January, 2018

Computational Complexity **Exercise 1**

1. Show that if any NP-complete language L is in co-NP, then co-NP=NP. □

2. Show that if $g(n) = o(f(n))$, $f(n) \geq \log n$ space constructible, then Nspace$(g(n)) \neq$ Nspace$(f(n) \log f(n))$. (Hint: Let $L \in$ Nspace$(g(n))$. Assume that you have two non-deterministic machines M and $\overline{M}$, each using $O(g(n))$ space accepting L and $\overline{L}$ (why should such machines exist?). Now you can use these machines to do a diagonal argument as in the proof of the deterministic space hierarchy theorem). □

3. Define **polyL** as $\bigcup_i Dspace(\log^i n)$. Steve's class **SC**=$\{L : \exists$ constants k, l and a deterministic Turing machine M such that M accepts L in $O(n^k)$ space and $O(\log^l n)$ time $\}$ Why does it NOT follow from the definition that **SC**=**PolyL** or **SC = polyL**$\cap$**P**? □

4. We have defined **NP** as $\bigcup_{i\geq 0}$NTIME(n^i). Here is an alternative definition. We define **NP** to be the class of all languages $L \subseteq \Sigma^*$ such that there is a deterministic Turing machine M that runs in time polynomial in the size of its input and an integer k such that if $x \in L$ then there exists $y \in \Sigma^*$ with $|y| \leq |x|^k$ and $M(x, y) = 1$ whereas, if $x \notin L$, for all $y \in \Sigma^*$ $M(x, y) = 0$. This is the *certificate* characterization of the class **NP** (See Korman et.al., Introduction to algorithms for a treatment of NP-completeness using this definition). *Show that* the two definitions are equivalent. (ie., show that for any language $L \in$ **NP**, such machine M exists and conversely, given such M, we can construct a polynomial time NDTM M' for accepting $\{x : \exists y, |y| \leq |x|^k M(x, y) = 1\}$. □

5. Let $L \in NTIME(2^{n^c})$ for some constant $c > 0$. Define the language $L_{pad} = \{(x, 1^z) : x \in L, z = 2^{|x|^c}\}$. Show that $L_{pad} \in$ **NP**. Using the above observation argue that if **NEXP** $\neq$ **EXP** then **P**$\neq$ **NP**. The technique used in this proof is called *padding argument*. □

6. Here is another question using the padding argument. Let L be any language and k any positive integer. Define $L_{pad} = \{x\#0^l : x \in L, |x| + l + 1 = |x|^k\}$. Show that $L \in$ **P** if and only if $L_{pad} \in$ **P**. Using this result, argue that P$\neq$ Dspace(n). Using Savitch theorem, extend the arguments to prove that Nspace$(n) \neq$ **P**. (Use the stronger version of the non-deterministic space hierarchy theorem proved in the second question). □

7. Prove that **PolyL** $\neq$ **P** □