Due: At the beginning of lecture on Tuesday, July 4. **Worth**: 10%

1. [20 marks]

   (a) If $A \leq_m B$ and B is a regular language, does that imply that A is a regular language? Explain why or why not.

   (b) Let L be a finite language over an alphabet Σ. Is L always decidable? Prove your claim.

   (c) For an input string w, we define w^R to be the reverse of w. Use Rice's Theorem to prove that the following language is undecidable:

   $$T = \{\langle M \rangle : M \text{ is a Turing machine that accepts } w^R \text{ if and only if it accepts } w\}.$$

2. [15 marks]

   Let us now consider the "real-word" problem of protecting our computers from viruses. We would like to build a filter (a virus checker) which will detect programs which are viruses before they are executed. Unfortunately you will show that no virus checker can detect all viruses without itself being a virus. Of course, we must formalize what we mean by these terms.

   Consider a modern computer which uses some fixed operating system, under which all programs run. A *program* can be thought of as a function from strings to strings: it takes one string as input and produces another as output. On the other hand, a program itself can be thought of as a string.

   By definition, a program P spreads a *virus* on input x if running P with input x causes the operating system to be altered, and it is *safe* on input x if this doesn't happen. A program P is said to be *safe* if it is safe on every input string.

   A *virus checker* is a program, perhaps called `IsSafe`, that when given input $\langle P, x \rangle$, where P is a program and x is a string, produces the output 'YES' if P is safe on input x and 'NO' otherwise.

   Prove that if the possibility of a virus exists – i.e., there is a program and an input that would cause the operating system to be altered – then there can be no virus checker that is both safe and correct.

3. [10 marks]

   Prove that the following language is undecidable (you may not use Rice's Theorem):

   $$L_3 = \big\{\langle M, k \rangle : M \text{ is a Turing machine that accepts some string of length } k\big\}.$$

4. [15 marks]

   Consider the following language:

   $$S = \big\{\langle M \rangle : M \text{ is a Turing machine and } L(M) = \{\langle M \rangle\}\big\}.$$

   Show that neither S nor $\overline{S}$ is Turing-recognizable.