Solutions of Assignment 1

> Theory of Computation, Fall 2022

Keys

Q1

$$(q_1, aab) \vdash_M (q_2, ab) \vdash_M (q_3, b) \vdash_M (q_1, e) \tag{1}$$

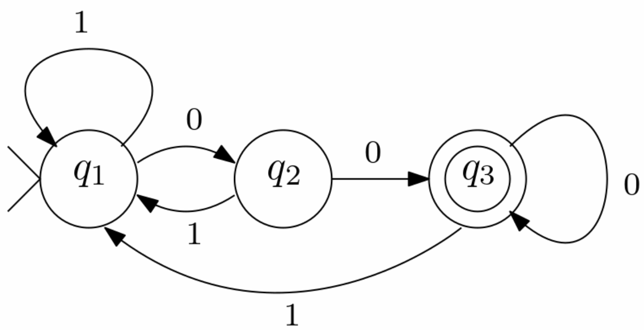
Q2

(a) True. (b) True. (c) False. (d) True. (e) True. (f) True. (g) True.

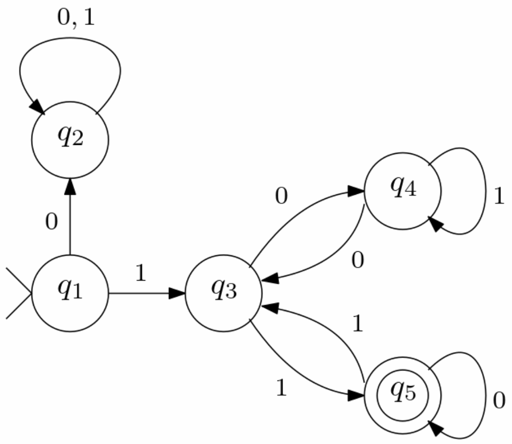
Q3

To prove a language is regular, just prove some FA accepting it exists.

(a)



(b)



Q4

Construct a new state q_0, a dead state, which differs from any other existing states.

$$\begin{gathered} K' = K \cup \{q_0\} \\ s' = s \\ F' = F \\ \delta' = \delta \cup \{((q,a),q_0) | \forall q \in K, \forall a \in (\Sigma' \backslash \Sigma)\} \cup \{(q_0,b),q_0 | \forall b \in \Sigma'\} \end{gathered} \tag{2}$$

Besides, the definition of δ' can be also described as the following.

$$\begin{aligned} &\delta'(q,c) = \delta(q,c), && \forall q \in K, && \forall c \in \Sigma \\ &\delta'(q,a) = q_0, && \forall q \in K, && \forall a \in (\Sigma' \backslash \Sigma) \\ &\delta'(q_0,b) = q_0, && && \forall b \in \Sigma' \end{aligned} \tag{3}$$

Q5

To prove that $A \cap B = \{w | (w \in A) \wedge (w \in B)\}$ is regular, construct a FA accepting $A \cap B$.

Since A and B are regular, there is:

- $\exists M_1 = (K_1, \Sigma, \delta_1, s_1, F_1)$ accepting A;
- $\exists M_2 = (K_2, \Sigma, \delta_2, s_2, F_2)$ accepting B.

Construct a new FA $M_3 = (K_3, \Sigma, \delta_3, s_3, F_3)$ as following.

$$\begin{gathered} K_3 = K_1 \times K_2 \\ s_3 = (s_1, s_2) \\ F_3 = \{(q_1,q_2) \in K_1 \times K_2 | (q_1 \in F_1) \wedge (q_2 \in F_2)\} \\ \delta_3((q_1,q_2),a) = (\delta_1(q_1,a), \delta_2(q_2,a)), \forall q_1 \in K_1, q_2 \in K_2, a \in \Sigma \end{gathered} \tag{4}$$

M_3 accepts $A \cap B$, hence $A \cap B$ is regular.

Besides, the definition of F_3 can be also described as $F_3 = F_1 \times F_2$.

Q6

To prove that $L = \{w_1 \# w_2 | (w_1 \in A) \wedge (w_2 \in B)\}$ is regular, construct a FA accepting L.

Since A and B are regular, there is:

- $\exists M_1 = (K_1, \Sigma, \delta_1, s_1, F_1)$ accepting A;
- $\exists M_2 = (K_2, \Sigma, \delta_2, s_2, F_2)$ accepting B.

It is obvious that $K_1 \cap K_2 = \emptyset$.

Construct a new state q_0, a dead state, which differs from any other existing states.

Construct a new FA $M_3 = (K_3, \Sigma', \delta_3, s_3, F_3)$ as following.

$$\begin{gathered}
\Sigma' = \Sigma \cup \{\#\} \\
K_3 = K_1 \cup K_2 \cup \{q_0\} \\
s_3 = s_1 \\
F_3 = F_2 \\
\delta_3 = \delta_1 \cup \delta_2 \cup \{((q, \#), q_0) | q \in ((K_1 \backslash F_1) \cup K_2)\} \cup \{((q_0, a), q_0) | a \in K_3\} \cup \{((q', \#), s_2) | q' \in F_1\}
\end{gathered} \tag{5}$$

Besides, the definition of δ_3 can be also described as the following.

$$\begin{array}{lll}
\delta_3(q_1, b) = \delta_1(q_1, b), & \forall q_1 \in K_1, & \forall b \in \Sigma \\
\delta_3(q_2, b) = \delta_2(q_2, b), & \forall q_2 \in K_2, & \forall b \in \Sigma \\
\delta_3(q, \#) = q_0, & \forall q \in ((K_1 \backslash F_1) \cup K_2)\} & \\
\delta_3(q_0, a) = q_0, & & \forall a \in \Sigma' \\
\delta_3(q', \#) = s_2, & \forall q' \in F_1 &
\end{array} \tag{6}$$

M_3 accepts L, hence L is regular.

