1. Rice’s theorem states that no non-monotonic property of r.e. sets is semi-decidable.
   a. Give three examples of non-monotonic properties of r.e. sets.
   b. Let P be any non-monotonic property of r.e. sets and let \( L_1 \subseteq L_2 \) be r.e. sets such that \( L_1 \) has property P and \( L_2 \) does not. Let M be some TM and let w be any string. Show how to construct a new Turing machine \( M' \) such that:
      - If M halts on w then \( L(M') = L_2 \)
      - Otherwise \( L(M') = L_1 \)
   c. Show that recognizing \( L_{NHP} = \{M, w | M \text{ does not halt on } w \} \) reduces to the problem of recognizing \( L_P = \{M | L(M) \text{ has property } P \} \).

2. Determine whether the following sets are (a) recursive, (b) recursively enumerable but not recursive, or (c) not recursively enumerable. Prove your answers. (Do not use Rice’s theorem.)
   a. \( \{M | L(M) \text{ is context free} \} \)
   b. \( \{M | L(M) = L(M)^R \} \)
   c. \( \{M | M \text{ takes more than 20 steps on all input} \} \)
   d. \( \{M, i | M \text{ enter state } q_i \text{ on some input} \} \)

3. Prove that a language L is recursively enumerable iff there is a generator G that enumerates the strings in L.

4. Prove that a language L is recursive iff there is a generator G that enumerates the strings in L in canonical order.

5. Let \( L_1 \) and \( L_2 \) be recursive languages. Classify the following problems as decidable, semi-decidable, or undecidable. Prove your answers.
   a. Is \( L_1 \neq L_2 \)?
   b. Is \( x \) in \( L_1 \cap L_2 \)?
   c. Is \( L_1 = L_2 \)?