There are two required problems and one optional problem.

Recall that a *property of recognizable languages* is a set A of Turing machine encodings $\langle M \rangle$ such that, for any two Turing machines M and N with $L(M) = L(N)$, either $\langle M \rangle$ and $\langle N \rangle$ are both elements of A or neither $\langle M \rangle$ nor $\langle N \rangle$ is an element of A. In other words, whether a given Turing machine encoding $\langle M \rangle$ is an element of A depends only on the language $L(M)$ rather than some other aspect of M. There are two trivial properties:

- $\{\langle M \rangle : M$ is a Turing machine and M is a Turing machine$\}$, and
- $\{\langle M \rangle : M$ is a Turing machine and M is not a Turing machine$\}$.

The second trivial property is decided by a Turing machine that simply rejects all of its inputs. The first trivial property is decided by a Turing machine that accepts all valid Turing machine encodings and rejects all other inputs. So those two properties are decidable. Rice's theorem says: Every non-trivial property of recognizable languages is undecidable.

Consider Problems 5.9, 5.10, 5.11, 5.12, 5.13, 5.32a, 5.32b. In each of these problems, you are asked to prove that a language is undecidable.

A. Which of these seven undecidability results is an immediate consequence of Rice's theorem? (I am not asking you to prove the other ones.)

Let R be a Turing machine such that $L(R) = \emptyset$. In class we proved Rice's theorem under the assumption that $\langle R \rangle \notin A$. I asserted that the other case, where $\langle R \rangle \in A$, can be proved similarly.

B. Where does the proof given in class break, if A is not a property of recognizable languages, but merely some set of Turing machine encodings $\langle M \rangle$?

Problem C below is optional.

C. Prove Rice's theorem for the case where $\langle R \rangle \in A$.