1. (a) Find an NFA, $M$, which is equivalent to the RLG $G$ given below.

\[
G: \quad \rightarrow E, \quad E \rightarrow 01, \quad E \rightarrow 01B, \quad B \rightarrow 0C, \quad C \rightarrow 11, \quad C \rightarrow 0D, \quad C \rightarrow E, \quad D \rightarrow 1B, \quad D \rightarrow \lambda.
\]

(b) Find an RLG, $G$, which is equivalent to the NFA in Problem 2 below.

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

(b) Define what is the Busy Beaver function.

3. (a) Define what it means for $f$ to be obtained from $g$ and $h$ by primitive recursion.

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

4. (a) Define what it means for the function $f: \mathbb{N}^n \rightarrow \mathbb{N}$ to be obtained from the total function $g: \mathbb{N}^{n+1} \rightarrow \mathbb{N}$ by minimization.

(b) Let $f(x) = \lfloor x/4 \rfloor$ and $h(x) = 0$ (if $x$ is a multiple of 4) & $h(x) = 1$ (otherwise). Show that $f$ and $h$ are $\mu$-recursive functions.

[You may use the fact that PRED, MONUS, ADD, MULT & SIGN are primitive recursive if needed in #4, but you are not allowed to do so in #3.]

5. (a) Define what is a Turing semi-decidable binary relation $R$ on $\mathbb{N}$.

(b) Show what happens at each step if (i) $\lambda$ and (ii) $1$ are the inputs for the TM, $M$, shown on the right.

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

(a) $L_1 = \{a^k.b^n : k \text{ (mod 3)} > 2 + n^2 \text{ (mod 3)}\}$

(b) $L_2 = \{b^k.a^n : k > 2 + n^2\}$.

[If you say that it is regular, you must find a regular expression; if you say it is non-regular, you must give a complete proof.]
1(a)  
(b) $B \rightarrow bC$, $C \rightarrow cD$, $D \rightarrow aB$, $D \rightarrow bE$, $E \rightarrow \lambda$.

2(a)  
Eliminate $A$:  
\[ L(M) = R_1^* R_2^* (R_4 + R_3 R_1^* R_0)^* = (bca)^* bc b (aab + (ac + aab)) (bc b)^* bc b \]  

(b) $p(n)$ is the maximum number of 1's a TM with $n$ states can produce when started on the blank tape. Here $H_n$ is the set of TMs with $n$ states and tape alphabet $\{1, \lambda\}$ which halt on the blank tape.

3(a)  
$f: \mathbb{N}^n \rightarrow \mathbb{N}$ is said to be obtained from $g: \mathbb{N}^n \rightarrow \mathbb{N}$ and $h: \mathbb{N}^{n+2} \rightarrow \mathbb{N}$ by primitive recursion if $f(x, 0) = g(x)$ and $f(x, s(y)) = h(x, y, f(x, y))$. Here $x = (x_1, \ldots, x_n)$.

(b) $f(x, y) = 2x + 3y + 3$. So $f(x, 0) = 2x + 3 \Rightarrow g(x) = 2x + 3$, and $f(x, s(y)) = 2x + 3(y+1) + 3 = f(x, y) + 3 \Rightarrow h(x, y, f(x, y)) = f(x, y) + 3$, i.e. $f = \text{prec}(g, h)$ and $h = \text{so} s o s o I_{3,3}$. Now $g(y) = 2y + 3$. So $g(0) = 3$ and $g(s(y)) = 2(y+1) + 3 = g(y) + 2$. Hence $g = \text{prec}(s o s o I_{0,0}, s o s o I_{2,2})$. Thus $f = \text{prec}(g, h) = \text{prec}(\text{prec}(s o s o I_{0,0}, s o s o I_{2,2}), s o s o I_{3,3})$ and so $f$ is primitive recursive.
4(a) \( f: \mathbb{N} \to \mathbb{N} \) is obtained from the total function \( g: \mathbb{N}^{n+1} \to \mathbb{N} \) if
\[
f(x) = \begin{cases} 
\text{smallest value of } y \text{ such that } g(x, y) = 0 \text{ where } x = \langle x_1, \ldots, x_n \rangle, \\
\text{undefined, if } y \not\in \mathbb{N} \text{ for all } x \in \mathbb{N}.
\end{cases}
\]
(b) Let \( g(x, y) = x \div y \). Then \( \mu(y)[g(x, y) = 0] = (x/4)^y = f(x) \). So \( f = \mu[g, 4] = \mu[\text{MONUS} \circ [I_{12} \land \text{MULTO} \\
(sosososososososososososososososososoz_{12, 2}) \land I_{2, 2}], 0] \)
\:. \( f \) is \( \mu \)-recursive.
(c) \( h(x) = \text{SIGN}(4, [x/4] - x) = \text{SIGN}(4, f(x) - x) \). So \( h = \text{SIGN} \circ [\text{MONUS} \circ [\text{MULTO} [sosososososososososososososososososoz_{12, 2}], I_{1, 1}]] \). So \( h \) is \( \mu \)-recursive.

5(a) \( R \) is Turing semi-decidable if we can find a TM \( M \) such that \( M \) halts on \( \langle m, n \rangle \) in an accepting state or fails to halt, if \( \langle m, n \rangle \not\in R \).
(b) \( \langle A, \bot \rangle \rightarrow \langle B, \bot \rangle \rightarrow \langle C, \bot \rangle \rightarrow \langle D, \bot \rangle \rightarrow \langle E, \bot \rangle \) halts
\( \langle A, 1 \rangle \rightarrow \langle A, x \rangle \rightarrow \langle B, x \rangle \rightarrow \langle C, 1 \rangle \rightarrow \langle B, 11 \rangle \rightarrow \langle B, 111 \rangle \rightarrow \langle D, 111 \rangle \rightarrow \langle E, 111 \rangle \) halts

6(a) \( n \equiv 0 \pmod{3} \Rightarrow k \equiv (2 + 0^2) \pmod{3} \Rightarrow \) no possible value of \( k \)
\( n \equiv 1 \pmod{3} \Rightarrow k \equiv (2 + 1^2) \pmod{3} \Rightarrow k \equiv 1 \pmod{3} \)
\( n \equiv 2 \pmod{3} \Rightarrow k \equiv (2 + 2^2) \pmod{3} \Rightarrow k \equiv 2 \pmod{3} \), so
\( (a+aa)(aa)^*b(bb)^*+(aa)^*(a+aa)(aa)^*bb(bb)^*)^* \) describes \( L \). So \( L \) is reg.
(b) Suppose \( L_2 = \{ b^k a^n : k > 2n^2 \} \) was regular. Then we can find a \( \lambda \)-free NFA \( M \) with \( N \) states such that \( L(M) = L_2 \). Now
\( b^{n+3} a^N \in L_2 \) because if we take \( k = N^2 + 3 \) and \( N = N \), then \( k > 2n^2 \).
So \( M \) will accept \( b^{N+3} a^N \). Since it takes \( N+1 \) states to process the \( a^N \) part of the string, any acceptance track of \( b^{N+3} a^N \)
must have a loop as shown below.

\[\begin{array}{c}
q_0 & \overset{a^j, j \geq 1}{\longrightarrow} & a^{N-E-j} \\
& & \overset{N-E-j}{\rightarrow}
\end{array}\]

Now if we ride the loop twice, we will see that \( M \) accepts the string \( b^{N+3} a^{N+1} \). But \( N^2 + 3 > 2 + (N+1)^2 \), so this contradicts the fact that \( L(M) = L_2 \). Hence \( L_2 \) is a non-regular language.