Due: May 1

1. (a) \( \text{REJECT}_{TM} \) is defined as \( \{ \langle M, w \rangle | M \text{ is a Turing machine, and } M \text{ rejects } w \} \). Prove that \( \text{REJECT}_{TM} \) is Turing recognizable.

(b) Show that \( \text{REJECT}_{TM} \) is undecidable using diagonalization. Your proof should be similar to, but not the same as, the proof that \( A_{TM} \) is undecidable.

(c) Give a second proof that \( \text{REJECT}_{TM} \) is undecidable by reducing \( A_{TM} \) to \( \text{REJECT}_{TM} \).

[This will involve some creativity because the technique we used to reduce \( A_{TM} \) to \( \text{HALT}_{TM} \) will not work here.]

2. Let \( \text{CONTEXT-FREE}_{TM} = \{ \langle M \rangle | M \text{ is a Turing machine and } L(M) \text{ is context-free} \} \). Prove that \( \text{CONTEXT-FREE}_{TM} \) is not decidable by reducing \( A_{TM} \) to \( \text{CONTEXT-FREE}_{TM} \).

[Hint: Use a proof similar to the proof of Theorem 5.3.]

3. Let \( \text{NONREGULAR}_{TM} = \{ \langle M \rangle | M \text{ is a Turing machine, and } L(M) \text{ is not a regular language} \} \). Suppose that you want to reduce \( A_{TM} \) to \( \text{NONREGULAR}_{TM} \) by transforming \( \langle M, w \rangle \) to \( \langle M_2 \rangle \). (So if \( \langle M, w \rangle \) is in \( A_{TM} \), then \( \langle M_2 \rangle \) is in \( \text{NONREGULAR}_{TM} \), and if \( \langle M, w \rangle \) is not in \( A_{TM} \), then \( \langle M_2 \rangle \) is not in \( \text{NONREGULAR}_{TM} \).)

(a) Fill in the blanks in the following two statements in a way that states what you have to do to make the reduction work. Make your statements as general as possible. (In both cases you will be writing down something about the behavior of the Turing machine \( M_2 \).)

- If \( M \) accepts \( w \), then

- If \( M \) does not accept \( w \), then

(b) Give the definition of the desired Turing machine \( M_2 \), given \( M \) and \( w \).

4. Problem 5.9.


6. Problem 5.15.

7. Problem 5.27

[Hint: This is hard. First show that the emptiness problem for two-dimensional finite automata is undecidable, using computation histories, then reduce the emptiness problem to the equivalence problem.]