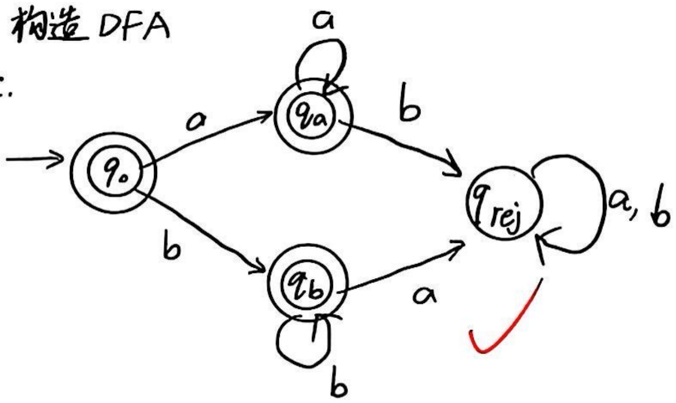
袁晨阳

13/13

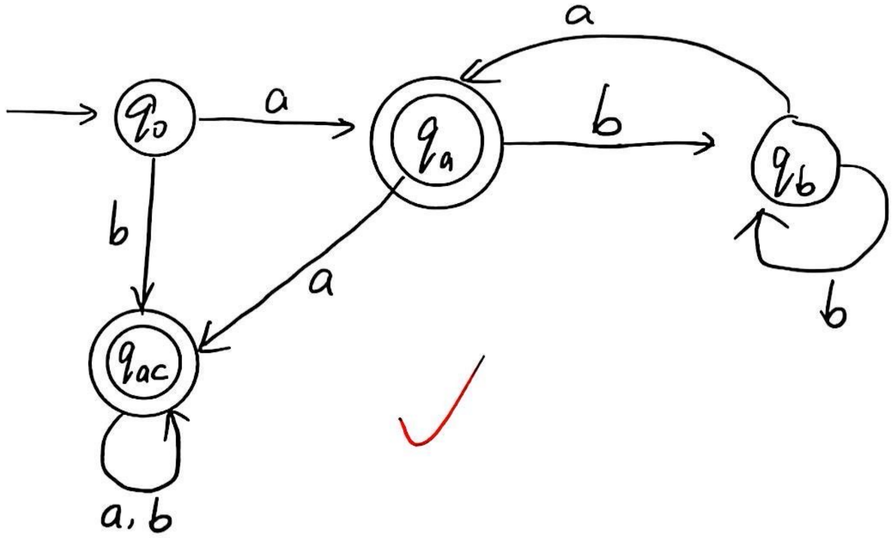
1.1

1.5 构造DFA

c.



e.



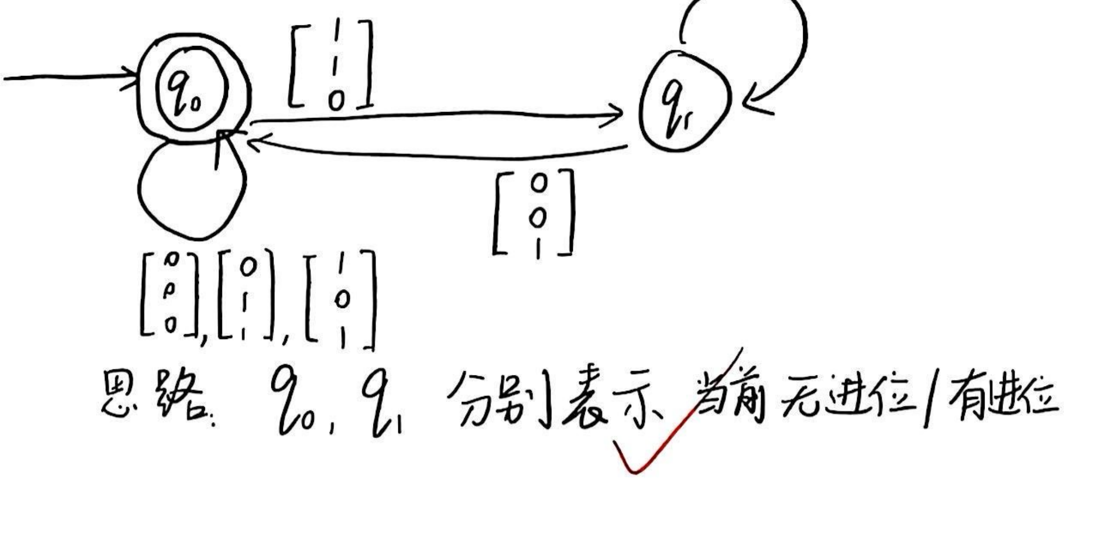
1.14

a. 因为DFA是确定的，每个字符串最后对应到的状态只有一个，所以交换接受状态和非接受状态能改变该字串的接受与否。

所以新DFA识别语言B的补集。

1.37 构造NFA识别 B^R:

$\begin{bmatrix}1\\1\\1\end{bmatrix},\begin{bmatrix}1\\0\\0\end{bmatrix},\begin{bmatrix}0\\1\\0\end{bmatrix}$



思路：q_0，q_1 分别表示当前无进位/有进位

输入串由低位输入至高位，故该NFA识别 B^R，B^R 是正则的

所以 $B=(B^R)^R$ 是正则的.

