CSE396, Spr'19 Problem Set 10 (last) Due Thu. 5/9, 11:59pm

The **Final Exam** is on *Tuesday, May 14, 11:45am–2:45pm* in the lecture room, **Norton 112**. It is "open notes" but does not allow electronics and does not allow books. The notes should fit into one "binder"—which can be literally a 3-ring notebook binder or a similarly-sized carrying case for notes. Please also avoid bringing a backpack if you can—else you will be asked to stow it (in front where we can watch them). The essence of these conditions is fairness regarding (no) e-resources and not cramping one's neighbor.

The last week of class will finish up some material from section 5.1 and then do much of Chapter 7, taking however the "alternative proof" in Chapter 9 of the Cook-Levin Theorem. Chapter 6 is skipped, likewise sections 5.2. Lecture notes on the Cook-Levin proof are at https://www.cse.buffalo.edu/~regan/cse396/CSE396CookLevin.pdf

Homework—part online and all *individual work*—due **Thu.** 5/9, 11:59pm.

(1) Using *TopHat*, the "Worksheet" titled **Spr'19 HW10.1**. There are 10 questions, each worth 2 points, for 20 total. All are unique-answer questions with 1 attempt given. There is some dependence on material being treated fully in the last week of lectures, but the preambles to Q1 and Q6 in particular give a road map to it (including some things already mentioned).

(2) Show that the following decision problem is undecidable, expressly by reducing the A_{TM} problem to it. You may reference the fact that the *Turing Kit* program is written in Java.

INSTANCE: A Java program P.

QUESTION: Does P, when run on empty input, print the words "Hello World"?

Also, if we let H stand for the language of (the source codes of) programs P that do print "Hello World" when run, is H recognizable (i.e., computably enumerable)? Justify your answer. (12 + 6 = 18 pts.)

(3) Define $L = \{\langle M_1, M_2\rangle : M_1, M_2 \text{ are DTMs and } L(M_1) \cup L(M_2) = \Sigma^*\}$. Prove by two reductions that L is neither c.e. nor co-c.e. One reduction should be from A_{TM}, K_{TM}, or NE_{TM}, the other from D_{TM} or E_{TM}. (12 + 12 = 24 pts., for 62 total on the set)

Extra—not graded: Suppose we are given an NFA N with n states and alphabet $\Sigma = \{0, 1\}$. Can we always decide within $n^{O(1)}$ time whether $L(N) = 1^*$? First give a polynomial-time decider for the $L(N) \subseteq 1^*$ part. Then what about showing comprehensiveness? Something to note is that at a state p you might have arcs on 1 to an accepting state q and a rejecting state r. If the input string has just one more '1' then of course you need to go to q. But if it has two more 1's then you shouldn't disregard the option of going to r, because r might have an arc on '1' to an accepting state r' while q might not. So the problem might not be so easy—but could it be NP-hard like deciding $L(N) = (0 + 1)^*$? (This is tricky but food for thought; it will be addressed on the answer key.)