Assignment 1 solutions

---

1. The following problems each have a short answer, perhaps just a few sentences and maybe an equation or two. Try to make your answers clear and to the point, and choose the simplest answer whenever possible.

   (a) Let Σ be an alphabet and let $A \subseteq \Sigma^*$ be any infinite language. Prove that there must exist a language $B \subseteq A$ that is not regular.

   (b) Let Σ be an alphabet. Prove that there are countably many finite languages over Σ.

   (c) Let Σ be an alphabet, let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA, and let $p \in Q$ be a state of M. Define a language

   $$A = \{w \in \Sigma^* \,:\, \text{when } M \text{ is run on input } w \text{ it enters the state } p \text{ at least once}\}.$$

   Give a precise, formal description of a DFA that recognizes the language A.

Solution.

(a) Because A is infinite, there are uncountably many subsets $B \subseteq A$. As there are only countably many regular languages over the alphabet Σ, there must exist a nonregular language $B \subseteq A$.

(b) Here are two possible solutions to this problem (but of course you were only expected to provide one answer):

The first solution uses the fact (discussed in class) that there are countably many regular languages over any alphabet Σ. Because every finite language is regular, one has that the set of finite languages is a subset of the set of regular languages—so the fact that there are countably many finite languages over Σ follows.

The second solution argues directly that there are countably many finite languages over an alphabet Σ. One simple way to argue this is to observe that for every $n \in \mathbb{N}$, there are only finitely many languages over Σ that have at most n strings <u>and</u> are such that each string in the language has length at most n. (Note that we are using n to limit two separate aspects of these languages: the number of elements and the length of each element. This sort of trick will often be used later in the course.) If we list all of the languages obtained in this way for increasing values of n, we obtain a countable sequence of finite languages. As every finite language appears somewhere in the sequence, we conclude that there are countably many finite languages.

(c) Define a DFA $K = (Q, \Sigma, \eta, q_0, \{p\})$, where $\eta : Q \times \Sigma \rightarrow Q$ is as follows:

$$\eta(q, \sigma) = \begin{cases} \delta(q, \sigma) & \text{if } q \neq p \\ p & \text{if } q = p \end{cases}$$

for all $q \in Q$ and $\sigma \in \Sigma$. In words, K is similar to M, except that (i) we take p to be its only accept state, and (ii) all transitions out of p are removed and replaced by self-loops pointing back to p. It holds that $\mathrm{L}(K) = A$ as required.

2. Let us say that a string x is obtained from a string w by *deleting symbols* if it is possible to remove zero or more symbols from w so that just the string x remains. For example, the following strings can all be obtained from 0110 by deleting symbols:

$$\varepsilon,\ 0,\ 1,\ 00,\ 01,\ 10,\ 11,\ 010,\ 011,\ 110,\ \text{and } 0110.$$

For the two parts of this question that follow, assume that $\Sigma = \{0,1\}$ and that $A \subseteq \Sigma^*$ is a regular language.

(a) Prove that this language is regular:

$$B = \left\{ x \in \Sigma^* \,:\, \begin{array}{l} \text{there exists a string } w \in A \text{ such that } x \\ \text{is obtained from } w \text{ by deleting symbols} \end{array} \right\}.$$

(b) Prove that this language is regular:

$$C = \left\{ x \in \Sigma^* \,:\, \begin{array}{l} \text{there exists a string } w \in A \text{ such that } w \\ \text{is obtained from } x \text{ by deleting symbols} \end{array} . \right\}$$

Solution.

