1. Give state diagrams (pictures) for Turing Machines that decide the following languages over the alphabet \{0,1\}:
   1. \{w \mid w \text{ contains an equal number of 0s and 1s}\}
   2. \{w \mid w \text{ does not contain twice as many 0s as 1s}\}.

2. A Turing machine that halts on all inputs is called a halting Turing machine (also known as Decider). Prove the following:
   (a) If \(M_1\) and \(M_2\) are two halting Turing machines, then there exists a halting Turing machine that recognizes \(L(M_1) \cap L(M_2)\).
   (b) If \(M_1\) and \(M_2\) are two (not necessarily halting) Turing machines, then there exists a Turing machine that recognizes \(L(M_1) \cap L(M_2)\).

3. A Turing machine with doubly infinite tape (TMDIT) is similar to an ordinary Turing machine except that its tape is infinite to the left as well as to the right. The tape is initially filled with blanks except for the portion that contains the input. Computation is defined as usual except that the head never encounters an end to the tape as it moves leftward.
   Show that the class of languages recognized by TMDITs is the same as the class of Turning-recognizable languages.

4. Show that a language is decidable if and only if some enumerator prints the strings in the language in lexicographical order.

5. Let \(L\) be a decidable language such that every string in \(L\) is the encoding of a Turing Machine that is a decider. Prove that there exists a decidable language that is not decided by any T.M. \(M\) with \(\langle M \rangle \in L\). That is, prove that there exists a language \(L'\) that is decidable and, for all Turing machines \(M\) with \(\langle M \rangle \in L\) we have that \(L(M) \neq L'\).