CMSC 28100-1 / MATH 28100-1
Introduction to Complexity Theory
Fall 2017 – Homework 8

November 16, 2017

Exercise 1. Show that if P = NP, then every language in NP is NP-complete, except for $\varnothing$ and Σ^*.

Exercise 2. Consider the following language:

$$K = \{(M, x, 1^t) : M \text{ is an NTM that accepts } x \text{ within } t \text{ steps}\}.$$

(a) Show that $K \in \mathrm{NTIME}(n)$.

(b) Show directly (not by reduction from another known NP-complete language) that K is NP-complete.

Exercise 3. Show that if SAT $\in$ P, then there is a deterministic polynomial-time Turing machine M such that for all formulas ϕ, if ϕ is satisfiable then $M(\phi)$ outputs a satisfying assignment to ϕ, and otherwise M rejects. This is called solving the "search version" of SAT (searching for a witness, rather than merely determining if one exists).

Exercise 4. A language L is *p-selective* if there is a polynomial-time (deterministic) Turing machine M such that the following hold.

(a) For every $x, y \in \Sigma^*$, we have $M(x, y) \in \{x, y\}$, that is, on input (x, y), the machine M outputs either x or y.

(b) For every $x, y \in \Sigma^*$, if at least one of x and y is in L, then M outputs a string in L (which must be either x or y by (a)).

Show that if SAT is p-selective, then P = NP. Hint: use ideas from Exercise 3.