(a) As the language A is regular, there must exist a DFA $M = (Q, \Sigma, \delta, q_0, F)$ such that $\mathrm{L}(M) = A$. We may obtain an NFA N_B for B by simply adding one ε-transition to M corresponding to each of its transitions. In effect, this allows N_B to follow the same transitions as M, either by reading a symbol from the input or following an ϵ-transition (which effectively simulates the deletion of a hypothetical symbol read by M). More formally, we may take $N_B = (Q, \Sigma, \eta, q_0, F)$, where η is defined as

$$\begin{aligned} \eta(q,\sigma) &= \{\delta(q,\sigma)\} \\ \eta(q,\varepsilon) &= \{\delta(q,\tau) \,:\, \tau \in \Sigma\} \end{aligned}$$

for all $q \in Q$ and $\sigma \in \Sigma$. It holds that $\mathrm{L}(N_B) = B$ and therefore B is regular.

(b) Along similar lines to the solution to part (a), we may prove that C is regular by specifying an NFA N_C such that $\mathrm{L}(N_C) = C$. We will again assume that $M = (Q, \Sigma, \delta, q_0, F)$ is a DFA such that $\mathrm{L}(M) = A$. This time we will design N_C so that it has the same state set as M and simulates M, but in addition it may nondeterministically choose to ignore symbols—which is done by adding self-loop transitions on all of its states. More precisely, let us define $N_C = (Q, \Sigma, \mu, q_0, F)$, where μ is defined as

$$\mu(q,\sigma) = \{\delta(q,\sigma)\} \cup \{q\}$$

for all $q \in Q$ and $\sigma \in \Sigma$. It holds that $\mathrm{L}(N_C) = C$ and therefore C is regular.

3. The following two questions are yes/no questions. In each case, answer "yes" or "no," and give an argument in support of your answer.

   (a) Let $\Sigma = \{0\}$, let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA, and assume that M accepts every string $w \in \Sigma^*$ such that $|w| < |Q|$. Is it necessarily the case that $\mathrm{L}(M) = \Sigma^*$?

   (b) Let $\Sigma = \{0\}$, let $N = (Q, \Sigma, \delta, q_0, F)$ be an NFA, and assume that N accepts every string $w \in \Sigma^*$ such that $|w| < |Q|$. Is it necessarily the case that $\mathrm{L}(N) = \Sigma^*$?

Solution. (a) Yes. We don't really need to make use of the assumption $\Sigma = \{0\}$ to answer this problem, so let us prove it in general (although it is fine if you did use the assumption $\Sigma = \{0\}$ in your answer).

Using the notation we discussed in Lecture 2, we have that $\delta^*(q_0, w)$ denotes the state that M ends up on when run on input w. The assumption of the problem therefore implies that

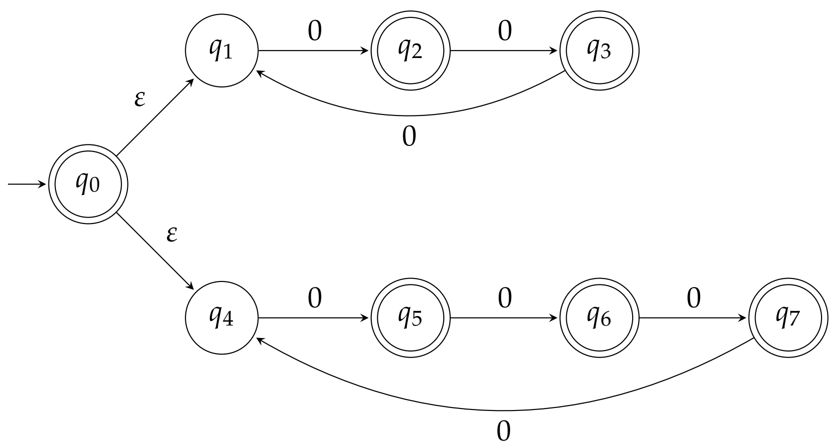
$$\big\{\delta^*(q_0, w) \,:\, w \in \Sigma^*,\ |w| < |Q|\big\} \subseteq F,$$

for otherwise there would be a string of length less than $|Q|$ that is rejected by M. To complete the proof, it suffices to observe that

