

Homework 1

Name: 方嘉聪 ID: 2200017849

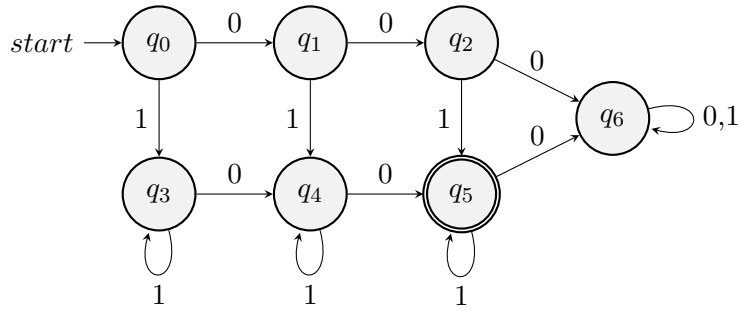
Problem 1.(10 points).

(a) (5 points) Give a DFA recognizing the following language. The alphabet is $\{0, 1\}$.

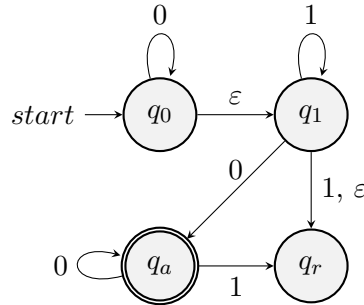
$$\{w \mid w \text{ contains exactly two 0s and at least one 1}\}$$

(b) (5 points) Give an NFA recognizing the language $0^*1^*0^+$ with four states (including a reject state). The alphabet is $\{0, 1\}$.

Answer. 1. The DFA shows below:



2. The NFA shows below (q_r is the rejecting state, q_a is the accepting state):



Problem 2.(20 points). For any string $w = w_1w_2 \cdots w_n$, the reverse of w , written w^R , is the string w in reverse order, $w_n \cdots w_2w_1$. For any language A , let $A^R = \{w^R \mid w \in A\}$. Show that if A is regular, so is A^R .

Answer. 证明思路: 如果 A 是一个正则语言, 那么存在一个 DFA $N = (Q, \Sigma, \delta, q_0, F)$, 使得 $L(N) = A$. 我们的目标是构造一个 NFA $N^R = (Q^R, \Sigma^R, \delta^R, q_0^R, F^R)$, 使得 $L(N^R) = A^R$. 具体的, 我们令 N 中所有的状态转移“反向”, 而后引入一个新的起始状态, 使得其用 ϵ 指向 N 的所有终止状态, 并将 N^R

的终止状态设置为 N 的起始状态即可. 符号化表示为

$$\begin{aligned} Q^R &= Q \cup \{q_0^R\}, \quad \Sigma^R = \Sigma_\varepsilon, \\ q_0^R &\text{ is the new start state,} \\ F^R &= \{q_0\}, \\ \delta^R(q, a) &= \begin{cases} \{p \mid \delta(p, a) = q\} & \text{if } q \in Q, \\ \emptyset & \text{if } q = q_0^R \wedge a \neq \varepsilon, \\ F & \text{if } q = q_0^R \wedge a = \varepsilon. \end{cases} \end{aligned}$$

对于任意的 $w = w_1 w_2 \cdots w_n \in A$, 由于 N 是一个 DFA, 存在一个状态序列 q_0, q_1, \dots, q_n , 使得 $q_0 = q_0, q_n \in F$, 且 $\delta(q_i, w_{i+1}) = q_{i+1}$. 那么由上述定义, 对应的 N^R 状态序列为 $q_0^R, q_n, q_{n-1}, \dots, q_0$ 满足 $q_0^R = q_0^R, q_0 \in F^R$ 且 $q_{i-1} \in \delta^R(q_i, w_i)$, 这即为 N^R 的一个接受状态序列, 从而 $w^R \in L(N^R)$. 因此, A^R 是一个正则语言. \triangleleft

Problem 3.(20 points). Let A be any language. Define $DROP-OUT(A)$ to be the language containing all strings that can be obtained by removing one symbol from a string in A . Thus, $DROP-OUT(A) = \{a \mid \exists y \in \Sigma, \exists x, z \in \Sigma^* : xyz \in A \wedge a = xz\}$. Show that the class of regular languages is closed under the DROP-OUT operation. In other words, if A is a regular language, so is $DROP-OUT(A)$. \blacktriangleleft

Answer. 设 \exists DFA $M = (Q, \Sigma, \delta, q_0, F)$ s.t. $L(M) = A$, 下面我们从 M 出发构造一个 NFA N 使得 $L(N) = DROP-OUT(A)$. 基本思路是使用两个 M (记为 M_1, M_2) 组成 N , 将 M_1 的起始状态作为 N 的起始状态, 将 M_2 的终止状态作为 N 的终止状态, 并对 M_1 中每个有“出边”的状态 q (记为 $\delta(q, a) = p$) 添加一个 ε 转移指向 M_2 的对应状态 p' . 这样得到的 N 与 M 的差别有且仅有一个字符删除的操作 (仅将 M 中的一个状态转移更改为了 ε 转移), 从而 $L(N) = DROP-OUT(A)$.

