CS46 Homework 7

This homework is due at 11:59PM on Thursday, April 15. For this homework, you will work with a partner. It's ok to discuss approaches at a high level with other students, but most of your discussions should just be with your partner. Your partnership's write-up is your own: do not share it, and do not read other teams' write-ups. If you use any out-of-class references (anything except class notes, the textbook, or asking Joshua), then you **must** cite these in your post-homework survey. Please refer to the course webpage or ask me any questions you have about this policy.

Over the past few weeks, we've seen several ways of proving a language is decidable, including:

- Building a TM by *emulating* the operations of a DFA (or NFA, or regular expression or...).
- Building a TM by thinking algorithmically about the computational problem the language describes.
- Building a TM by using a TM we've built for other purposes.
- Using closure properties.

The main **learning goal** of this homework assignment is to put these techniques into practice by using them to show several languages are decidable.

Note: You must submit your solutions in a file named **hw7.tex**, and your submission must compile without errors using `pdflatex`. Any .pdf submissions will be ignored. Any .tex files not named hw7.tex, .tex files that don't compile, or not submitting a post-homework survey will earn up to a **-0.5** point deduction.

1. (Sipser 4.21) Prove that the following language is decidable:

$$\{\langle M \rangle \mid M \text{ is a DFA and } \forall w, \text{ if } w \in L(M) \text{ then } w^R \in L(M)\}$$

2. (Sipser 4.12) Let

$$A = \{\langle R, S \rangle | \text{R and S are regular expressions and } L(R) \subseteq L(S)\}$$

   Show that A is decidable.

3. (Sipser 3.19) Show that every infinite Turing-recognizable language has an infinite decidable subset.

4. **(extra credit)** (Sipser 3.17) Let $B = \{\langle M_1 \rangle, \langle M_2 \rangle, \ldots\}$ be a Turing-recognizable language consisting of Turing machine descriptions. Show that there is a decidable language C consisting of Turing machine descriptions such that every machine described in B has an equivalent machine in C, and every machine described in C has an equivalent machine in B.

5. **(extra credit)** A_{TM} is a language consisting of descriptions of Turing machines, and it is Turing-recognizable. Why does the previous question not imply that A_{TM} is decidable?