Sample solution HW 1

(2.5 points) Question 1. Construct an NFA for each of the following languages, where $\Sigma = \{a, b\}$.

(a) The language $L_1$ that consists of all the words in which $b$ appears at least twice.

(b) The language $L_2$ that consists of all the words that starts with $bb$ and ends with $ab$.

(c) The language $L_3$ that consists of all the words that contains $aba$.

For example: $aba$ and $ababaa$ are in $L$, since they contain $aba$. On the contrary, $aaaaa$ and $abbabbabb$ are not in $L$, since they do not contain $aba$.

(d) The language $L_4$ that consists of all the words that do not contain $bb$.

(e) The language $L_5$ that consists of all the words $w$ such that if $w$ contains $bb$, then $w$ ends with $ab$.

Solutions for question 1.

(1.a)

(1.b)

(1.c)

(1.d)
(1.e) Define the following two languages.

\[ K_1 = \{ w \mid w \text{ does not contain } bb \} \]
\[ K_2 = \{ w \mid w \text{ contains } bb \text{ and ends with } ab \} \]

Note that \( L_5 = K_1 \cup K_2 \), and \( K_1 \) is the language in question (1.d). We can construct an NFA for \( K_2 \) as follows.

From here, we can easily construct an NFA (with \( \epsilon \)-move) for the language \( K_1 \cup K_2 \) as follows.
(2.5 points) Question 2. Construct the regular expression for each of the languages above.

Solutions for question 2. In the following $\Sigma = \{a, b\}$.

(1.a) $a^*ba^*b\Sigma^*$.

(1.b) $bb\Sigma^*ab$.

(1.c) $\Sigma^*aba\Sigma^*$.

(1.d) $a^*(baa^*)^*(b \cup \emptyset^*)$. Intuitively, the regex means that every $b$ must be followed by at least one $a$, unless it is the last one.

(1.e) $e_1 \cup \Sigma^*bb\Sigma^*ab$, where $e_1$ is the regex in (1.d).
(2 points) Question 3. A string $w \in \{0, 1\}^*$ represents an integer in a standard way. For example, the string 000 represents the integer 0, and so do 0 and 000000. The string 00100 and 100 both represent the integer 4.

Construct a DFA for the following language over the alphabet $\{0, 1\}$:

$$L_0 := \{w \mid w \text{ represents an integer divisible by } 3\}$$

Hint: Consider $(2i + j) \mod 3$, for every $i \in \{0, 1, 2\}$ and $j \in \{0, 1\}$.

Solution for question 3. Note that if a word $w \in \{0, 1\}^*$ represents an integer $N_w$, then $w0$ and $w1$ represent the integer $2 \cdot N_w$ and $2 \cdot N_w + 1$, respectively.

Now, calculate the following:

$$2 \cdot 0 + 0 \equiv 0 \mod 3 \quad 2 \cdot 0 + 1 \equiv 1 \mod 3$$
$$2 \cdot 1 + 0 \equiv 2 \mod 3 \quad 2 \cdot 1 + 1 \equiv 0 \mod 3$$
$$2 \cdot 2 + 0 \equiv 1 \mod 3 \quad 2 \cdot 2 + 1 \equiv 2 \mod 3$$

Then, we can construct a DFA for the language $L_0$ with three states $p_0, p_1, p_2$ corresponding to 0, 1, 2, respectively.

In the above, we assume that $\epsilon$ represents 0, which is divisible by 3.
If you insist that $\epsilon$ shouldn’t be in $L_0$, the following DFA is also acceptable.
Question 4. For a language $L \subseteq \Sigma^*$ (not necessarily regular), we define the equivalence relation $\sim_L$ on $\Sigma^*$, where $u \sim_L v$ if for every $w \in \Sigma^*$, $uw \in L$ if and only if $vw \in L$.

Equivalently, we can say that $u \sim_L v$, if one of the following holds.

- Both $uw$ and $vw$ are in $L$.
- Both $uw$ and $vw$ are not in $L$.

(a) (1 points) Prove that $\sim_L$ is an equivalence relation.

(b) (2 points) In the following, let $(\#(\sim_L))$ (read: the index of $\sim_L$) denote the number of equivalence classes of $\sim_L$.

Prove that $L$ is a regular language if and only if $(\#(\sim_L))$ is finite.

Solutions for question 4. We will show that $\sim_L$ is an equivalence relation.

- Reflexive: For every $u \in \Sigma^*$, $u \sim_L u$.
  
  It is rather trivial. For every $w \in \Sigma^*$, $uw \in L$ if and only if $uw \in L$, hence, $u \sim_L u$.

- Symmetric: For every $u, v \in \Sigma^*$, if $u \sim_L v$, then $v \sim_L u$.
  
  This is also rather trivial. Suppose $u \sim_L v$, which means for every $w \in \Sigma^*$, $uw \in L$ if and only if $vw \in L$, which is equivalent to $vw \in L$ if and only if $uw \in L$. Therefore, $v \sim_L u$.