具体的, 设 $N := (Q \times \{0, 1\}, \Sigma_\varepsilon, \delta_N, (q_0, 0), F \times \{1\})$, 用笛卡尔积表示 N 新的状态, $(q, 0)$ 表示原有 M_1 的状态, $(q, 1)$ 表示原有 M_2 的状态. 我们有如下的形式化表示:

$$\delta_N((q, i), a) = \begin{cases} \{(q', i) \mid \delta(q, a) = q'\}, & \text{if } a \neq \varepsilon \wedge i \in \{0, 1\}, \\ \{(q', 1) \mid \delta(q, a) = q'\}, & \text{if } a = \varepsilon \wedge i = 0. \end{cases}$$

对于任意 $w = w_1 w_2 \cdots w_n \in A$, 存在一个状态序列 q_0, q_1, \dots, q_n , 使得 $q_0 = q_0, q_n \in F$, 且 $\delta(q_i, w_{i+1}) = q_{i+1}$. $\forall i \in \{1, 2, \dots, n\}$, 记 $w_{-k} := w_1 \cdots w_{k-1} \varepsilon w_{k+1} \cdots w_n$, 考虑 N 的状态序列

$$(q_0, 0), (q_1, 0), \dots, (q_{k-1}, 0), (q_k, 1), \dots, (q_n, 1)$$

容易验证满足 $(q_0, 0) = (q_0, 0), (q_n, 1) \in F \times \{1\}$ 且

$$(q_i, 0) \in \delta_N((q_{i-1}, 0), w_i), \forall i \in \{0, 1, \dots, k-1\}$$

$$(q_k, 1) \in \delta_N((q_{k-1}, 0), \varepsilon)$$

$$(q_i, 1) \in \delta_N((q_{i-1}, 1), w_{i+1}), \forall i \in \{k+1, k+2, \dots, n-1\}.$$

因此, $w_{-k} \in L(N)$, 即 $w_{-k} \in DROP-OUT(A)$. 因此, $DROP-OUT(A)$ 是一个正则语言. \triangleleft

Problem 4.(20 points). For a language A , define an equivalence relation between two strings: $x \equiv y$ means $\forall z \in \Sigma^* : xz \in A \Leftrightarrow yz \in A$. This allows the set Σ^* to be divided into different equivalence classes.

For example, if $A = \{w \mid w \text{ contains exactly two 0s and at least one 1}\}$, then $1 \equiv 1111, 101 \equiv 110$, but $1 \not\equiv 011$.

- (5 points) How many different equivalence classes are there for A in the above example?
- (5 points) Prove that for every regular language A , Σ^* can be divided into a finite number of equivalence classes.
- (5 points) Prove that if Σ^* can be divided into a finite number of equivalence classes for a language A , then A is a regular language.
- (5 points) With the above conclusions, prove $A = \{0^n 1^n \mid n \in \mathbb{N}\}$ is not a regular language.

◀

Answer. 记 Σ^* 关于 A 的等价类集合为 $EC(A)$.

- 7 个等价类, $EC(A) = \{\varepsilon, 0, 1, 00, 01, 001, 000\}$.
- 对每个正则语言 A , \exists a DFA $M = (Q, \Sigma, \delta, q_0, F)$, s.t. $L(M) = A$. 设状态数 $|Q| = k$, 我们来证明 $|EC(A)| \leq k$. 用反证法, 如果 $|EC(A)| > k$, 那么由鸽巢原理, $\exists x, y \in EC(A)$, s.t. $x \neq y \wedge \delta(q_0, x) = \delta(q_0, y)$, 这里 $\delta(q_0, x)$ 表示 M 在输入 x 后所处的状态. 那么对 $\forall z \in \Sigma^*$, $\delta(q_0, xz) = \delta(q_0, yz)$, 进而有 $xz \in A \Leftrightarrow yz \in A \implies x \equiv y$, 这与 $x \neq y$ 矛盾. 因此, $|EC(A)| \leq k$. 即 Σ^* 关于 A 的等价类个数是有限的.
- 设 $EC(A) = \{x_1, x_2, \dots, x_k\}$, 我们构造一个 DFA $N = (Q, \Sigma, \delta, q_0, F)$ s.t. $|Q| = k \wedge L(N) = A$. 设 $Q = \{q_1, q_2, \dots, q_k\}$, 令 $F = \{q_i \mid x_i \in A\}$, $q_0 = q_i$ (such that $x_i \equiv \varepsilon$). 令状态转移函数为

$$\delta(q_i, a) = q_j \iff \exists a \in \Sigma \text{ s.t. } x_j \equiv x_i a \quad (1)$$

