Problem 1. [Category: Proof] For each of the languages below indicate whether it is (a) decidable (a.k.a. recursive), (b) recognizable (a.k.a. recursively enumerable) but not decidable, or (c) not recognizable.

You may use the following facts:

- the language \( \text{SelfReject} = \{\langle M \rangle \mid M \text{ does not accept } \langle M \rangle \} \) is not recursively enumerable;
- the language \( \text{Accept} = \{\langle M, w \rangle \mid M \text{ accepts } w \} \) is recursively enumerable but not decidable.

- \( L_1 = \{\langle M, w \rangle \mid M \text{ does not halt on input } w \} \).
- \( L_2 = \{\langle M, w, n \rangle \mid M \text{ accepts } w \text{ within } n \text{ steps} \} \).
- \( L_3 = \{\langle M \rangle \mid M \text{ accepts } \langle M \rangle \} \).

Problem 2. [Category: Proof] We saw in class that the language \( \text{Accept} = \{\langle M, w \rangle \mid M \text{ accepts } w \} \) is undecidable. Use a reduction to argue that the language \( L = \{\langle M_1, M_2, w \rangle \mid \text{exactly one of } M_1 \text{ and } M_2 \text{ accepts } w \} \) is undecidable.

Problem 3. [Category: Proof] Can you show that \( L \) from the previous problem is not recognizable? Hint: In the previous problem you showed \( \text{Accept} \leq_m L \). Show that \( \overline{\text{Accept}} \leq_m L \) using similar ideas, and then use the fact that \( \overline{\text{Accept}} \) is not recognizable to complete the proof.