CS 181d, Complexity Theory
Fall 2012
Homework 5
Due Tuesday, October 16 in class

Please note that this homework is due earlier than a regular homework and that it is shorter as well. If you have a Euro remaining, you may use it to turn this homework in on Wednesday, October 17 at 9:35 AM.

The take-home exam will be distributed in class on Tuesday, October 16 and will be due back no later than Friday, October 19 at 5 PM.

1. [**30 Points**] **The Time Hierarchy Theorem!** Recall the amazing Space Hierarchy Theorem that stated that if $f(n)$ is space-constructible then there exists a language L such that L is decided in space $O(f(n))$ but is not decided by any machine that uses asymptotically less space - that is space $o(f(n))$. In this problem, you will prove a similar theorem for time.

   There are several ways of defining the notion of a "time constructible" function. The definition that we'll use is this: *A function $f(n)$ is time constructible if there exists a standard single-tape Turing Machine that on any input of length n runs for exactly $f(n)$ steps and then halts.* Although we won't prove it here, the set of time constructible functions is closed under multiplication, addition, exponentiation, composition, etc. That is, it is a very rich set of functions!

   Your goal is to prove the following Time Hierarchy Theorem: "If $f(n)$ is a time constructible function then there exists a language that is decidable on a 1-tape Turing Machine in $O([f(n)]^2)$ time but is not decidable by any 1-tape Turing Machine in $o(f(n))$ time."

   Prove this result using the same approach that we used to prove the Space Hierarchy Theorem. (Note that here we are being specific about the fact that the TM's in question have just one tape because we can do things faster with two tapes than with one.)

2. [**35 Points**] **Gödel's Incompleteness Theorem!** In class, we did most of the proof of Gödel's Incompleteness Theorem, but we didn't complete a few important pieces of the "gadgetry" in the reduction. In particular, write formulae for the following pieces that we omitted. (You may use formulae that we developed in class to do this.)

   - digit(v, y, b): Assuming that we've already established that y is of the form p^i before using this formula, this formula should be True iff the contents at location i of v is the p-ary digit b. (We assumed this in formula in class and you'll need it again in your halt formula below.)

- halt(v, d): This formula should be True iff the number v encoding our configuration has a halting state somewhere. Recall that d here is a number p^L where L is the length of the configuration sequence v.
- match3$_\delta(v, y, z)$: This formula should be True iff the three p-ary digits of v at position y (immediately before, at, and immediately after position y) and the three p-ary digits of v at position z (immediately before, at, and immediately after position z) are consistent with the δ transition function of the given TM M. That is, if $y = p^i$ and $z = p^j$ then the three digits of v at indices $i - 1$, i, and $i + 1$ match their corresponding digits of v at positions $j - 1$, j, and $j + 1$, respectively. You may appeal to the set W_δ that we defined in class.
- Finally, where in our reduction was it important that p is prime? (In other words, what would have given us problems if p was not assumed to be prime?)