$$\big\{\delta^*(q_0, w) \,:\, w \in \Sigma^*\big\} = \big\{\delta^*(q_0, w) \,:\, w \in \Sigma^*,\ |w| < |Q|\big\},$$

for then we have that $\delta^*(q_0, w) \in F$ for every $w \in \Sigma^*$. To see that the above set equality holds, consider the fact that the shortest path between two connected vertices in a directed graph must always have length less than the total number of vertices—for otherwise there would be a loop that could be removed to obtain a shorter path. As the state $\delta^*(q_0, w)$ is reachable from q_0, for any choice of $w \in \Sigma^*$, it must be reachable by a path of length less than $|Q|$, and it must therefore hold that $\delta^*(q_0, w) = \delta^*(q_0, x)$ for some choice of $x \in \Sigma^*$ with $|x| < |Q|$.

(b) No. For example, the following NFA has 8 states, it accepts every string $w \in \Sigma^*$ with $|w| < 8$ (and in fact it accepts every $w \in \Sigma^*$ with $|w| \leq 11$), but rejects $w = 000000000000$ (length 12).



One explanation for why the situation is different for NFAs than it is for DFAs is that a state being reachable in an NFA by a string that is accepted does not imply that the state must be an accept state—for there could be a different state that is reachable on the same input that is an accept state.

4. Let $\Sigma = \{0, 1\}$, and define a language

$$\text{Middle} = \{u1v \,:\, u, v \in \Sigma^* \text{ and } |u| = |v|\}.$$

In words, Middle is the language of all binary strings of odd length whose middle symbol is 1. Prove that Middle is not regular.

Solution. Assume toward contradiction that Middle is regular. By the pumping lemma, there exists a pumping length $n \geq 1$ for Middle with the property described by that lemma: for every string $w \in \text{Middle}$ with $|w| \geq n$, it is possible to write $w = xyz$ for strings $x, y, z \in \Sigma^*$ such that

(a) $y \neq \varepsilon$,

(b) $|xy| \leq n$, and

(c) $xy^iz \in \text{Middle}$ for every $i \in \mathbb{N}$.

Let $w = 0^n10^n$. Given that $w \in \text{Middle}$ and $|w| = 2n + 1 \geq n$, there must exist strings $x, y, z \in \Sigma^*$ such that $w = xyz$, and for which the above three items (a), (b), and (c) are satisfied. As $y \neq \varepsilon$ and $|xy| \leq n$, we conclude that $y = 0^k$ for some choice of an integer k with $k \geq 1$. As $xy^iz \in \text{Middle}$ for every $i \geq 0$, we conclude that

$$0^{n+k}10^n = xy^2z \in \text{Middle}.$$

However, the string $0^{n+k}10^n$ is <u>not</u> contained in Middle, as the only 1 in this string is not in the middle position (because $k \geq 1$). We have therefore derived a contradiction, and thus Middle is not regular.

5. Let Σ be an alphabet and let $A \subseteq \Sigma^*$ be a regular language. Prove that the language

$$B = \{w \in \Sigma^* \,:\, www \in A\}$$

is regular.

Solution. Given that A is regular, there must exist a DFA $M = (Q, \Sigma, \delta, q_0, F)$ such that $\mathrm{L}(M) = A$. We will construct an NFA N for which it holds that $\mathrm{L}(N) = B$, which will establish that B is regular, as required.

The main idea behind the construction of N is that N will independently simulate three copies of M; the first copy simulates M when it reads the first instance of w, the second copy simulates M when it reads the second instance of w, and the third copy simulates M when it reads the third instance of w. The main difficulty in doing this, however, is that we don't know what state we should start the second copy of M in, because we don't know where the first copy will finish, and likewise for the third copy. To get around this difficulty we will use the power of nondeterminism to guess what these states should be, which requires that we check the validity of these guesses in addition to the acceptance of the third copy of M.

