Complexity Theory - Assignment 1

Kishlaya Jaiswal, Satya Prakash Nayak

February 6, 2020

Exercise 1

Proof. We shall prove the contrapositive that $P = NP \implies EXP = NEXP$. So assume $P = NP$ then it suffices to show $NEXP \subseteq EXP$.
Let $L \in NEXP$, then there exist a non-deterministic turing machine M with running time $T(x) = O(2^{|x|^c})$ that decides L. Define $L_p = \{x10^{T(x)} \mid x \in L\}$.

We claim that $L_p \in NP$. For any $y \in L_p$, we have $|y| = |x| + T(x) + 1$. Define NTM M' that does the following: on input y it throws out the last $10^{T(x)}$ parts and run M on x (remaining part) and does whatever M does. Since M runs in $T(x)$ time, M' is a NP machine. Therefore, $L_p \in NP \implies L_p \in P$, so there exists a deterministic poly-time machine D which decides L_p. Now we can construct a deterministic machine D' which on input x, will pad it with $10^{T(x)}$ and check whether $x10^{T(x)} \in L_p$ using the machine D on input x. Hence, $L \in EXP$. $\square$

Exercise 2

Proof. We shall show $SAT \leq_p HALT$.
Let N be the poly-time NTM that decides SAT. Consider the modified NTM M which on input x, accepts if N accepts x and otherwise if N rejects then M goes into a loop, and hence never halts. Let α be the encoding for this machine M.
Given a SAT-instance ϕ encoded as x, consider the string (α, x) as an instance of $HALT$ problem. (Note that: $M_\alpha = M$)
Suppose $\phi \in SAT$, then there exists a satisfying assignment and so N accepts $x \implies M = M_\alpha$ halts and accepts x and therefore $(\alpha, x) \in HALT$. Conversely, $\phi \notin SAT$, then N rejects $x \implies M = M_\alpha$ doesn't halt on x and therefore $(\alpha, x) \notin HALT$.
Thus, $HALT$ is NP-hard.

Furthermore, $HALT \notin NP$ because otherwise, if there exists a NTM which decides $HALT$, then given a machine M_α and an input x, we could decide if M_α halts on input x, hence solving the HALTING problem. So $HALT$ is not NP-complete. $\square$

Exercise 3

Proof. $L_1 \in NP \implies \exists$ a polynomial p_1 and a poly-time predicate B_1 such that $x \in L_1 \iff \exists w, |w| \leq p_1(|x|) \wedge B_1(w,x) = 1$
$L_2 \in NP \implies \exists$ a polynomial p_2 and a poly-time predicate B_2 such that $x \in L_2 \iff \exists w, |w| \leq p_2(|x|) \wedge B_2(w,x) = 1$

To show $L_1 \cup L_2 \in NP$, consider the polynomial $p = p_1 + p_2$ and the poly-time predicate $B = B_1 \vee B_2$, then: $x \in L_1 \cup L_2 \iff (x \in L_1) \vee (x \in L_2) \iff (\exists w, |w| \leq p_1(|x|) \wedge B_1(w,x) = 1) \vee (\exists w, |w| \leq p_2(|x|) \wedge B_2(w,x) = 1) \iff \exists w, |w| \leq p(|x|) \wedge B(w,x) = 1$

To show $L_1 \cap L_2 \in NP$, consider the polynomial $p = p_1 + p_2$ and the poly-time predicate $B(w_1 \# w_2) = B_1(w_1) \wedge B_2(w_2)$, then: $x \in L_1 \cap L_2 \iff (x \in L_1) \wedge (x \in L_2) \iff (\exists w_1, |w_1| \leq p_1(|x|) \wedge B_1(w_1,x) = 1) \wedge (\exists w_2, |w_2| \leq p_2(|x|) \wedge B_2(w_2,x) = 1) \iff \exists w, w = w_1 \# w_2, |w| \leq p(|x|) \wedge B(w,x) = 1$ $\square$

