Answer the questions below, paying particular attention to the logic of your arguments. Short descriptions for how a function might computed (on any machine we've talked about) are sufficient to show that it is computable.

1. Show that if A is recognizable and A reduces to $\bar{A}$, then A is decidable.

2. Let $B = \{<M> \mid M$ accepts exactly one of the strings 00 and 11$\}$.
   a. What does Rice's theorem say about B?
   b. Show that halting problem reduces to B.
   c. Show that halting problem reduces to the complement of B.
   d. Are B or its complement recognizable?

3. A language is *co-recognizable* if its complement is recognizable. Argue why each of the following languages is or is not recognizable and why it is or is not co-recognizable.
   a. $L_1 = \{<M> \mid M$ enters state q_{27} for some input string $x\}$.
   b. $L_2 = \{<M> \mid L(M)$ contains at most two strings}.
   c. $L_3 = \{<M> \mid L(M) \subseteq \Sigma^*\}$

4. A computable verifier is a deterministic Turing machine V that takes two arguments: x (the input) and y (the proof). A computable verifier always halts. Show that a language L is recognizable if and only if there exists a computable verifier V such that
   a. if $x \in L$, then there is a string y such that V(x,y) accepts, and
   b. if $x \notin L,$ then V(x,y) rejects for every string y.

5. Consider the following property: $P = \{<M> \mid L(M)$ is accepted by some Turing machine that has an odd number of states$\}$. Show that P is a trivial property.

6. [Bonus] Read about The Recursion Theorem in the Sipser text. One implication of the recursion theorem is that in any general purpose programming language, one can write code that outputs the code itself. Write a python program that prints its own code. Do not use any file operations.

https://www.udacity.com/course/viewer#!/c-ud557/l-1209378918/m-2986218594