形式语言与自动机 2022 期末考试

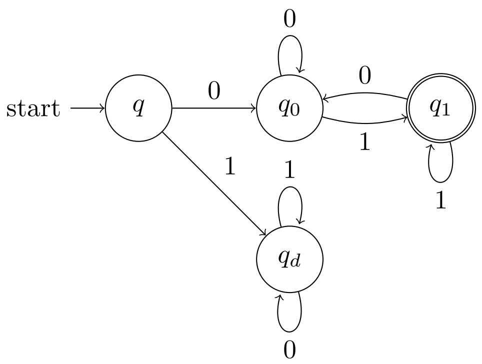
BY: ICS 2022

一、（12 分）

1. （6 分）给出识别以 0 开头 1 结尾的串的 DFA 和正则表达式。
2. （6 分）$L = \{1^n 0^m \mid n + m \text{ 为奇数}\}$，画出识别 L 的 DFA（提示：将 L 拆成两个正则语言的交，并分别画出它们的 DFA）。

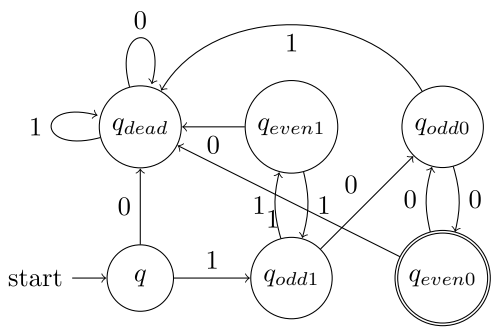
解:

1. **DFA**:



RE: $0(0+1)^*1$.

2. $L = L(E_1) \cup L(E_2)$. $L(E_1) := \{1^{2k+1}0^{2l} : k, l \geq 0\}$, $L(E_2) := \{1^{2l}0^{2k+1} : k, l \geq 0\}$.



DFA for E_1:

DFA for E_2: 类似, 略.

注: 答案作者没有想出如何构造两个正则表达式的**交**, 这里似乎应该是**并**.

注：可以构造为 奇数串 和 1*0* 的交

□

二、（10 分）

$L=\{w\in\{a,b\}^*\mid w$ 的首字母、尾字母、中间字母相同$\}$，给出识别 L 的 CFG 和 PDA。

解: CFG:

$$
\begin{aligned}
S &\to S_a|S_b\\
S_a &\to aAa\\
S_b &\to bBb\\
A &\to a|aAa|aAb|bAa|bAb\\
B &\to b|aBa|bBa|aBb|bBb
\end{aligned}
$$

PDA: 直接使用 CFG 转化为 PDA: $P=(\{q\},\{a,b\},\{a,b,S,S_a,S_b,A,B\},\delta,q_0=q,Z_0=S,\{q\})$, 其中

- $\delta(q,a,a)=\{(q,\epsilon)\}$, $\delta(q,b,b)=\{(q,\epsilon)\}$.
- $\delta(q,\epsilon,S)=\{(q,S_a),(q,S_b)\}$;
- $\delta(q,\epsilon,S_a)=\{(q,aAa)\}$;
- $\delta(q,\epsilon,S_b)=\{(q,bBb)\}$;
- $\delta(q,\delta,A)=\{(q,a),(q,aAa),(q,aAb),(q,bAa),(q,bAb)\}$;
- $\delta(q,\delta,B)=\{(q,b),(q,aBa),(q,aBb),(q,bBa),(q,bBb)\}$.

这样 $N(P)=L$. $\square$

三、（10 分）

给出生成回文串的图灵机，即若输入为 w，停机时纸带上的内容为 ww^R。

解: $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$, 其中:

- $Q = \{q, t, f, acc\} \cup \{(\sigma, 0) : \sigma \in \Sigma\}$;
- $\Gamma = \Sigma \cup \{(\gamma, 1) : \gamma \in \Gamma\}$;
- $q_0 = a$;
- $F = \{acc\}$; ▷ 我们用停机来定义, 这里其实是没有用的
- $\forall x \in \Sigma, \delta(a, x) = (a, (x, 1), R)$;
- $\delta(a, B) = (t, B, L)$; ▷ 读取完字符串
- $\forall x \in \Sigma,\ \delta(t, x) = ((x, 0), (x, 1), R)$; ▷ 记录当前读取的字符
- $\delta((x, 0), B) = (t, x, L)$; ▷ 写下记录的字符
- $\forall x \in \Gamma,\ \delta(t, x) = (t, x, L)$; ▷ 向左边移动找到未读取的字符
- $\forall x \in \{(\sigma, 0) : \sigma \in \Sigma\}, y \in \{(\gamma, 1) : \gamma \in \Gamma\},\ \delta(x, y) = (x, y, R)$; ▷ 一直向右移动
- $\delta(t, B) = (f, B, R)$;
- $\forall (x, 1) \in \{(\gamma, 1) : \gamma \in \Gamma\},\ \delta(f, (x, 1)) = (f, x, R)$; ▷ 还原

首先读取字符串, 然后镜像复制, 同时标记原来的字符串已经复制过的元素. 最后还原原始字符串, 图灵机停机.

