CSE 547T Automata Theory Spring Semester 2017

Homework 4: Complexity Theory (Revised)

Assigned: 4/5/2017 *Due Date: 4/24/2017*

This homework must be completed and submitted electronically. Formatting standards, submission procedures, and (optional) document templates for homeworks may be found at

`http://classes.engineering.wustl.edu/cse547/ehomework/ehomework-guide.html`

Advice on how to compose homeworks electronically, with links to relevant documentation for several different composition tools, may be found at

`http://classes.engineering.wustl.edu/cse547/ehomework/composing-tips.html`

Please remember to

- **create a separate PDF file (typeset or scanned) for each problem;**
- **include a header with your name, WUSTL key, and the homework number at the top of each page of each solution;**
- **include any figures (typeset or hand-drawn) inline or as floats;**
- **upload and submit your PDFs to Blackboard before class time on the due date.**

Always show your work and prove your constructions correct.

1. Suppose language L can be decided in time $t(n)$ by a single-tape, deterministic TM. Show that, if $t(n) = \omega(n^2)$, then L can be decided by a single-tape, deterministic TM in time $t(n)/c$ for any $c > 0$.

2. A *unary language* is a subset of 0^*. Suppose that some unary language A is NP-complete; show that we can use A to decide SAT in polynomial time (and hence, that $P = NP$).

   (*Hint*: consider the process of enumerating all 2^n truth assignments to a formula of size n to see if any one is satisfying. If $SAT \leq_p A$, how can this enumeration be sped up so as to become polynomial-time? You can't actually decide A fast, but you do get a polytime transducer f mapping Boolean formulas to strings in 0^*, s.t. $f(\phi) \in A$ iff ϕ is satisfiable.)

3. The *longest path with forbidden pairs* problem LPFP(G,k,F) presents a directed graph G and a set F of vertex pairs u, v in G and asks, is there a path p in G of length at least k, such that for every pair $u, v \in F$, at most one of u and v is on p?

   Show that LPFP is NP-complete. (*Hint*: reduce from 3SAT.)

4. We saw in class that 3SAT is NP-complete problem. Show that the restriction 2SAT, in which each clause has (exactly) two literals, is in P. (*Hint*: consider the implications.)