1 Decidability ($\star$)

1. Let L be a regular language. Then L is Turing-decidable.

   To prove this theorem, consider any regular language L. Then there is a DFA $M = (Q, \Sigma, \delta, q_0, F)$ for L. Give an explicit construction of a TM that decides L, based on the DFA M.

2. Let L be a context-free language. Show that L is Turing-decidable by giving an implementation-level description of a TM for L from a PDA for L.

2 Undecidability ($\star\star$)

Prove that the following languages are Turing-undecidable by giving reductions from the Halting Problem $L_H = \{\langle M, w\rangle \mid M \text{ halts on input } w\}$. In each case assume the alphabet $\Sigma = \{0, 1\}$.

1. $L_\emptyset = \{\langle M\rangle \mid L(M) = \emptyset\}$ (the language of all TMs that do not accept any words).
2. $L_\subseteq = \{\langle M_1, M_2\rangle \mid L(M_1) \subseteq L(M_2)\}$. (the language of all pairs of TMs M_1, M_2 such that every word accepted by M_1 is also accepted by M_2.)
3. $L_{st} = \{\langle M, w, q\rangle \mid q \text{ is a state of } M \text{ and } M \text{ enters state } q \text{ while computing on } w\}$
4. ($\star\star\star$) $L_{RM} = \{\langle M\rangle \mid L(M) \text{ is regular }\}$.

3 Closure properties ($\star$)

Which of the following are true? Prove the true statements and find counter-examples for the false ones.

1. If L is a Turing-decidable language and $L' \subseteq L$ then L' is Turing-decidable.
2. If L is a Turing-decidable language and $L \subseteq L'$ then L' is Turing-decidable.
3. If L is a Turing-decidable language then $\overline{L} = \Sigma^* \setminus L$ is Turing-decidable.