1.31 设有两 DSA 接受 A、B，分别为 $D_1:(Q_1,\Sigma,\delta_1,q_1,F_1)$ 和 $D_2(Q_2,\Sigma,\delta_2,q_2,F_2)$

下面构造新的NFA接受 A <perfect shuffle> B.

$$N(Q_1\times Q_2\times\{0,1\},\Sigma,\delta,(q_1,q_2,0),(F_1\times F_2,0))$$

其中 $\delta((q_m,q_n,0),a)=(\delta_1(q_m,a),q_n,1)$

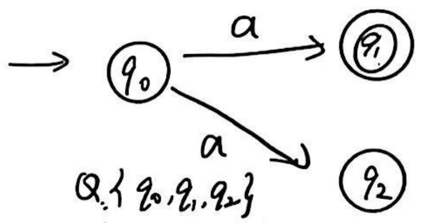
$$\delta((q_m,q_n,1),a)=(q_m,\delta_2(q_n,a),0)$$

0类状态与1类状态分别表示下一个接受到A/B序列.

1-2

1.14. b

i)



$Q:\{q_0,q_1,q_2\}$

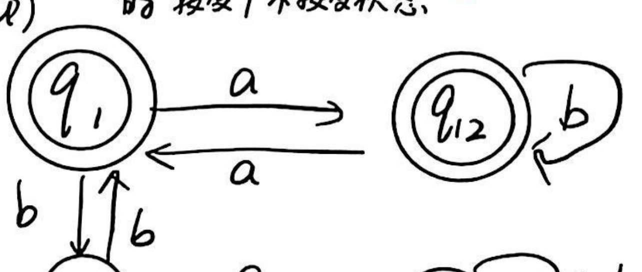
$\Sigma:\{a,b\}$

接受 a. 交换M接受状态与非接受状态后还是接受a.

ii) 封闭. NFA识别的语言也可用DFA识别 此时根据1.14.a 可交换DFA的接受/不接受状态

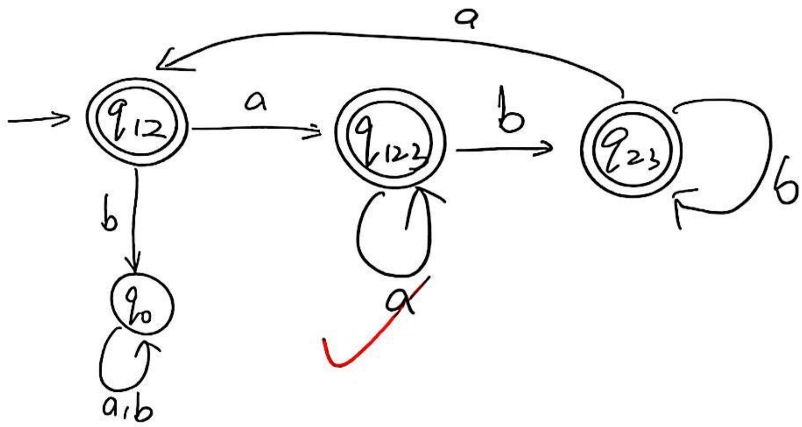
(a)

1.16



扫描全能王
3亿人都在用的扫描App

(b)



1.32 先类似1.31构造出一个NFA N，然后在同一个状态组(q_a, q_b)的两类节点之间添加一对ε连接.

即：$\delta((q_a, q_b, 0), \varepsilon) = (q_a, q_b, 1)$

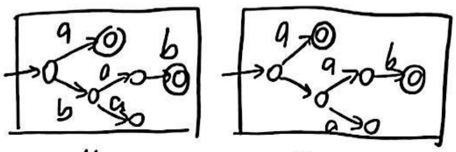
$\delta((q_a, q_b, 1), \varepsilon) = (q_a, q_b, 0)$

相当于测试 $2^{|w|}$ 种将字符串w中每一个字符分配到A/B语言的方案.

