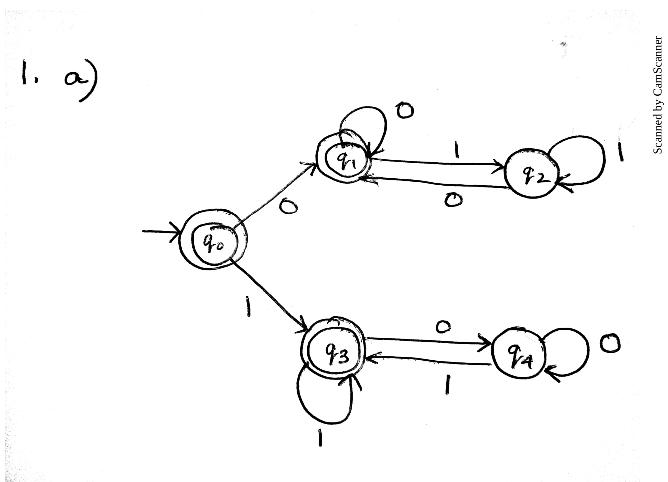
# DFA and NFA Solution

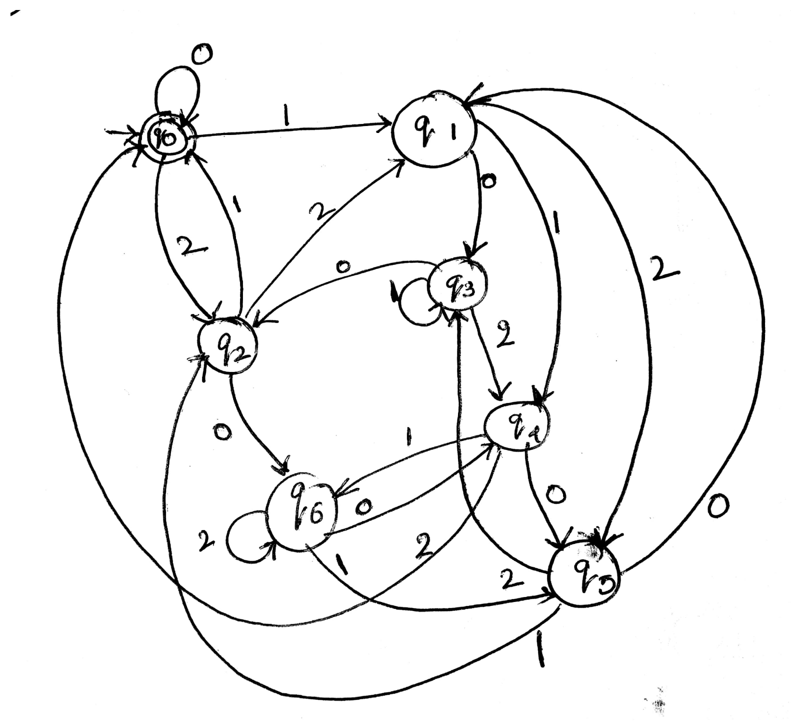
21 Jan 2019

1. Construct DFAs for the following languages.

   (a) $L_1 = \{\omega | \omega \text{ contains an equal number of occurrences of 01 and 10}\}$
   **Solution**

