Languages and Computation (COMP 2049) Lab 05

Properties of Regular Languages

(1) Consider the alphabet $\Sigma = \{0, 1\}$. In this exercise, we interpret every string $w \in \Sigma^*$ as the binary representation of a natural number. For simplicity, we ignore the leading zeros. For instance, the strings 0011 and 11 both represent the natural number 3.

(a) Draw the transition graph of a deterministic finite automaton (DFA) $M = (Q, \Sigma, \delta, q_0, F)$ which accepts the set $\{2k + 1 \mid k \in \mathbb{N}\}$ of odd natural numbers in binary format.
(Hint: Two states should be enough: one for even, one for odd.)

(b) Draw the transition graph of a DFA $N = (P, \Sigma, \delta', p_0, G)$ which accepts the set $\{3k + 1 \mid k \in \mathbb{N}\}$ of natural numbers that have remainder 1 when divided by 3, in binary format.
(Hint: Three states should be enough, corresponding to the remainder of division by 3, which coule be 0, 1, or 2.)

(c) Draw the transition graph of a DFA M_1 that accepts the set of odd natural numbers which have remainder 1 when divided by 3, in binary format, i. e., $L(M_1) = L(M) \cap L(N)$.

(d) Draw the transition graph of a DFA M_2 that accepts the set of odd natural numbers which do not have remainder 1 when divided by 3, in binary format, i. e., $L(M_2) = L(M) - L(N)$.

(e) Implement all the above DFAs M, N, M_1, and M_2, in JFLAP. In particular:

(i) Use JFLAP to arrange the nodes of M_1 and M_2 in such a way that the graph is planar, i. e., the edges do not intersect.

(ii) Try the DFAs on various input values of your choice:

- In JFLAP, you may go to the "`Input`" tab, and then select "`Multiple Run`".
- To enter the empty string λ, click on "`Enter Lambda`" in the bottom right corner.

Some sample values are provided in Table 1. Make sure you also try binary input values with leading zeros.

Table 1 Sample input values: Some binary values are given with leading zeros (e. g., 00111 instead of just 111 for 7) to make sure that these cases are also handled correctly by the DFAs.

Input (Binary)	Decimal	M $(2k + 1)$	N $(3k + 1)$	M_1 (Intersection)	M_2 (Set Difference)
λ	–	Reject	Reject	Reject	Reject
11	3	Accept	Reject	Reject	Accept
00111	7	Accept	Accept	Accept	Reject
1010	10	Reject	Accept	Reject	Reject
01100	12	Reject	Reject	Reject	Reject
001101	13	Accept	Accept	Accept	Reject

(2) We have learned that the class of regular languages is closed under finite unions. In other words, if $L_1, \ldots, L_n$ are regular languages, then $\bigcup_{i=1}^{n} L_i$ is also a regular language. Now consider the following claims:

(a) For any infinite family $L_0, L_1, \ldots, L_n, \ldots$ of regular languages, the union $\bigcup_{i\in\mathbb{N}} L_i$ is also regular.

(b) If L_1 and L_2 are non-regular languages, then $L_1 \cup L_2$ is also non-regular.

For each claim, determine whether it is true or false. If true, then a proof must be presented. If false, then a counterexample must be provided.