1. Let \( L \) be the language accepted by the NFA shown on the right. Find NFAs which accept (a) \( L^c \) (b) \((L^c)^2\).

2. (a) Find an NFA which is equivalent to the RLG given below.


(b) Convert the NFA shown below on the right in Qu.#3 into an equivalent RLG.

3. (a) Find a regular expression for the language accepted by the NFA shown on the right.

(b) Write down what the Halting Problem says and define what is the Busy beaver function.

4. (a) Define what are the operations known as composition and primitive recursion.

(b) Show that \( f(x,y) = 2x+3y+4 \) is a primitive recursive function by finding primitive recursive functions \( g \) and \( h \) such that \( f = \text{prim.
rec.} (g,h) \).

[You must show that your \( g \) and \( h \) are primitive recursive.]

5. (a) Define what is a Turing computable function with domain \( D \).

(b) Show what happens at each step if 10101 is the input for the TM, \( M \) shown on the right.

(c) Find the language accepted by \( M \).

6. Determine which of the following languages are regular and which are not.

(a) \( L_1 = \{a^k \cdot b^n : n+2 = k^2 \pmod{3}\} \)

(b) \( L_2 = \{b^k \cdot c^n : n+2 > k^2 \} \)

[If you say that it is regular, you must find a regular expression for it; if you say it is non-regular, you must give a complete proof.]
1. \( R(\lambda) = \{ B \} \rightarrow \{ B, C \} \rightarrow \{ B, C \} \rightarrow B \)

\( M_D = \)

\( \text{NFA for } L^c \)

\( \text{OAS-NFA for } L^c \)

\( \text{NFA for } (L^c)^R \)

2. (a)

\( \rightarrow B, \; B \rightarrow \{ B \}, \; B \rightarrow \{ C \}, \; C \rightarrow \{ D \}, \; C \rightarrow \{ A \} \)

\( A \rightarrow \{ B \}, \; D \rightarrow \{ E \}, \; E \rightarrow \{ A \}, \; E \rightarrow \{ C \}, \; E \rightarrow \lambda \).

2. (b)

3. (a) Elim. D:

\( \text{Elim. A:} \)

\( \text{Elim. C:} \)

\( \text{Ans:} \; x_1^* \cdot y_2 \cdot (\tau_4 + y_3 \cdot y_2^*)^* \)

\( = (1 + 010) \cdot 011 \cdot (011 + (011 + 010) \cdot (1010)^*)^* \)

(b) The Halting problem asks if there is a TM H such that for any arbitrary TM M and an arbitrary input w, H will halt on \(<M, w>\) in an accepting state, if M halts on w, and H will halt on \(<M, w>\) in a non-accepting state, if M does not halt on w.

The Busy-beaver function is defined by \( B(n) = \max \) no. of 1s a TM in \( H_n \) can produce. Here \( H_n \) is the set of all TMs with tape alphabet \{0, 1\} & \( n \) states which halts on the blank tape.
4. (a) Composition is the operation that produces a function \( f : \mathbb{N}^n \to \mathbb{N} \) from \( g_1, \ldots, g_k : \mathbb{N}^n \to \mathbb{N} \) and \( h : \mathbb{N}^k \to \mathbb{N} \) by putting \( f(\bar{x}) = h(g_1(\bar{x}), \ldots, g_k(\bar{x})) \). Here \( \bar{x} = \langle x_1, x_2, \ldots, x_n \rangle \).

   (b) \( f(x, y) = 2x + 3y + 4 \). If \( f = \text{prim. rec.}(g, h) \) then
       \[ f(x, 0) = 2x + 4 \iff g(x), \quad g(y) = 2y + 4, \quad g(0) = 0 \]
       \[ f(x, y+1) = 2x + 3(y+1) + 4 \iff g(y+1) = 2(y+1) + 4 \]
       \[ = (2x + 3y + 4) + 3 \iff g(y) + 2 \]
       \[ = h(x, y + 1) + 3 \iff h(x, y, f(x, y)) \]
       \[ \therefore h = \text{so}_3 \circ \text{so}_1 \quad \text{and} \quad g = \text{prim. rec.}(\text{so}_3 \circ \text{so}_1, \text{so}_3 \circ \text{so}_1) \]

5. (a) \( L \) is Turing computable if we can find a TM \( M \) such that for each \( w \in \mathbb{D} \), \( \langle q_0, w \rangle \vdash^* \langle q_f, f(w) \rangle \) is a halted computation in \( M \) with \( q_f \in \mathbb{A}(M) \).

   (b) \( \langle A, \langle 0, 1, 0 \rangle \rangle \vdash \langle C, \langle 0, 0, 0 \rangle \rangle \vdash \langle B, \langle 0, 0, 0 \rangle \rangle \vdash \langle E, \langle 0, 0, 0 \rangle \rangle \vdash \langle B, \langle 0, 0, 0 \rangle \rangle \vdash \langle E, \langle 0, 0, 0 \rangle \rangle \) \( \vdash^* \langle D, \langle 0, 0, 0 \rangle \rangle \vdash^* \langle D, \langle 0, 0, 0 \rangle \rangle \) \( \vdash^* \langle D, \langle 0, 0, 0 \rangle \rangle \).

6. (a) \( L_1 = \{ a^k b^n : n+2 \equiv k^2 \pmod{3} \} \) is regular because \( (\text{aa}^*)^* (\text{bb})^* b + (\text{aa})^* a (\text{bb})^* b b + (\text{aa})^* a (\text{bb})^* b b \) is a regular expression for \( L_1 \).

   (b) \( L_2 \) is not regular. Suppose \( L_2 \) was regular. Then we can find an NFA \( M \) such that \( L(M) = L_2 \). Let \( N \) be the number of states in \( M \) and consider the string \( b^N c^{N^2} \). Since \( n+2 = N^2 + 2(N^2) = N^2 \), \( M \) accepts \( b^N c^{N^2} \).

Since if it takes \( N+1 \) states to process \( b^N \), the acceptance track must contain a loop as shown below, with \( j > 1 \).

\[ \xymatrix{ b \ar@/^/[r] & b^{N-j} \ar@/^/[r] \ar@/^/[l] & b^j \ar@/^/[r] & \cdots \ar@/^/[r] \ar@/^/[l] & b^{N-j} \ar@/^/[r] \ar@/^/[l] & c^{N^2} \ar@/^/[r] & \cdots \ar@/^/[r] \ar@/^/[l] & 2^{N+j} \}
\]

If we were the loop twice, we will see that \( M \) accepts \( b^{N+2j} c^{N^2} \).

But this contradicts \( L(M) = L_2 \) because \( n+2 = N^2 + 2(N^2) = N^2 \). So \( L_2 \) is really non-regular.