Computational Complexity

Homework Sheet 2

Hand in via Canvas before May 7, 2024, at 23:59

`https://canvas.uva.nl/courses/42595/assignments/496111`

For this homework assignment, solve Exercises 1 and 2. In addition, choose exactly one of Exercises 3 and 4, and solve it. This adds up to a total of 10 points that you can (maximally) obtain.

Exercise 1 (3pt)**.** Prove that $\mathsf{NTIME}(n) \neq \mathsf{P}$.

- $\mathsf{NTIME}(n)$ can be characterized as the set of all decision problems that can be verified in linear time with a linear-size certificate. That is, $A \in \mathsf{NTIME}(n)$ if and only if there is a linear-time Turing machine $\mathbb{M}$ and a constant c such that for all $x \in \{0,1\}^*$ it holds that $x \in A$ if and only if there exists some $u \in \{0,1\}^{c \cdot |x|}$ such that $\mathbb{M}(x,u) = 1$. You are allowed to use this characterization of $\mathsf{NTIME}(n)$.
- *Hint:* Use the Nondeterministic Time Hierarchy Theorem.

Exercise 2 (3pt)**.** Is there an oracle such that, relative to this oracle, ...? If so, then give such an oracle and prove that it works. If not, prove why not.

(a) $\mathsf{P} = \mathrm{EXP}$

(b) $\mathsf{coNP} \subseteq \mathsf{P}$ and $\mathsf{NP} \not\subseteq \mathsf{P}$

(c) $\mathsf{DTIME}(n) = \mathsf{DTIME}(n^2)$

(d) $\mathsf{NP} = \mathsf{coNP} \neq \mathrm{EXP}$

For example, in **(a)** you have to either (i) show that there exists an oracle A such that $\mathsf{P}^A = \mathrm{EXP}^A$ or (ii) show that such an oracle does not exist. In **(b)**, you have to either (i) show that there there exists an oracle A such that $\mathsf{coNP}^A \subseteq \mathsf{P}^A$ and $\mathsf{NP}^A \not\subseteq \mathsf{P}^A$ or (ii) show that such an oracle does not exist.

Exercise 3 (4pt)**.** In this exercise, we will construct a decision problem $A \subseteq \{0\}^*$ that is not auto-reducible, using diagonalization. (For a definition of auto-reducibility, see the previous homework sheet.)

(a) Consider the function $b : \mathbb{N} \to \mathbb{N}$ such that $b(0) = 1$ and for each $n > 0$ it holds that $b(n) = 2^{b(n-1)}$. Show that there exists some i_0 such that for all $i \geq i_0$ it holds that $b(i) > b(i-1)^{i-1}$.

(b) Let $\mathbb{M}$ be a polynomial-time oracle Turing machine[1] that—when given input $x \in \{0\}^*$—does not query x to the oracle. Show that there exists some i such that $\mathbb{M} = \mathbb{M}_i$, and $\mathbb{M}_i^O$ runs in time at most n^i for inputs of size $n \geq 2$, for all oracles O.

- *Hint:* Remember that we can choose our representation scheme $i \mapsto \mathbb{M}_i$ in such a way that every Turing machine has infinitely many representations.

(c) Suppose that $\mathbb{M}_i^O$—from **(b)**—is given the string $0^{b(i)}$ as input, for some $b(i) > 1$. What can you say about the size of the queries that $\mathbb{M}_i^O$ makes to O?

[1] With *polynomial-time oracle Turing machine*, we mean an oracle Turing machine $\mathbb{M}$ for which there exists a polynomial p such that for each oracle $O \subseteq \{0,1\}^*$ and each input x, $\mathbb{M}$ takes time at most $p(|x|)$ when using O as the oracle. In other words, there is a fixed polynomial p such that $\mathbb{M}$ runs in time $p(n)$, regardless of the choice of oracle O.

(d) Construct a set $A \subseteq \{0\}^*$ that is not auto-reducible. Construct A in stages A_i such that $A = \bigcup_{i \geq 1} A_i$. Recursively define $A_i \subseteq \{0\}^{b(i)}$ in such a way that A is not auto-reducible by construction. Make sure to prove that the set is not auto-reducible.

- *Hint:* suppose you have constructed $A_1, \ldots, A_{i-1}$. Let $A_{\leq i-1} = \bigcup_{1 \leq j \leq i-1} A_j$. Consider the behavior of machine $\mathbb{M}_i^{A_{\leq i-1}}$ with oracle access to $A_{\leq i-1}$ when given input $0^{b(i)}$—that does not query $0^{b(i)}$. Based on the output of $\mathbb{M}_i^{A_{\leq i-1}}$ on $0^{b(i)}$, choose whether $0^{b(i)}$ is in A_i or not.

Definition 1. Define the complexity class DP as follows:

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

Definition 2. Let $G = (V, E)$ be an undirected graph. A subset $C \subseteq V$ of vertices is called a *clique* of G if every $v_1, v_2 \in C$ with $v_1 \neq v_2$ are connected by an edge in E.

Consider the decision problems Clique and Exact-Clique:

$$\mathsf{Clique} = \{\ (G, k) \mid G \text{ is an undirected graph that has a clique of size } k\ \}.$$

$$\mathsf{Exact\text{-}Clique} = \{\ (G, k) \mid G \text{ is an undirected graph that has a clique of size } k \text{ but has no clique of size } k+1\ \}.$$

Exercise 4 (4pt)**.**

(a) Prove that if $\mathsf{DP} \subseteq \mathsf{NP}$, then the Polynomial Hierarchy collapses.

(b) Prove that Exact-Clique is DP-complete (under polynomial-time reductions).

- *Hint:* Establish (and prove) and use the following lemma. Let L be an arbitrary problem in NP. Then there exists a polynomial-time reduction R from L to Clique such that for every $x \in \{0,1\}^*$ the instance $(G, k) = R(x)$ has the following additional properties:
  - G has no clique of size $k+1$.
  - If G has no clique of size k, then it also has no clique of size $k-1$.

Remark 1. Answers will be graded on two criteria: they should (1) be correct and intelligent, and also (2) concise and to the point.

Remark 2. If you find a solution to one of the exercises in a paper or book, you can use this to inform your solution. Make sure that you write down the solution in your own words, conveying that you understand what is going on.