CMSC 28100-1 / MATH 28100-1
Introduction to Complexity Theory
Fall 2017 – Homework 7

November 9, 2017

Exercise 1. Suppose $T\colon \mathbb{N} \to \mathbb{N}$ is a function with $T(n) \geqslant n+1$ such that there exists a *single tape* nondeterministic Turing machine M_T such that for every input x of length n, all branches of computation of M_T on x take time at most $T(n)$ (i.e., the machine M_T is $T(n)$-time bounded) and at least one branch of computation of M_T takes time exactly $T(n)$.

Show that a k-tape nondeterministic Turing machine running in a nondeterministic time $T(n)$ can be simulated by a 2-tape nondeterministic Turing machine running in nondeterministic time $O(T(n))$. Note that, unlike the case of deterministic time, there is no additional $\log(T(n))$ factor. Only state essential ideas (i.e., why we don't need the extra $\log(T(n))$ factor) in plain English like an algorithm.

Exercise 2. Show that if $\mathrm{DSPACE}(n) \subseteq \mathrm{P}$, then $\mathrm{PSPACE} = \mathrm{P}$. Recall that $\mathrm{PSPACE} = \bigcup_{k\geqslant 1} \mathrm{DSPACE}(n^k)$. Hint: padding and PSPACE-completeness.

Exercise 3. Show that $\mathrm{DSPACE}(n) \neq \mathrm{P}$. Hint: reinspect Exercise 2 under the light of hierarchy theorems.

Exercise 4. Show that $\mathrm{NSPACE}(n) \neq \mathrm{P}$.