COMPSCI 3331 - Midterm review partial solutions

Fall 2022

These questions are provided for review purposes and are not guaranteed to be the same or similar to the questions on the midterm. Some questions may be easier than the questions on the midterm and some may be harder.

These solutions may be updated with further solutions if time permits.

1. Disprove that for words w, x, that if $wx = xw$ then $x = w$.

For any word z and any integers i, j, if we let $x = z^i$ and $w = z^j$, then $xw = wx = z^{i+j}$, which disproves the statement. For instance, let $x = abab$ and $w = ab$.

2. Prove or disprove that for languages $L_1, L_2 \subseteq \Sigma^*$, where Σ has at least two letters, $\overline{L_1 L_2} = \overline{L_1} L_2 \cup L_1 \overline{L_2}$.

The statement does not hold. Let $\Sigma = \{a, b\}$. Let L_1, L_2 be the following languages:

$$\begin{aligned} L_1 &= (a+b)^* \\ L_2 &= (a+b)^+ \end{aligned}$$

Recall that $(a+b)^+$ is just a short-hand for $(a+b)(a+b)^*$, i.e., $\varepsilon \notin (a+b)^+$.

Then we can verify these equalities

$$\begin{aligned} \overline{L_1} &= \emptyset \\ \overline{L_2} &= \{\varepsilon\} \\ L_1 L_2 &= (a+b)^+ \\ \overline{L_1 L_2} &= \{\varepsilon\} \\ \overline{L_1} L_2 &= \emptyset \\ L_1 \overline{L_2} &= (a+b)^* \\ \overline{L_1} L_2 \cup L_1 \overline{L_2} &= (a+b)^* \end{aligned}$$

Thus, the equality does not hold.

3. Prove or disprove that for languages $L_1, L_2, L_3 \subseteq \Sigma^*$, where Σ has at least two letters, that $L_1(L_2 \cap L_3) = (L_1L_2) \cap (L_1L_3)$.

4. Show that the following languages are regular.

(a) $L = \{w \in \{a,b\}^* \ : \ |w|_a = 0 \text{ or } |w|_b \geq 5\}$.

(b) $L = \{w \in \{a,b\}^* \ : \ aa \text{ does not appear in } w\}$.

(c) $L = \{w \in \{0,1\}^* \ : \ w \text{ is divisible by 4 when interpreted as a binary number } \}$ (assume that the numbers are big-endian).

(d) $L = \{w \in \{a,b,c\}^* \ : \ \exists x \in \{a,b,c\}, u \in \{a,b,c\}^* \text{ such that } w = xux\}$.

For extra practice, try to develop multiple models for each of these languages (DFAs, NFAs, regular expressions). However, some may be difficult to do.

(c) The words that are divisible by four in binary are those that end in 00 (since we assume that the numbers are big-endian, the numbers **end** with 00).

So a regular expression for this language is $1(0+1)^*00$. We add a 1 at the front as all binary numbers start with one.

5. For each of the following regular expressions, construct an ε-NFA corresponding to the regular expression.

(a) $(a+b)(a+b)(a+b)^*$.

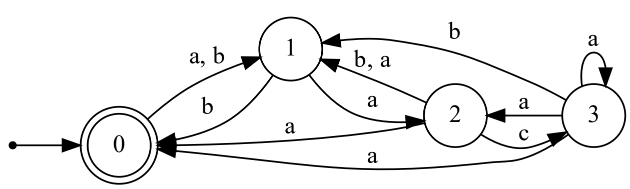
(b) $c^*((a+\varepsilon)c)^* + b^*$

(c) $(aa)^* + (bb)^* + cc$

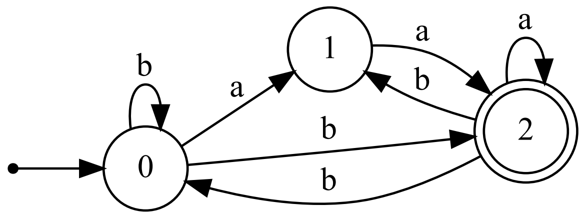
(d) $a(b(cd)^*)^*$

6. For each of the ε-NFAs constructed in the previous question, convert it into an NFA without ε-trasitions.

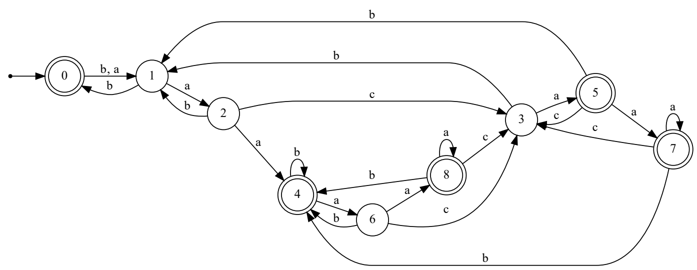
7. Convert the following NFAs to DFAs.



