1. Count the number of functions \( f : A \rightarrow B \) from a finite set \( A \) of size \( m \) into a finite set \( B \) of size \( n \). Deduce that the number of (deterministic) Turing machines over a finite alphabet \( \Sigma \) using a finite state set \( Q \) is

\[
\left(3|Q||\Sigma|\right)^{|\left(Q\setminus 1\right)||\Sigma|}.
\]

How many Turing machines are there with

(a) one non-halting state and one non-blank letter,
(b) one non-halting state and two non-blank letters,
(c) two non-halting states and one non-blank letter.
(d) two non-halting states and two non-blank letters.

2. Consider the following sets of instructions for Turing machines. Draw corresponding state diagrams and describe the behaviour of the given Turing machine when starting with a tape which is all blank except for a string of \( n \) consecutive 1’s, scanning the left-most 1 on the tape.

(a) \( X = \{ q_1 \ b \ b \ C \ q_0, \ q_1 \ \ b \ R \ q_2, \ q_2 \ b \ 1 \ C \ q_0, \ q_2 \ b \ R \ q_1 \} \).
(b) \( X = \{ q_1 \ b \ b \ C \ q_0, \ q_1 \ b \ R \ q_2, \ q_2 \ b \ 1 \ C \ q_2, \ q_2 \ b \ R \ q_1 \} \).
*(c) \( X = \{ q_1 \ b \ b \ C \ q_0, \ q_1 \ b \ R \ q_2, \ q_2 \ b \ 1 \ L \ q_2, \ q_2 \ b \ R \ q_1 \} \).

3. Design a Turing machine that inputs a consecutive string of 1’s, scanning the left-most 1 and halts on a blank tape if the number of 1’s is a multiple of 3, halts on a tape with a single 1 if that number is a multiple of 3 plus 1, and halts with two consecutive 1’s if that number is a multiple of 3 plus 2.

4. Show that if a task can be performed by a Turing machine \( M \) then there exists a Turing machine \( M' \) that performs the same task, but with instructions that avoid the move symbol \( C \).

5. Design a Turing machine that inputs a string \( 1 \ldots 1 * 1 \ldots 1 \) of consecutive 1’s broken in the middle by an asterisk, initially scanning the left-most 1, erases the asterisk, and outputs a consecutive string of 1’s, with the same total number of 1’s as the first string, scanning the left-most 1 when it halts. Interpret this computation in terms of addition of positive integers.

6. Design a Turing machine that inputs a string \( 1 \ldots 1 * 1 \ldots 1 \) of \( m \) consecutive 1’s, followed by an asterisk, followed by \( n \) consecutive 1’s, where \( m \) and \( n \) are positive integers, initially scanning the left-most 1, erases the asterisk, and outputs a string of \( mn \) consecutive 1’s, scanning the left-most 1 when it halts. (This “computes” positive integer multiplication.)