In detail, let us define N as follows:

$$N = (R, \Sigma, \eta, r_0, G).$$

Here, the state set of N is $R = Q^5 \cup \{r_0\}$, for a state r_0 not contained in Q^5, which we take as our start state. The transition function

$$\eta : R \times \Sigma \to \mathcal{P}(R)$$

is defined as follows:

(i) $\eta(r_0, \varepsilon) = \{(p, q, q_0, p, q) \,:\, p, q \in Q\}$,

(ii) $\eta((p, q, r, s, t), \sigma) = (p, q, \delta(r, \sigma), \delta(s, \sigma), \delta(t, \sigma))$ for all $p, q, r, s, t \in Q$ and $\sigma \in \Sigma$, and

(iii) for all other inputs η outputs $\varnothing$.

Finally, the set of accept states of N is defined as

$$G = \{(p, q, p, q, r) \,:\, p, q \in Q,\ r \in F\}.$$

In general, the way we interpret the state (p, q, r, s, t) of N is as follows:

- We initially guessed that we'll be reading a string w with $\delta^*(q_0, w) = p$ and $\delta^*(p, w) = q$, and we'll have to wait until the end to determine whether or not this was a good guess.
- For whatever string w we have actually read thus far, it holds that $\delta^*(q_0, w) = r$, $\delta^*(p, w) = s$, and $\delta^*(q, w) = t$.

We now see that the condition that N accepts w is equivalent to the existence of states $p, q \in Q$ such that $\delta^*(q_0, w) = p$, $\delta^*(p, w) = q$, and $\delta^*(q, w) \in F$, which in turn is equivalent to the condition that $www \in A$. It therefore holds that $\mathrm{L}(N) = B$.

An alternative solution to this problem (which I did not see until after Lecture 6 in which we discussed the basic technique being used) is as follows. As above, let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA such that $A = \mathrm{L}(M)$. For each pair $(p, q) \in Q \times Q$, define a new DFA

$$M_{p,q} = (Q, \Sigma, \delta, p, \{q\})$$

and define $A_{p,q} = \mathrm{L}(M_{p,q})$. In words, $A_{p,q}$ is the set of all strings that cause M to move from state p to state q. Of course we have that $A_{p,q}$ is regular for every choice of p and q because it is the language recognized by the DFA $M_{p,q}$.

Now, consider the language

$$\bigcup_{\substack{p,q \in Q \\ r \in F}} \bigl(A_{q_0,p} \cap A_{p,q} \cap A_{q,r}\bigr). \tag{1}$$

This is a finite union of finite intersections of regular languages, and therefore it is regular. To complete the solution we will observe that this language is precisely B.

Suppose first that $w \in B$, and let $p = \delta^*(q_0, w)$, $q = \delta^*(p, w)$, and $r = \delta^*(q, w)$. From these definitions it follows that $w \in A_{q_0,p} \cap A_{p,q} \cap A_{q,r}$. Moreover, it must hold that $\delta^*(q_0, www) = r$, and because $w \in B$ implies $www \in A$, and therefore M accepts www, it follows that $r \in F$. We therefore have that w is contained in the set (1).

On the other hand, suppose w is contained in the set (1). It must therefore hold that there exist states $p, q \in Q$ and $r \in F$ such that $p = \delta^*(q_0, w)$, $q = \delta^*(p, w)$, and $r = \delta^*(q, w)$. Consequently it holds that $\delta^*(q_0, www) = r$, which implies that M accepts www, and therefore $w \in B$.

We have proved that

$$B \subseteq \bigcup_{\substack{p,q \in Q \\ r \in F}} \bigl(A_{q_0,p} \cap A_{p,q} \cap A_{q,r}\bigr) \qquad \text{and} \qquad \bigcup_{\substack{p,q \in Q \\ r \in F}} \bigl(A_{q_0,p} \cap A_{p,q} \cap A_{q,r}\bigr) \subseteq B,$$

so the two sets are equal and the solution is complete.