Assignment 4
Due: Thursday, July 25

1. [6 points] In this question you are asked to prove that two languages are decidable. Assuming that you do this by describing DSMs that decide these languages, feel free to use high-level descriptions—but be sufficiently precise about the algorithmic details of these machines so that someone with a lot of time and patience, but little creativity, could formally express them as DSMs.

(a) Prove that this language is decidable:
\[ A = \{ \langle \langle C \rangle, \langle D \rangle \rangle : C \text{ and } D \text{ are DFAs with } L(D) \subseteq L(C) \}. \]

(b) Prove that this language is decidable:
\[ B = \{ \langle D \rangle : D \text{ is a DFA and } L(D) \text{ is finite} \}. \]

2. [8 points] In this question you are asked to prove that two languages are semidecidable but not decidable.

(a) Prove that this language is semidecidable but not decidable:
\[ A = \left\{ \langle \langle M \rangle, \langle x \rangle \rangle : M \text{ is a DSM, } x \text{ is a string over the input alphabet of } M, \right. \]
\[ \text{and } M \text{ accepts at least one string } y \text{ with } y \leq x \left. \right\}. \]

(For this language, the notation \( u \leq v \) means that either \( u = v \) or \( u \) comes before \( v \) with respect to the lexicographic ordering of strings over the input alphabet of \( M \).)

(b) Prove that this language is semidecidable but not decidable:
\[ B = \{ \langle \langle M \rangle, \langle K \rangle \rangle : M \text{ and } K \text{ are DSMs with } L(M) \cup L(K) \neq \emptyset \}. \]

3. [5 points] Define a language \( A \) as follows:
\[ A = \{ \langle M \rangle : M \text{ is a DSM that runs forever on input } \varepsilon \}. \]

Prove that \( E_{\text{DSM}} \leq_m A \).
(The language \( E_{\text{DSM}} \) was defined in Lecture 17. Because we already proved that \( E_{\text{DSM}} \) is not semidecidable, a correct answer to this question establishes that \( A \) is also not semidecidable.)

4. [5 points] Prove that there cannot exist a DSM \( K \) that operates as follows:
- If the input to \( K \) is \( \langle M \rangle \), where \( M \) is a DSM that halts on all input strings and accepts infinitely many input strings, then \( K \) accepts.
- If the input to \( K \) is \( \langle M \rangle \), where \( M \) is a DSM that halts on all input strings and accepts only finitely many input strings, then \( K \) does not accept.

In other words, your goal is to prove that it is not possible for a DSM \( K \) to recognize that \( L(M) \) is infinite, for an input DSM \( M \), even if you are guaranteed that the input DSM \( M \) never runs forever.

5. [1 point] For each of the questions above, list the full name of each of your 360 classmates with whom you worked on that question. (If you didn’t work with anyone, that is fine: just indicate that you worked alone.)