1.33

设NFA N 识别A

复制N为N_1、N_2，分别表示未跳过和已跳过. 设N中状态 q_k 分别变为 $q_{k,1}$ 和 $q_{k,2}$



N_1　　N_2

$N_1(Q_1, \Sigma, \delta_1, q_{0,1}, F_1)$

$N_2(Q_2, \Sigma, \delta_2, q_{0,2}, F_2)$

将N_1中所有的接受状态变为普通状态

在N_1中添加一个新的状态 q_{virt} 作为起始状态，通过ε连接到$q_{0,1}$

取所有的互不相同三状态 $q_s, q_{m,1}, q_{e,1} \in Q_1 \cup \{q_{virt}\}$ 使得 $\exists\, a \in \Sigma_\varepsilon, b \in \Sigma$,

$\delta_1(q_s, a) = q_{m,1}$, $\delta_1(q_{m,1}, b) = q_{e,1}$

此时在N_1、N_2中添加连接：$\delta(q_s, a) = q_{e,2}$　　效果：跳过字符b

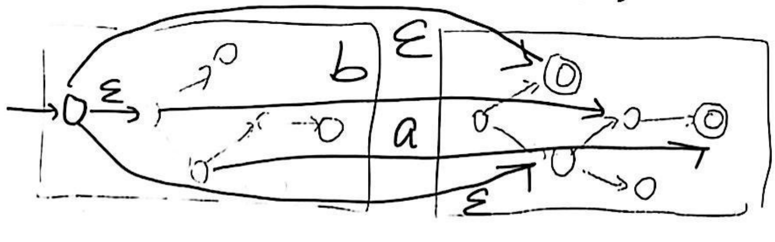


示意图.

扫描全能王
3亿人都在用的扫描App

1.42

$n=1$: 

$n>1$: 设 $1, 2, 4, 2^3, \cdots 2^n$ 模 n 值分别为 $k_0, k_1, k_2, \cdots k_n, \cdots$

根据抽屉原理，必存在 $i<j$ 有 $k_i = k_j$

+1

取最小的 i, j 记为 s, t

所以 2^n 模 n 值为 $k_0, k_1, \cdots k_{s-1}, \underbrace{k_s, k_{s+1}, \cdots k_t}_{\text{循环}}$

对于上述每一个 k_i 设一个状态组 $q_{i,0}, q_{i,1}, \cdots q_{i,n-1}$ 表示当前模 n 余数.

```
def connect(i, j):
    for v in range(n):
        δ(q_{i,v}, 0) = q_{j,v}
        δ(q_{i,v}, 1) = q_{j,(v+k_i) mod p}
```

n

将 $(0,1), (1,2), (2,3), \cdots, (t-1, t), (t, s)$ 传入 connect 函数

构造出转移函数 δ

此时自动机 $D(Q, \{0,1\}, \delta, q_{0,0}, \{q_{k,0} \mid k = 0,1,2,\cdots,t\})$

识别 C_n^R（因为是从低位到高位输入的）

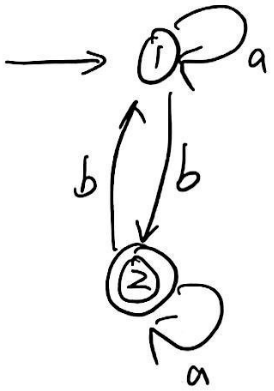
$\therefore C_n^R$ 正则

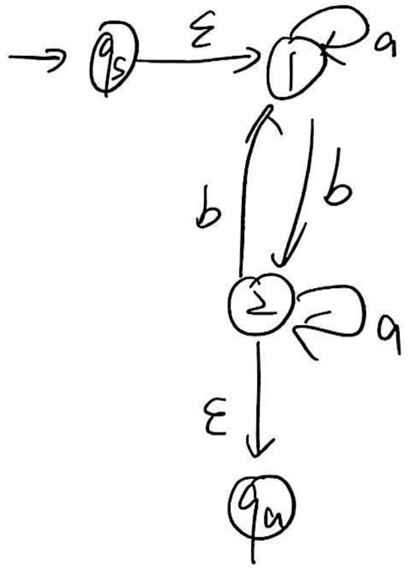
$\therefore C_n$ 正则.

