Rice’s Theorem. Let \( \mathcal{L} \) be any set of languages that satisfies the following conditions:

- There is a Turing machine \( Y \) such that \( \text{Accept}(Y) \in \mathcal{L} \).
- There is a Turing machine \( N \) such that \( \text{Accept}(N) \notin \mathcal{L} \).

The language \( \text{AcceptIn}(\mathcal{L}) := \{ \langle M \rangle \mid \text{Accept}(M) \in \mathcal{L} \} \) is undecidable.

You may find the following Turing machines useful:

- \( M_{\text{ACCEPT}} \) accepts every input.
- \( M_{\text{REJECT}} \) rejects every input.
- \( M_{\text{HANG}} \) infinite-loops on every input.

Prove that the following languages are undecidable using Rice’s Theorem:

1. \( \text{AcceptRegular} := \{ \langle M \rangle \mid \text{Accept}(M) \text{ is regular} \} \)
2. \( \text{AcceptILLINI} := \{ \langle M \rangle \mid M \text{ accepts the string ILLINI} \} \)
3. \( \text{AcceptPALINDROME} := \{ \langle M \rangle \mid M \text{ accepts at least one palindrome} \} \)
4. \( \text{AcceptTHREE} := \{ \langle M \rangle \mid M \text{ accepts exactly three strings} \} \)
5. \( \text{AcceptUNDECIDABLE} := \{ \langle M \rangle \mid \text{Accept}(M) \text{ is undecidable} \} \)

To think about later. Which of the following languages are undecidable? How would you prove that? Remember that we know several ways to prove undecidability:

- Diagonalization: Assume the language is decidable, and derive an algorithm with self-contradictory behavior.
- Reduction: Assume the language is decidable, and derive an algorithm for a known undecidable language, like \( \text{Halt} \) or \( \text{SelfReject} \) or \( \text{NeverAccept} \).
- Rice’s Theorem: Find an appropriate family of languages \( \mathcal{L} \), a machine \( Y \) that accepts a language in \( \mathcal{L} \), and a machine \( N \) that does not accept a language in \( \mathcal{L} \).
- Closure: If two languages \( L \) and \( L' \) are decidable, then the languages \( L \cap L' \) and \( L \cup L' \) and \( L \setminus L' \) and \( L \oplus L' \) and \( L^* \) are all decidable, too.

6. \( \text{Accept}\{\varepsilon\} := \{ \langle M \rangle \mid M \text{ accepts only the string } \varepsilon; \text{ that is, } \text{Accept}(M) = \{ \varepsilon \} \} \)
7. \( \text{Accept}\emptyset := \{ \langle M \rangle \mid M \text{ does not accept any strings; that is, } \text{Accept}(M) = \emptyset \} \)
8. \( \text{Accept}\emptyset := \{ \langle M \rangle \mid \text{Accept}(M) \text{ is not an acceptable language} \} \)
9. \( \text{Accept=}\text{Reject} := \{ \langle M \rangle \mid \text{Accept}(M) = \text{Reject}(M) \} \)
10. \( \text{Accept}\neq\text{Reject} := \{ \langle M \rangle \mid \text{Accept}(M) \neq \text{Reject}(M) \} \)
11. \( \text{Accept}\cup\text{Reject} := \{ \langle M \rangle \mid \text{Accept}(M) \cup \text{Reject}(M) = \Sigma^* \} \)