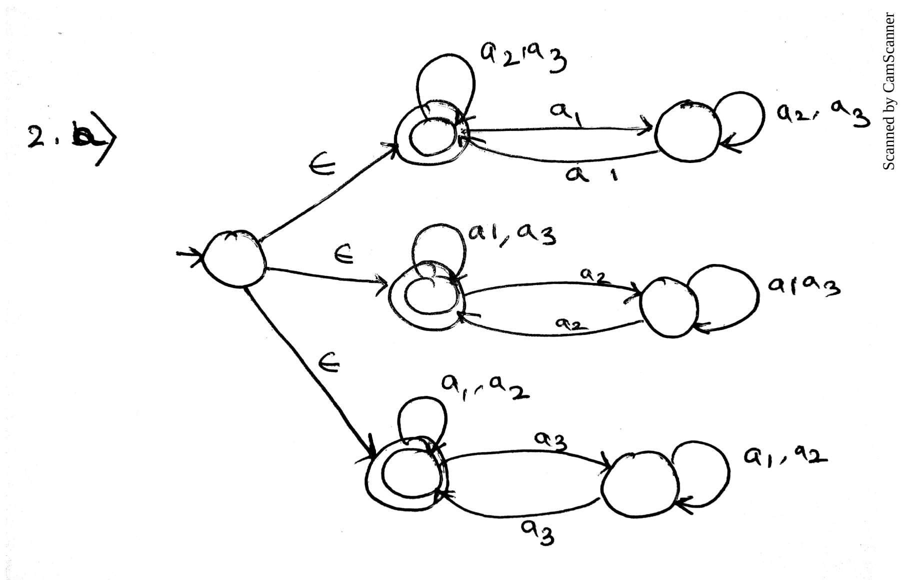
   (b) Ternary Strings (base3), (i.e. $\Sigma = \{0, 1, 2\}$) whose integer equivalent is divisible by 7. (To submit)
   **Solution**



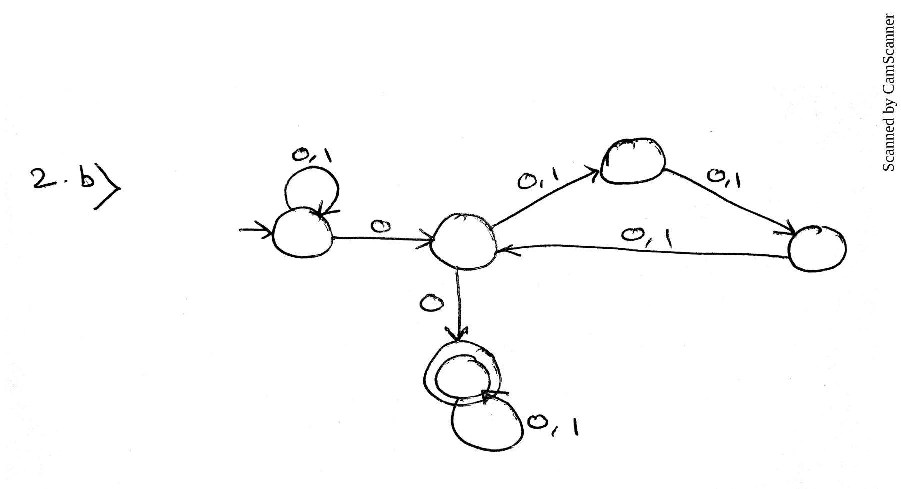


2. Construct NFAs for the following languages.

   (a) $L_2 = \{\omega | \omega$ is a string in which at least one $a_i$ occurs even number of times (not necessarily consecutively), where $1 \leq i \leq 3$ over $\Sigma = \{a_1, a_2, a_3\}\}$.
   **Solution**




(b) $L_3 = \{\omega | \omega$ contains two 0s separated by a substring whose length is a multiple of 3 $\}$, $\Sigma = \{0, 1\}$. (To submit)
**Solution**



3. Prove the following properties.

(a) For languages $A$ and $B$, the shuffle of $A$ and $B$ is the language $L = \{\omega | \omega = a_1 b_1 \cdots a_k b_k\}$, where $a_1 \cdots a_k \in A$ and $b_1 \cdots b_k \in B$, $\forall a_i, b_i \in \Sigma^*$. Prove that the class of regular languages is closed under Shuffle operation. **Solution**