四、（10 分）

已知 $\{0^p \mid p\text{为质数}\}$ 不是上下文无关语言，利用语言的封闭性判断并证明 $\{a^n b^m \mid n+m\text{不为质数}\}$ 是否是正则语言。

解: 记 $L=\{0^p \mid p\text{为质数}\}$. 若 $\bar{L}$ 是正则的, 则 L 也是正则的, 这与 L 不是上下文无关语言矛盾, 所以 $\bar{L}$ 不是正则语言.

考虑同态 $h(0)=a$, 则 $M=h(\bar{L})=\{a^n \mid n\text{不为质数}\}$ 不是正则的, 否则 $h^{-1}(M)=\bar{L}$ 是正则语言, 矛盾.

如果 $L'=\{a^n b^m \mid n+m\text{不为质数}\}$ 是正则语言, 我们考虑同态 $g(b)=a$, 则 $g(L')=M$ 是正则的, 矛盾. 所以 $\{a^n b^m \mid n+m\text{不为质数}\}$ 不是正则语言.

五、（10 分）

利用泵引理证明 $\{(0^m1^m)^m \mid m \geqslant 1\}$ 不是上下文无关语言。

解: 设 $L = \{(0^m1^m)^m \mid m \geqslant 1\}$ 是上下文无关语言, 则存在泵引理常数 n, 使得 $\forall z \in L, |z| \geq n, \exists uvwxy = z$, 满足:

- $|vwx| \leq n$.
- $|vx| > 0$.
- $\forall i \geq 0, uv^iwx^iy \in L$.

取 $z = (0^n1^n)^n$. 则存在 $uvwxy = z$, 其中 $|vwx| \leq n$, 所以 vwx 具有形式 0^i1^j 或者 1^i0^j. 我们首先考虑 $vwx = 0^i1^j$, $i + j \leq n$.

- v 或者 x 中有 01, 显然 $uv^2wx^2y \notin L$.
- $v = 0^k$, $x = 0^l$, 则 $\forall r \geq 0, uv^rxy^rz \not\in L$, 因为有一个 (0^m1^m) 的结构被破坏了.

同理可以说明 $vwx = 1^j0^i$ 的情况.

综上, 我们证明了 $\{(0^m1^m)^m \mid m \geqslant 1\}$ 不是上下文无关语言.

六、（12 分）

对于下列语言，在 [A. 正则 B. 上下文无关 C. 递归 D. 递归可枚举 E. 所有语言] 中选出一定包含它的最小集合，并用一句话简要说明：

1. 某个不可判定语言的补集。
2. 某个 NP 语言的补集。
3. 某个上下文无关语言的补集。
4. 某个递归语言与某个递归可枚举语言的交集。
5. $\{a^i b^j c^k d^l \mid i = j = k\}$。
6. $\{a^i b^j c^k d^l \mid i = l \wedge j = k\}$。

解:

1. E. HALT 的补集不是递归可枚举的。
2. C. co-NP 也是可以判定的.
3. C. 上下文无关语言在补集操作下不封闭, 但显然这是可判定的.
4. D, 考虑$\sum*$（显然递归）和任意一个递归可枚举语言的交.
5. C.
6. B. 容易构造 CFG(略).

七、（10 分）

利用规约证明 $REGULAR_{\mathrm{TM}}$，即 $\{M \mid M$为图灵机且$L(M)$为正则语言$\}$ 不可判定。

解: 我们证明 $A_{\mathrm{TM}} \leq_m REGULAR_{\mathrm{TM}}$. 设存在 R, R 判定 $REGULAR_{\mathrm{TM}}$. 考虑图灵机 S, 输入为 $\langle M, w\rangle$:

1. 构造图灵机 T, 其输入为 x:
   - 如果 x 具有形式 0^n1^n, 那么 T 接受 x.
   - 否则, 在 M 上模拟 w. 如果 M 接受 w, 那么 T 接受 x. 否则拒绝 x.
2. 用 R 模拟 $\langle T\rangle$, S 接受 $\langle M, w\rangle$, 当且仅当 R 接受 $\langle T\rangle$.

