Problem 1

Design a Turing machine that accepts \( L = \{ a^n b^n c^n \mid n \geq 0 \} \).

a. Give a brief, intuitive description of how your Turing machine works.

b. Give a formal description of your Turing machine by specifying \( Q \), \( F \), \( \Sigma \), and \( \Gamma \) and either writing out the transition function \( \delta \) or drawing a transition diagram.

For a guide to the following problems, see the solutions to similar problems in the textbook, e.g., 4.10, 4.12, and 4.14 on pages 213–4.

Problem 2

Prove or disprove: The following language is decidable:

\[ L = \{ \langle M, n \rangle \mid M \text{ is a Turing machine with } n \text{ states, and there exists a string } w \in \Sigma^* \text{ of length at most } n^2 \text{ such that } M \text{ accepts } w \text{ in at most } n^3 \text{ transitions} \} \]

Problem 3

Let \( A \) be a Turing-recognizable language consisting of descriptions of Turing machines, \( \{ \langle M_1 \rangle, \langle M_2 \rangle, \ldots \} \), where every \( M_i \) is a decider. Prove that some decidable language \( D \) is not decided by any decider \( M_i \) whose description appears in \( A \).

Hint: You may find it helpful to consider an enumerator for \( A \).

Problem 4

Determine which one of the following languages is decidable and which one is undecidable, where \( M \) is a Turing machine. Briefly justify your answer, either sketching how a Turing machine could decide the language or how you could solve a known undecidable problem if a decider existed for the language.

a. \( \{ \langle M, a \rangle \mid M \text{ writes a character } a \text{ when started on the empty tape} \} \)

b. \( \{ \langle M \rangle \mid M \text{ writes a non-blank character when started on the empty tape} \} \)