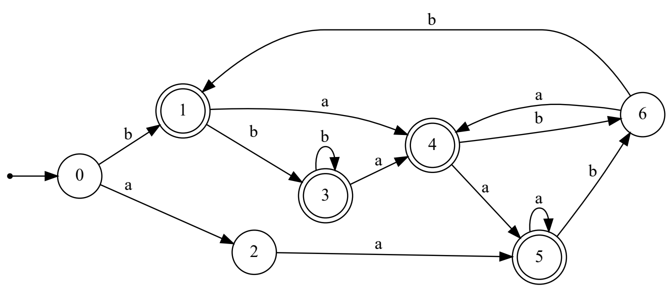
(a)



(b)



(a)



(b)

8. Suppose $M = (Q, \Sigma, \delta, q_0, F)$ is an NFA that accepts a language L with $L \neq \{\varepsilon\}$ and $L \neq \emptyset$, and $M' = (2^Q, \Sigma, \delta', \{q_0\}, F')$ is the DFA constructed from M by the subset construction. Suppose that the state Q is a *reachable* state of M' (by reachable, we mean that there is a word y such that $\delta'(\{q_0\}, y) = Q$).

(a) Is this state a final state? Explain why or why not.

(b) Show that for **some letter** $a \in \Sigma$, the transition on a loops on this state.

(c) Show that the language L is an infinite language (that is, it has infinitely many words in the language).

(a) Since the language L accepted by the original NFA M is non-empty, there is some word that is accepted by the NFA. So that means there must be at least one final state $q_f \in F$ in the original NFA.

Now, as the state Q of M' contains all the states of M, it certainly contains q_f. Thus, by the definition of the subset construction, Q is a final state.

(b) As Q is reachable, that means that there is some word w such that $\delta(\{q_0\}, w = Q$. Let a be the last letter of w (assume that $w \neq \varepsilon$). Let w' be the prefix of w such that $w = w'a$. Then for some set $P \subseteq Q$, we have $\delta(\{q_0\}, w') = P$ and $\delta(P, a) = Q$. That is, P is the subset that we arrive at when reading w', and then when reading a, we go to Q.

But now, as $P \subseteq Q$, we must have that $\delta'(Q, a) = Q$, as we already have $\delta'(P, a) = Q$, so adding additional states to P will not affect where the state is mapped on the letter a. Thus, on the letter a, the state Q loops in M' if $w \neq \varepsilon$.

On the other hand, if $w = \varepsilon$, then $Q = \{q_0\}$, and thus $L(M) = \Sigma^*$ ($L \neq \{\varepsilon\}$ is the only other option, but we have excluded that.). Thus, $\delta(q_0, a) = q_0$ for all letters.

(c) The language L contains the all of the words in wa^*, for the word w that shows that Q is reachable in M. That is because $\delta'(\{q_0\}, w) = Q$ and $\delta'(Q, a) = Q$.

9. Show that the regular languages are closed under difference. That is, if L_1, L_2 are regular languages, show that $L_1 - L_2$ is regular.

We use the same cross product construction as we used in the construction for intersection and union from class. Let $M_1 = (Q_1, \Sigma, delta_1, q_1, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ be DFAs for L_1 and L_2 respectively.

Let $M = (Q_1 \times Q_2, \Sigma, \delta, (q_1, q_2), F)$ be the DFA with

- $\delta((q_i, q_j), a) = (\delta_1(q_i, a), \delta_2(q_j, a))$ and
- $F = F_1 \times (Q - F_2)$

This DFA M accepts $L_1 - L_2$. To see this, note that if $\delta_i(q_i, w) = q_i'$ for $i = 1, 2$, then $\delta((q_1, q_2), w) = (q_1', q_2')$, as δ simulates both the machines in parallel. Thus, $F_1 \times (Q - F_2)$ captures the requirements: the word is accepted by M_1 but rejected by M_2.

10. Show that the regular languages are closed under the operation of suff(L), which is defined as follows:

$$\text{suff}(L) = \{u \in \Sigma^* \ : \ \exists v \in L \text{ such that } u \text{ is a suffix of } v\}.$$

11. Show that the following languages are not regular.

(a) $L_2 = \{w\#y \ : \ w, y \in \{a, b\}^*, |w| < |y|\}$ ($\subseteq \{a, b, \#\}^*$)

(b) $L_3 = \{w\#y \ : \ w, y \in \{a, b\}*, |w|_a = |y|_b\}$.

(c) $L_6 = L^*$ where $L = \{a^n b^n \ : \ n \geq 0\}$

(d) $L_8 = \{w \in \{a, b\}^* \ : \ \exists u \in \{a, b\}^*, w = uu^R\}$.