Q7

To prove that $\bar{A} = \{w \in \Sigma^* | w \notin A\}$ is regular, construct a FA accepting $\bar{A}$.

Since A is regular, $\exists M = (K, \Sigma, \delta, s, F)$ accepting A.

Construct a new FA $M' = (K, \Sigma, \delta, s, K \backslash F)$. It accepts $\bar{A}$, hence $\bar{A}$ is regular.

Grading

100 pts. in total

- Q1. 10 pts.
- Q2. 3 pts. each, 21 pts. in total
- Q3. 7 pts. each, 14 pts. in total
- Q4. 15 pts.
- Q5. 15 pts.
- Q6. 15 pts.
- Q7. 10 pts.

Notes

Q1

No comment.

Q2

(a) True. "A string is a **finite** sequence of symbols from some alphabet".

(b) True. "An alphabet is a **finite** set of symbols", "Σ^* is the set of all strings of any length over Σ", which contains a finite number of distinct symbols. So do its subsets, known as languages.

(c) False. No comment.

(d) True. The set of all strings accepted by a DFA is the only language it accepts. **A FA accepts a language only when it accepts all strings in it and rejects all strings not in it.**

(e) True. No comment.

(f) True.

Proof:

- It is obvious that the only empty finite language $\emptyset$ is regular.
- For all non-empty finite languages, it can be denoted as the union of finite number of languages each of that includes only one string.
  - It is easy to construct a FA accepting only one string, that is, accepting a language including only one string, hence every language including only one string is regular.
  - Since the class of regular languages is closed under union, all non-empty finite languages are regular.

Therefore, all finite languages are regular.

(g) True. $L \subseteq \Sigma^*$, $\Sigma^* \subseteq \Sigma'^*$ for $\Sigma \subseteq \Sigma'$, hence $L \subseteq \Sigma'^*$.

Q3

(a)

No comment.

(b)

- q_2 is a dead state;
- q_3 is the state for binary numbers with remainder 1 temporarily;
- q_4 is the state for binary numbers with remainder 2 temporarily;
- q_5 is the state for binary numbers with remainder 0 temporarily.

Q4

Just construct a new dead state, transition to it from any state when encountering newly added characters. Once stepping into the dead state, keep transition to itself whatever character it encounters.

Q5

We can prove that M_3 accepts $A \cap B$ as following.

- **M_3 accepts every string in $A \cap B$.**

  Proof:

  For any string $w \in A \cap B$,

  $$\begin{aligned} &\exists f_1 \in F_1, \quad (s_1, w) \vdash_M^* (f_1, e) \\ &\exists f_2 \in F_2, \quad (s_2, w) \vdash_M^* (f_2, e) \end{aligned} \tag{7}$$

  That is, given that $n = |w|$, and w_i denotes a substring of w that contains all characters other than first i characters, there is:

  $$\begin{aligned} &\exists f_1 \in F_1, \quad \exists q_{1,1}, q_{1,2}, \ldots, q_{1,n-1}, \quad (s_1, w) \vdash_M (q_{1,1}, w_1) \vdash_M (q_{1,2}, w_2) \vdash_M \cdots \vdash_M (q_{1,n-1}, w_{n-1}) \vdash_M (f_1, e) \\ &\exists f_2 \in F_2, \quad \exists q_{2,1}, q_{2,2}, \ldots, q_{2,n-1}, \quad (s_2, w) \vdash_M (q_{2,1}, w_1) \vdash_M (q_{2,2}, w_2) \vdash_M \cdots \vdash_M (q_{2,n-1}, w_{n-1}) \vdash_M (f_2, e) \end{aligned} \tag{8}$$

  With our definition of M_3, there is:

  $$((s_1, s_2), w) \vdash_M ((q_{1,1}, q_{1,2}), w_1) \vdash_M ((q_{1,2}, q_{2,2}), w_2) \vdash_M \cdots \vdash_M ((q_{1,n-1}, q_{2,n-1}), w_{n-1}) \vdash_M ((f_1, f_2), e) \tag{9}$$

  That is:

  $$((s_1, s_2), w) \vdash_M^* ((f_1, f_2), e) \tag{10}$$

  Since $f_1 \in F_1, f_2 \in F_2$, we get$(f_1, f_2) \in F_3$.

  Therefore, M_3 accepts w, hence M_3 accepts every string in $A \cap B$. End of proof.

- **M_3 rejects any string in $\Sigma^* \backslash (A \cap B)$.**

  Proof:

Replace f_1 with any non-final state $q_1 \in (K_1 \backslash F_1)$, and replace f_1 with any non-final state $q_2 \in (K_2 \backslash F_2)$ for the proof above, we get:

$$((s_1, s_2), w) \vdash_M^* ((q_1, q_2), e) \tag{11}$$

Therefore, M_3 rejects w, hence M_3 rejects every string in $A \cap B$. End of proof.

Since M_3 accepts every string in $A \cap B$ and rejects any string in $\Sigma^* \backslash (A \cap B)$, we get M_3 accepts $A \cap B$ as following.

Q6

Transition to the starting state of M_2 when encountering $\#$ in the final states of M_1. Transition to the dead state when encountering $\#$ in any other state.

After the extension of the alphabet, namely the introduction of $\#$ to Σ', $\#$ may appear anywhere in a string waiting to be judged as accepted or rejected by M_3. Therefore, all states in M_3 except final states of M_1, i.e.
$\forall q \in ((K_1 \backslash F_1) \cup K_2)$, states of M_2 included, transition to the dead state once encountering $\#$.

Q7

Invert the final states and non-final states.