Let $M_A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ be a DFA recognizing A and $M_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$be a DFA recognizing B. The NFA for shuffle of A and B will simulate both $M_A$ and $M_B$ on the input, while non-deterministically choosing which machine to run on a particular input symbol. So the NFA N will be obtained by a modified cross-product construction. Formally, let $N = (Q, \Sigma, \delta, q_0, F)$ where

i. $Q = Q_A Q_B$

ii. $q_0 = (q_A, q_B)$

iii. $F = F_A F_B$

iv. For $a \in \Sigma, \delta$ is given as
$\delta((p_A, p_B), a) = \{(\delta_A(p_A, a), p_B), (p_A, \delta_B(p_B, a))\}$
In all other cases, $\delta$ is $\phi$

The correctness can be established by showing that if N on an input w reaches a state $(p_A, p_B)$ then there is a way to break up $\omega$, so that running $M_A$ on some of the substrings reaches $p_A$ and running $M_B$ on the remaining substrings reaches $p_B$.
The above observation can be proved by induction on the length of $\omega$ and can be used to prove the correctness of the construction.

(b) Let $B$ and $C$ be languages over $\Sigma = \{0, 1\}$. We have defined a language $L = B \leftarrow C$ as $L = \{\omega \in B|$ for some $y \in C$, strings $\omega$ and $y$ contain equal numbers of 1's. $\}$. Show that the class of regular languages is closed under the $\leftarrow$ operation. (To submit)

**Solution**

Let $M_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$ and $M_C = (Q_C, \Sigma, \delta_C, q_C, F_C)$ be DFAs recognizing B and C respectively. Construct NFA M=$(Q, \Sigma, \delta, q_0, F)$ that recognizes B $\leftarrow$ C as follows. To decide whether its input $\omega$ is in B $\leftarrow$ C, the machine M checks that $\omega \in B$, and in parallel, non-deterministically guesses a string y that contains the same number of 1's as contained in $\omega$ and checks that y$\in$ C.

i. $Q = Q_B X Q_C$

ii. For (q,r) $\in$ Q and a$\in \Sigma$ define $\delta((q, r), a)$
$\{(\delta_B(q, 0), r)\}$ if a=0
$\{(\delta_B(q, 1), \delta_C(r, 1))\}$ if a=1
$\{(q, \delta_C(r, 0))\}$ if a=$\epsilon$

iii. $q_0 = (q_B, q_C)$

iv. $F = F_B X F_C$

(c) A homomorphism is a mapping h with domain $\Sigma^*$ for some alphabet $\Sigma$ which preserves concatenation: $h(v \cdot w) = h(v) \cdot h(w)$. Prove that the class of regular languages is closed under Homomorphism operation. (Home)

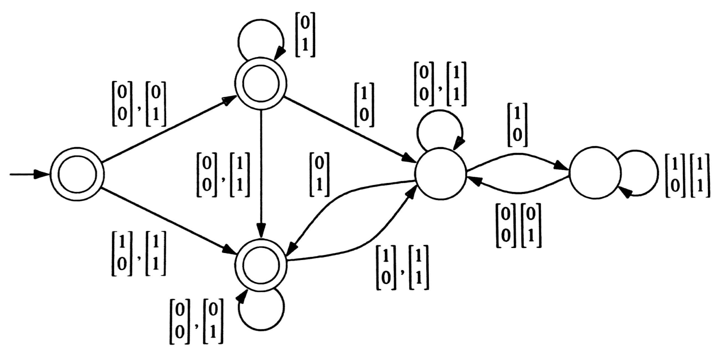
**Solution** Try to solve it yourself.

4. Consider $\Sigma = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}$. A string $\sigma \in \Sigma^*$ can be interpreted as two binary numbers, for example

$$\sigma = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 101100 \\ 010011 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}$$

where $x, y \in \{0, 1\}^*$. Design a DFA which accepts strings in $\Sigma^*$ such that $2x - y \leq 2$. Note that, for such a DFA transitions will be labeled with elements from $\Sigma = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}$. (Home)

Solution:


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