Exercise 4

Proof. . It is clear that $TAUT \in coNP$ because given any NO instance $\phi(x)$ of $TAUT$, a short certificate for verification is an assignment of variables x such that $\phi(x) = 0$. Clearly such an assignment can be expressed in $O(n \lg n)$ size (where $n =$ number of variables).
To show that $TAUT$ is $coNP - complete$, we note that, for any two NP languages A and B, $A \leq_p B \iff \overline{A} \leq_p \overline{B}$.
Thus, if L is $NP - complete$, then $\overline{L}$ is $coNP - complete$ because firstly, $\overline{L} \in coNP$ by definition and for any $L' \in coNP$, $\overline{L'} \leq_p L \implies L' \leq_p \overline{L}$. So, we conclude that $\overline{3SAT}$ is $coNP - complete$.
To show that $TAUT$ is $coNP - complete$, we show that $\overline{3SAT} \leq_p TAUT$. Given any formula φ, consider the formula $\overline{\varphi}$. This formula can be constructed in $O(|\varphi|)$ time. Now, φ doesn't have a satisfying assignment iff $\overline{\varphi}$ is a tautology.

Suppose $NP = coNP$. Since $3SAT$ is $NP - complete$, for every NP language L, $L \leq_p 3SAT$. But $TAUT \in coNP = NP \implies TAUT \leq_p 3SAT$. Similarly, since $TAUT$ is $coNP - complete$, for every $coNP$ language L, $L \leq_p TAUT$. But $3SAT \in NP = coNP \implies 3SAT \leq_p TAUT$.
Conversely, let $L \in NP$, then $L \leq_p 3SAT \leq_p TAUT$. So given a NO instance of the problem L, we reduce it to a NO instance of $TAUT$ problem and hence we have a short certificate for the NO instances of L as well implying that $L \in coNP$. Similarly, let $L \in coNP$, then $L \leq_p TAUT \leq_p 3SAT$. So given a YES instance of the problem L, we reduce it to a YES instance of $3SAT$ problem and hence we have a short certificate for the YES instances of L as well implying that $L \in NP$. Thus, $NP = coNP$. $\square$

Exercise 5

Proof. Suppose unary $NP \subseteq P$.
Let $L \in NEXP$, then there exist a non-deterministic turing machine M with running time $T(x) = 2^{|x|^c}$ that decides L.
Define $L_u = \{Unary(x) \mid x \in L\}$.

Observe that if $x \in L$, then x is a binary string of length $|x|$ and so the unary representation of x will have size atmost $2^{|x|}$.
We claim that $L_u \in NP$. For that we construct a machine M' which on input 1^y, converts y into it's binary x and then simulate M on x. M' accepts 1^y iff M accepts x. Since $|1^y| = y < 2^{|x|}$, M requires $2^{|x|} + 2^{|x|^c} = O(2^{|x|^c}) = O(y^c)$ time and hence $L(M') = L_u \in NP$.
Therefore, $L_u \in P$, so there exists a deterministic poly-time machine D which decides L_u.
Now we can construct a deterministic machine D' which on input x, will construct $Unary(x)$ and check if it is in L_u using the machine D. Hence, $L \in EXP$. □

Exercise 6

Proof. Since $P = NP$, we claim that $P = coNP$: $L \in coNP \implies \overline{L} \in NP \implies \overline{L} \in P \implies L \in coP = P$.
Thus $TAUT \in P$ and so there exists a poly-time algorithm A such that $A(\varphi(y)) = 1$ iff $\varphi(y)$ is a tautology.
We shall show that $\Sigma_2 SAT \in NP$. Consider a YES instance of $\Sigma_2 SAT$, that is a formula $\psi = \exists x \forall y(\phi(x, y) = 1)$ which admits a x_0 such that $\phi(x_0, y)$ is a tautology.
Then a short certificate for $\psi \in \Sigma_2 SAT$ is x_0; because clearly $|x_0| \leq |\psi|$ and in poly-time we can check if $\phi(x_0, y)$ is a tautology by calling $A(\phi(x_0, y))$.
In other words, $\psi \in \Sigma_2 SAT \iff \exists x_0$ such that $|x_0| \leq |\psi|$ and $A(\phi(x_0, y)) = 1$.
Hence $\Sigma_2 SAT \in P$. □