下面证明, $\langle M, w\rangle \in A_{\mathrm{TM}} \iff \langle M, w\rangle \in L(S)$.

如果 M 接受 w, 则对任意 x, T 接受 x. 此时 $L(T) = \Sigma^*$ 为正则语言, 所以 R 接受 T, S 接受 $\langle M, w\rangle$; 若 M 拒绝 w, 则 $L(T) = \{0^n1^n : n \geq 0\}$ 不是正则语言, R 拒绝 $\langle T\rangle$, S 拒绝 $\langle M, w\rangle$.

综上, 我们证明了 $A_{\mathrm{TM}} \leq_m REGULAR_{\mathrm{TM}}$, 从而 $REGULAR_{\mathrm{TM}}$ 不可判定. $\square$

注: 注意等价符号 $\iff$ 的含义是同真同假, 而不是充分必要.

八、（**14 分**）

下面的动态逻辑系统中，w 为输入变量，x_1, x_2 为系统状态变量，y 为输出变量：

$$x_1[k+1] = \neg(x_1[k] \oplus x_2[k]), \quad x_1[0] = 0$$

$$x_2[k+1] = u[k] \wedge x_1[k], \quad x_2[0] = 0$$

$$y[k] = x_1[k] \vee x_2[k]$$

1.（8 分）画出 Transition System。

2.（6 分）给出下列性质的 CTL 公式，并判断可否满足：

(a) 在所有可能演变中，至少一种演变中出现 $x_1 = 1 \wedge x_2 = 1$。

(b) 任何演变中都会出现 $y = 1$。

解:

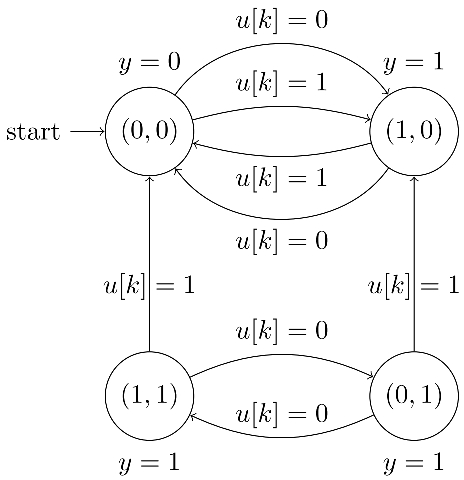


图 1: Transition System

1.

2. (a) $\mathbf{EF}(x_1 = 1 \wedge x_2 = 1)$. 不可满足.

   (b) $\mathbf{AF}(y = 1)$. 可满足.

九、（12 分）

定义字符串的 third 操作：$\text{third}(a_1a_2a_3a_4a_5a_6\cdots) = a_3a_6\cdots$，定义语言中的 third 操作：$\text{third}(L) = \{\text{third}(w) \mid w \in L\}$，证明 third 操作在正则语言下封闭。

解： 考虑构造 $\text{third}(L)$ 的 NFA.

设 $A = (Q, \Sigma, \delta, q_0, F)$ 是 L 的 DFA. 对于字符串 x, 状态 $q_1, q_2 \in Q$, 定义集合 S, $(x, q_1, q_2) \in S$ 当且仅当 A 从状态 q_1 开始, 在读取 x 后来到状态 q_2.

考虑 NFA $N = (Q \cup \{q'\}, \Sigma, \delta', q', F')$, 这里 $q' \notin Q$. 其中 δ' 如下定义:

- $\forall q \in Q, z \in \Sigma$, 令
$$\delta(s, z) = \{q^* \in Q \mid \exists x, y \in \Sigma : (xyz, q, q^*) \in S\}$$
- $\forall z \in \Sigma$, 令
$$\delta(q', z) = \{q^* \in Q \mid \exists x, y \in \Sigma : (xyz, q_0, q^*) \in S\}$$

特别地, $F' = F \cup \{q^* \in Q \mid \exists x \in \Sigma : \delta(q^*, x) \cap F \neq \varnothing\} \cup \{q^* \in Q \mid \exists x, y \in \Sigma, f \in F : (xy, q^*, f) \in S\}$. 这是因为, NFA 的接受状态有三种可能: f 是 DFA 的接受状态, 从 f 可以经过一次转移到达接受状态, 从 f 可以经过两次转移到达接受状态.