 3-colouring $c\colon V \to \{1,2,3,*\}$. Initially, $c(v) = *$ for all v. In round $i = 1, \ldots, n$, if i is odd (resp. even), then Alice (resp. Bob) chooses a colour $k \in \{1,2,3\}$ and we update the partial colouring by $c(i) := k$. The player who first creates a monochromatic edge ($\{u, v\} \in E$ such that $c(u) = c(v) \neq *$) loses the game. (If after n rounds there is no monochromatic edge, Alice wins.) Prove that the following problem is PSPACE-complete

$$\text{3-ColourGame} := \Big\{ \langle G \rangle : G \text{ is a graph such that Alice has a winning strategy for the 3-colouring game on } G \Big\}.$$

(*Hint: Deciding whether a graph is 3-colourable is* NP*-complete. It might help you to first figure out how to prove this. For example, one can reduce from SAT using the following three subgraphs. Source: p. 325 in Sipser's textbook*)