注意这里 i 可以等于 j . 下面简单证明一下(1)式中 a 是一定存在的, 否则如果 $\forall a \in \Sigma, x_j \not\equiv x_i a$, 那么考虑 $B = EC(A) \cup \{x_i a\}$, 则 B 也是 A 的一个等价类集合, 且 $|B| = k + 1 > |EC(A)|$, 这与 $EC(A)$ 的定义矛盾. 根据上述定义,

$$\forall x \in \Sigma^*, \exists q_i \in Q \text{ s.t. } \delta(q_0, x) = q_i \wedge x \in A \iff x_i \in A$$

这里 $\delta(q_0, x)$ 表示 N 在输入 x 后所处的状态. 因此, $L(N) = A \implies A$ 是一个正则语言.

- 注意到 $\forall k \in \mathbb{N}$, $\{0^k\}$ 是 A 的不同等价类, 注意到 $|\mathbb{N}| = \infty \implies |EC(A)| = \infty$, 由上述结论, A 不是一个正则语言.

◀

Problem 5.(10 points). Prove that the following languages are not regular.

- (5 points) $A = \{0^m 1^n \mid m \text{ and } n \text{ are coprime}\}$.
- (5 points) $B = \{w \mid w \in \{0, 1\}^* \text{ is not a palindrome}\}$. Here a palindrome is a string that reads the same forward and backward.

Answer. (a) 设 p, q 为两个不同的素数, 考虑 0^p 和 0^q , 我们来证明 $0^p \not\equiv 0^q$. 如果 $0^p \equiv 0^q$, 那么 $\forall x \in \Sigma^*, 0^p x \in A \Leftrightarrow 0^q x \in A$, 令 $x = 1^q$, 那么 $0^p 1^q \in A \implies 0^q 1^q \in A$, 这与 A 的定义矛盾. 因此 0^p 和 0^q 是 A 的不同等价类, 注意到素数的个数是无限的, 因而 $|EC(A)| = \infty$, 由问题 4 的结论, A 不是正则语言.

(b) 设 B 的补为 $B^c = \{w \mid w \in \{0, 1\}^* \text{ is a palindrome}\}$. 假设 B 是正则语言, 那么 B^c 也是正则语言, 考虑 $m = 0^{p+1}10^{p+1} \in B^c$. 由于 Pumping Lemma, $\exists x = 0^a, y = 0^b, z = 0^c 10^{p+1}$, s.t. $m = xyz \wedge a + b + c = p + 1 \wedge b > 0$, 进而 $xy^2z = 0^{p+1+b}10^{p+1} \in B^c$, 这显然与 B^c 的定义矛盾. 因此 B^c 不是正则语言, 进而 B 不是正则语言.

这里也给出另一种证明 (利用问题 4 的结论): 注意到对任意的 $k \in \mathbb{Z}^+$, $\{0^k 1\}$ 是 B 的不同等价类, 注意到 $|\mathbb{Z}^+| = \infty \implies |EC(B)| = \infty$, 由问题 4 的结论, B 不是正则语言.

Problem 6.(20 points). Let the rotational closure of language A be $RC(A) = \{yx \mid xy \in A\}$.

- (a) (10 points) Show that for any language A , we have $RC(A) = RC(RC(A))$.
 (b) (10 points) Show that the class of regular languages is closed under rotational closure. In other words, if A is a regular language, so is $RC(A)$.

Answer. (a) (1) 我们先证明 $RC(A) \subseteq RC(RC(A))$. 对 $\forall z \in RC(A) \implies z = \varepsilon \cdot z \in RC(A) \implies z = z \cdot \varepsilon \in RC(RC(A)) \implies RC(A) \subseteq RC(RC(A))$. 同理有 $A \subseteq RC(A)$.

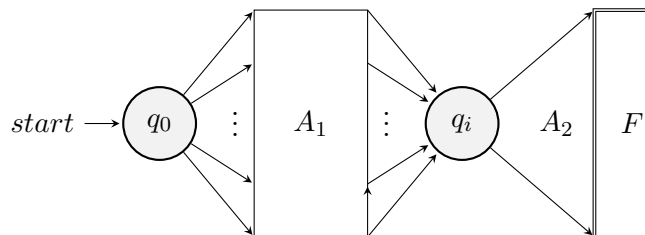
(2) 我们再证明 $RC(RC(A)) \subseteq RC(A)$. 对 $\forall z \in RC(RC(A)), \exists xy \in RC(A)$ s.t. $z = yx$.

