COMS 4236: Computational Complexity (Fall 2018)
Problem Set #3

Wenbo Gao - wg2313@columbia.edu

December 9, 2018

Problem 1

Problem. Problem 11.5.18 on page 275 of CC: Show that, if NP $\subseteq$ BPP, then RP = NP. (That is, if SAT can be solved by randomized machine, then it can be solved by randomized machines with no false positives, presumably by computing a satisfying truth as in Example 10.3.)

Problem 2

Problem. Let $0 < \epsilon_1 < \epsilon_2 < 1$ denote two constants. Let $D(\cdot, \cdot)$ be a deterministic polynomial-time computable Boolean function, and let L be a language (the setting so far is exactly the same as the definition of BPP.) D and L satisfy the following property: Given any $x \in \{0,1\}^n$, if y is sampled uniformly at random from $\{0,1\}^m$ for some m polynomial in n, then

$$x \in L \Rightarrow \Pr_{y \in \{0,1\}^m}[D(x,y) = 1] \geq \epsilon_2 \text{ and } x \notin L \Rightarrow \Pr_{y \in \{0,1\}^m}[D(x,y) = 1] \leq \epsilon_1.$$

Show that $L \in$ BPP. (Note that ϵ_2 can be smaller than $1/2$. Use the Chernoff bound.)

Problem 3

Problem. Similar to P/poly, one can define P/$\log n$, where the advice string is of length only $O(\log n)$ for input size n. Show that, if SAT $\in$ P/$\log n$, then P $=$ NP. (Hint: Self-reducibility.)

Problem 4

Problem. Show that, if PSPACE $\subseteq$ P/poly, then PSPACE $= \Sigma_2^P$. (Hint: Use self-reducibility to "implicitly" build a winning strategy for the existential player in the TQBF game.)