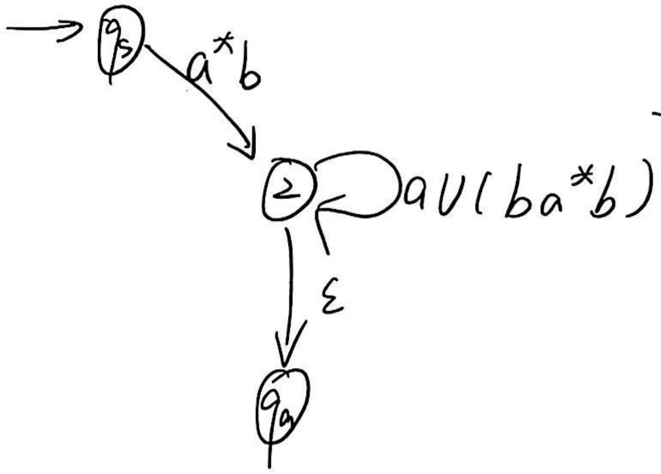
其实直接判断 C_n 更简单.

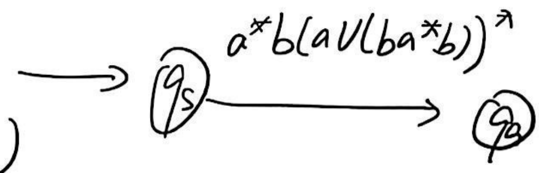
1-3

1.21 a









∴ RE: $a^*b(a\cup(ba^*b))^*$

1.29b • 若是正则语言则 $\exists p, \forall |w| \geq p, \exists x, y, z, w = xyz \; |y| \geq 1, |xy| \leq p$

有 $\forall n \; xy^nz \in A$

题目中w本身就是t这个意思

−0.5 • $\forall p$, 考虑 $w = t^3$, 其中 $t = a^{p-1}b$. 无论如何选择 y, 复制后都有 均成三份后前一份含有的 b 数量比 中间份或后一份少, 所以 xy^nz 无法写成 $(w')^3$ 形式.

不够偏, 比如取 $x = a^{p-2}$, $y = ab$, 复制三份.

• ∴ 不存在这样的泵长度 p. 不是正则语言.

$\underline{a^{p-1}\cancel{a}bab} \mid \underline{ababa^{p-2}} \mid \underline{aba^{p-1}b}$

w

可直接取 $w = a^pb$. 然后从 b 的位置入手

扫描全能王
3亿人都在用的扫描App

7 a. 考虑字符串 ab, abb, abbb, ……

其中任意两个字符串都可用 L 区分，因为对于 ab^i 和 ab^j $(i \neq j)$

有 $ab^ic^j \in F$，$ab^jc^i \notin F$

∴ F 指数无穷

∴ F 不正则

+2.5

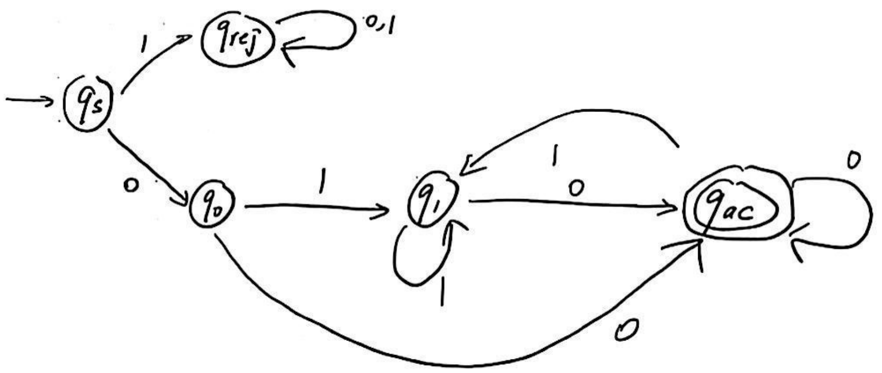
b. $p=2$. $\forall w, |w| \geq p$，令 $x=\varepsilon$, $y=w[0]$, $z=w[1:]$

xy^nz（即重复 w 第一个字母）一定不会破坏 F 的约束条件 ∴ $xy^nz \in F$

泵引理里不仅是重复，还可以拿掉 $(n=0)$. 故若 $w=aabcc$，y 不能取 a.

c. 泵引理只是一个语言是正则的必要条件，满足泵引理不代表语言正则.

1.68 a. 构造 DFA 识别 A：



效果：识别同时以 0 开头、结尾且不是同一个 0 的字符串，这样的串

一定满足 $0^k u 0^k$, $u \in \Sigma^*$ 表达式.

b. $\forall p$，考虑 $w = 0^p10^p$

若将 w 拆分成 xyz，$|y| \geq 1$，$|xy| \leq p$ 必有 y 全为 0. 设 $|y| = t$

∴ $xy^nz = 0^{p+t(n-1)}10^p$

$t \geq 2$ 时不符合 B 的表达式（前导 0 多于后缀 0）

∴ 不存在这样的 p，B 不是正则语言.

扫描全能王
3亿人都在用的扫描App