由于 $A \subseteq RC(A)$, 若 $xy \in A \implies z = yx \in RC(A)$.

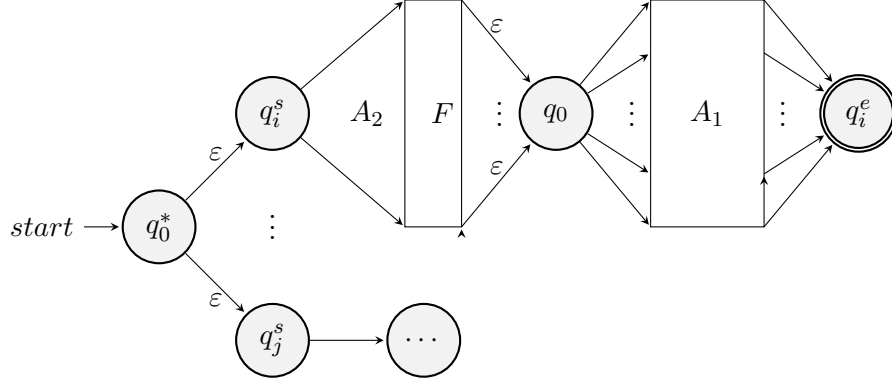
若 $xy \in RC(A) \setminus A \implies \exists x'y' \in A$, s.t. $y'x' = xy$. 如果 $y' = x, x' = y$, 那么 $z = yx \in A \subseteq RC(A)$. 如果 $y' \neq x, x' \neq y$, 不妨设 $y' = x\Delta, y = \Delta x' (\Delta$ 表示 x, y' 相差的子字符串, 另一种情况是类似的), 那么 $x'x\Delta = x'y' \in A \implies \Delta x'x = yx \in RC(A)$. 因此 $RC(RC(A)) \subseteq RC(A)$.

综上, $RC(A) = RC(RC(A))$.

- (b) 设 \exists 一个 DFA $M = (Q, \Sigma, \delta, q_0, F)$, s.t. $L(M) = A$. 我们从 M 出发构造一个 NFA $N = (Q', \Sigma', \delta', q'_0, F')$, s.t. $L(N) = RC(A)$. 基本思路是对每个状态 $q_i \in Q$, 我们将 Q 分成可以到达 q_i 的状态集合 A_1 和从 q_i 可以到达的状态集合 A_2 (如果 A_1, A_2 中有相同的状态则复制一份). 图示如下:



而后我们将 q_i 复制两份 (记为 q_i^s, q_i^e), 将 $q_0 A_1 q_i^e$ 与 $q_i^s A_2$ 位置对换, 再在 F 中的状态与 q_0 之间各增加一个 ε 转移. 对每个 $q_i \in Q$, 我们都可以进行这样的变换 (共有 $|Q|$ 个), 令 $F' = \{q_i^e \mid \forall q_i \in Q\}$, 最后再增加一个新的起始状态 q_0^* , 使得 q_0^* 通过 ε 转移指向每个起始 q_i^s . 图示如下:



下面我们简单证明一下 $L(N) = RC(A)$.

1. $\forall z \in RC(A), \exists xy \in A, s.t. z = yx$. 设 $x = w_1 w_2 \cdots w_i, y = w_{i+1} w_{i+2} \cdots w_k, \delta(q_{i-1}, w_i) = q_i$. 即 $q_0 \xrightarrow{x} q_i \xrightarrow{y} q_a \in F$. 那么对 N , 由定义有如下的状态转移序列使得 $z \in L(N)$:

$$q_0^* \xrightarrow{\varepsilon} q_i^s \xrightarrow{y} q_a \xrightarrow{\varepsilon} q_0 \xrightarrow{x} q_i^e \in F'$$

2. $\forall z \in L(N)$, 存在一个状态序列 $q_0^* \xrightarrow{\varepsilon} q_i^s \xrightarrow{y'} q_a \xrightarrow{\varepsilon} q_0 \xrightarrow{x'} q_i^e \in F'$ s.t. $z = y'x'$. 那么对 M , 考虑对应的状态转移序列 $q_0 \xrightarrow{x'} q_i \xrightarrow{y'} q_a \in F \implies x'y' \in A \implies y'x' \in RC(A)$.

因此, $L(N) = RC(A)$. 由于 N 是一个 NFA, 因此 $RC(A)$ 是一个正则语言.

◁