Tip:

This question is all about practicing reductions! Though it may seem a bit silly to keep re-proving that these languages are undecidable, working with familiar languages enables us to focus on mastering the **technique** of using reductions.

Exercise 1. Complete the following reductions:

(a) Given that $HALT$ is known to be undecidable, **prove** that A_{TM} is undecidable by showing that $HALT \leq_T A_{TM}$.

(b) Given that A_{TM} is known to be undecidable, **prove** that $HALT$ is undecidable by showing that $A_{TM} \leq_T HALT$.

(c) What do parts (a) and (b) tell you about the **relative difficulty** of $HALT$ and A_{TM}?

Exercise 2. In lecture, we proved that a language is Turing-recognizable **if and only if** it is enumerable. Given a recognizable language L and its corresponding recognizer R_L, identify and fix the flaw in the following (faulty) definition of its enumerator.

For each word `w` in Σ^*:
1. Run R_L on `w`.
2. If it accepts, output `w`.

Exercise 3. Consider the following language:

$$MIRROR = \{\langle M \rangle \mid M \texttt{ is a TM that accepts } w^R \texttt{ whenever it accepts } w\}$$

Show that $MIRROR$ is undecidable using a reduction.

Exercise 4. Prove that each of the following languages is undecidable:

(a) $INFINITE_{TM} = \{\langle M \rangle \mid M \text{ is a TM and } L(M) \text{ is an infinite language}\}$.

(b) $A_{1011} = \{\langle M \rangle \mid M \text{ is a TM and } 1011 \in L(M)\}$.

(c) $ALL_{TM} = \{\langle M \rangle \mid M \text{ is a TM and } L(M) \text{ is } \Sigma^*\}$.

References

[1] Sipser, Michael. *Introduction to the Theory of Computation.* Course Technology, 2005. ISBN: 9780534950972

[2] Critchlow, Carol and Eck, David *Foundation of computation.*, Critchlow Carol, 2011