- Transitive: For every $u, v, x \in \Sigma^*$, if $u \sim_L v$ and $v \sim_L x$, then $u \sim_L x$.
  
  Suppose $u \sim_L v$ and $v \sim_L x$. This means:
  
  - For every $w \in \Sigma^*$, $uw \in L$ if and only if $vw \in L$.
  - For every $w \in \Sigma^*$, $vw \in L$ if and only if $xw \in L$.

  Thus, for every $w \in \Sigma^*$, $uw \in L$ if and only if $xw \in L$. Therefore, $u \sim_L x$.

Now we show the second part. We start with the “only if” part. Let $L$ be a regular language and $A$ be its DFA.

For a word $w$, we denote by $A(w)$ the state of $A$ after reading $w$. Or, more formally, if $w = a_1 \cdots a_n$ and $q_0 a_1 q_1 \cdots a_n q_n$ is the run of $A$ on $w$, then $A(w) = q_n$.

We will first prove the following:

Claim 1 For every words $u, v$, if $A(u) = A(v)$, then $u \sim_L v$.

Proof. Let $u$ and $v$ be such that $A(u) = A(v)$. Let $u = a_1 \cdots a_n$ and $v = b_1 \cdots b_m$.

We have to show that for every $w \in \Sigma^*$, $uw \in L$ if and only if $vw \in L$. Let $w = c_1 \cdots c_k$.

Consider the run of $A$ on $uw$:

$$p_0 \ a_1 \ p_1 \ \cdots \ a_n p_n \ c_1 r_1 \ \cdots \ c_k r_k$$

Likewise, consider the run of $A$ on $vw$:

$$s_0 \ b_1 \ s_1 \ \cdots \ b_m s_m \ c_1 t_1 \ \cdots \ c_k t_k$$

Here both $p_0, s_0$ is the initial state of $A$. Since $A(u) = A(v)$, we have $p_n = s_m$. Furthermore, $A$ is deterministic. Thus, $r_1 = t_1, \ldots, r_k = t_k$, and therefore,

$$A(uw) = A(vw)$$
This completes proof of Claim 1. ■

Claim 1 immediately implies that \(#(\sim_L) \leq |Q|\), where \(Q\) is the set of states of \(\mathcal{A}\). Thus, \(#(\sim_L)\) is finite.

Now, we show the “if” direction. Let \(L\) be a language over \(\Sigma\), where \(\sim_L\) has finitely many equivalence classes \(C_1, \ldots, C_m\). Without loss of generality, we can assume that \(L \neq \emptyset\).

We first prove the following claim.

**Claim 2** There is \(i_1, \ldots, i_k \subseteq \{1, \ldots, m\}\) such that \(L = C_{i_1} \cup \cdots \cup C_{i_k}\). In other words, \(L\) is a union of some of the equivalence classes of \(\sim_L\).

**Proof.** Note that if \(w \sim_L v\), then either both of them belong to \(L\), or both of them do not belong to \(L\). Thus, Claim 2 follows immediately. ■

Now, consider the following DFA \(\mathcal{A} = (\Sigma, Q, q_0, F, \delta)\).

- \(Q = \{p_1, \ldots, p_m\}\), i.e., the number of states is precisely the number of equivalence classes in \(\sim_L\).
- \(q_0\) is \(p_j\), where \(j\) is such that \(\epsilon \in C_j\).
- \(F = \{p_{i_1}, \ldots, p_{i_k}\}\), where \(i_1, \ldots, i_k\) are the indices in (5.a).
- \(\delta : Q \times \Sigma \to Q\) is defined as follows. For every \(p_i \in Q\), for every \(a \in \Sigma\), we pick an arbitrary \(w \in C_i\), and define \(\delta(p_i, a) = p_j\), where \([wa] = C_j\).

Note that \(\delta\) is a well-defined function, i.e., for every \(w_1, w_2 \in C_i\), \([w_1a] = [w_2a]\). In other words, the end result \(p_j\) remains the same for whichever \(w\) we pick, as long as \(w\) is from \(C_i\).

We will show that \(L(\mathcal{A}) = L\). Recall that \(\mathcal{A}(w)\) is the state of \(\mathcal{A}\) after reading \(w\) starting from the initial state. From the construction of \(\mathcal{A}\), for every word \(w \in \Sigma^*\), if \([w] = C_j\), then \(\mathcal{A}(w) = p_j\). Now,

\[w \in L \text{ if and only if } w \in C_{i_1} \cup \cdots \cup C_{i_k}\]

and hence,

\[w \in C_{i_1} \cup \cdots \cup C_{i_k} \text{ if and only if } \mathcal{A}(w) \text{ is one of } p_{i_1}, \ldots, p_{i_k}\].

Thus, \(w \in L\) if and only if \(w \in L(\mathcal{A})\), and hence, \(L = L(\mathcal{A})\).

---

*Here, for \(w \in \Sigma^*\), \([w]\) denotes the equivalence class of \(\sim_L\) that contains \(w\). See the notation in Lecture 1.*