(a). Let n be the constant descried by the pumping lemma. Let $z = a^n\#a^{n+1}$. Then $z \in L_2$. Now, by the pumping lemma we can write $z = uvw$ with $|uv| \leq n$ and $v \neq \varepsilon$. Thus, $u = a^i$, $v = a^j$ and $w = a^{n-i-j}\#b^{n+1}$ for i, j with $i + j \leq n$ and $j \neq 0$.

Consider $k = 2$. Then $uv^kw = a^{n+j}\#b^{n+1}$. As $j \neq 0$, we must have $j \geq 1$ and so $n + j \geq n + 1$. Thus $uv^kw \notin L_2$ and by the pumping lemma L_2 is not regular.

(b). Let n be the constant descried by the pumping lemma. Let $z = a^n\#b^n$. Then $z \in L_3$. Now, by the pumping lemma we can write $z = uvw$ with $|uv| \leq n$ and $v \neq \varepsilon$. Thus, $u = a^i$, $v = a^j$ and $w = a^{n-i-j}\#b^n$ for i, j with $i + j \leq n$ and $j \neq 0$.

Consider $k = 0$. Then $uv^kw = a^{n-j}\#b^n$. As $j \neq 0$, we must have $j \geq 1$ and so $n - j > n$. Thus $uv^kw \notin L_3$ and by the pumping lemma L_3 is not regular.

(c). Let n be the constant descried by the pumping lemma. Let $z = a^nb^n$. Then $z \in L_6$. Now, by the pumping lemma we can write $z = uvw$ with $|uv| \leq n$ and $v \neq \varepsilon$. Thus, $u = a^i$, $v = a^j$ and $w = a^{n-i-j}b^n$ for i, j with $i + j \leq n$ and $j \neq 0$.

Note that all words in L_6 must have equal numbers of as and bs, as we repeat blocks of different lengths of a^nb^n for some n.

Consider $k = 0$. Then $uv^kw = a^{n-j}\#b^n$. As $j \neq 0$, we must have $j \geq 1$ and so $n - j > n$. Thus $uv^kw \notin L_6$, since it is not of the correct form, and it contains less as than bs. By the pumping lemma L_3 is not regular.

12. Suppose that L_1 is a regular language and L_2 is a non-regular languages. Which of the following statements are guaranteed to be true? Select all the statements that are true. If they are not true, give a counter example.

- $\overline{L_1}$ is a regular language.
- $\overline{L_2}$ is not a regular language.
- L_1^* is a regular language.
- L_2^* is not a regular language.
- $L_1 \cup L_2$ is not a regular language.
- $L_1 \cap L_2$ is not a regular language.
- L_1L_2 is not a regular language.

13. Let $G = (V, \Sigma, P, S)$ be a context-free grammar. Show that the set of all sentential forms for G is itself a CFL.

We show this by constructing a new CFG for the set of all sentential forms for G. Recall that a sentential form for a grammar is anything over $(\Sigma \cup V)^*$ that can be obtained in a partial (incomplete) derviation. So we will construct a new grammar G' that can stop early during any production, and the result will be a word in the grammar. To do this, we make

- Make the alphabet of the new grammar $\Sigma \cup V$, so that anything that contains nonterminals or letters is now a valid word in the new grammar.

- The new nonterminals of G' will have the form A' where A is an original nonterminal from the grammar G. These new nonterminals will do the same derivations as G, but at any time, the nonterminal can be replaced by the old nonterminal A (which is now a letter of the alphabet).

So let $G' = (V', \Sigma', P', S')$ where

(a) $V' = \{A' \,:\, A \in V\}$.

(b) $\Sigma' = \Sigma \cup V$

(c) P' consists of the following rules:

  - A rule of the form $A' \to A$ for all $A \in V$.
  - If $A \to \alpha$ is a rule from P with $\alpha \in (V \cup \Sigma)^*$ then $A' \to \alpha'$ is a rule in P', where $\alpha' \in (\Sigma \cup V')$ is defined by replacing A with A' in α.

(d) The start nonterminal of G' is S', the 'primed' version of S, the start symbol of G.

To see that this works, if $\gamma \in (V \cup \Sigma)^*$ is a sentential form of G, then $S \Rightarrow^* \gamma$. Now to generate this as a word by G', we first derive $S' \Rightarrow^* \gamma'$ in G' by using the primed version of the same productions in the same order as in the derivation of $S \Rightarrow^* \gamma$ in G.

Then for all nonterminals in γ' (remember that they are of the form A' for $A \in V$), then we apply the rule $A' \to A$. This replaces all the primed versions of nonterminals with the "letters" A, and we obtain γ. This completes the derviation of γ in G'.

The remaining questions (14-16) are not covered on the midterm, so no answers will be provided now. The questions will be on the review for the final exam.