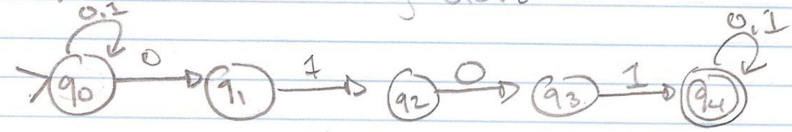
Justin Wong (jujwong)
CMPS 130
Fall 2015

(1) HW #3 10/18/15

1.7 (b, c, d, e, g, h), 1.13, 1.32, 1.36, 1.37, 1.41

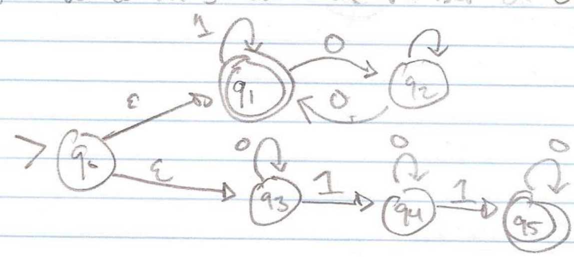
1.7 (b) The language of 1.6c with five states

$\{w \mid w$ contains the substring $0101\}$

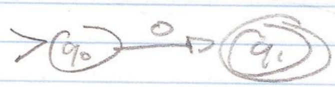


(c) The language of 1.6l with six states

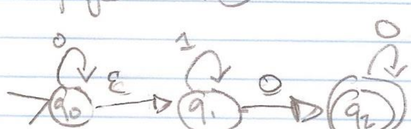
$\{w \mid w$ contains an even number of 0's or contains exactly two 1s$\}$



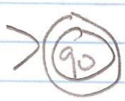
(d) The language $\{0\}$, with two states



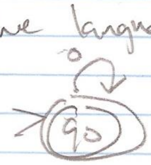
(e) The language $0^*1^*0^+$ with three states



(g) The language $\{\varepsilon\}$ with one state



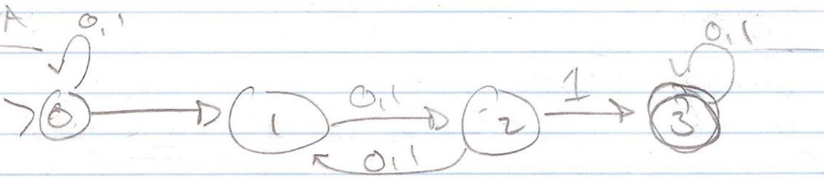
(h) The language 0^* with one state



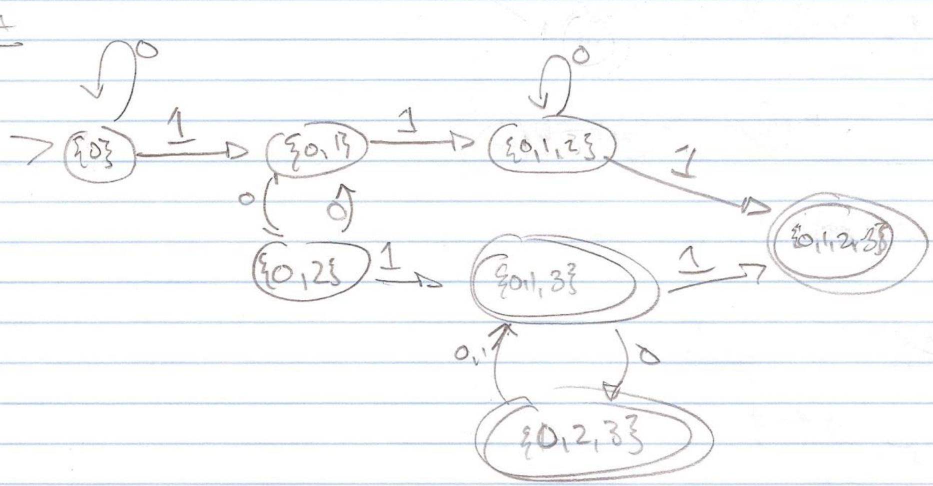
2

1.13 Let F be the language of all strings over {0,1} that do not contain a pair of 1's that are separated by an odd number of symbols.

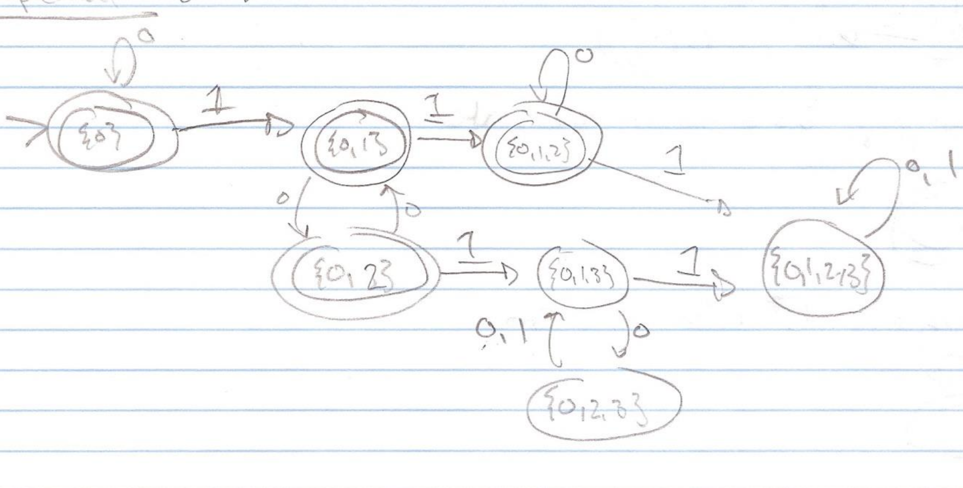
NFA



DFA



Complement of DFA



3

1.32

$\Sigma_3 = \left\{ \begin{bmatrix}0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\1\end{bmatrix}, \ldots, \begin{bmatrix}1\\1\\1\end{bmatrix} \right\}$

Consider each row to be a binary number

let $B = \{ w \in \Sigma_3^* \mid$ the bottom row of w is the sum of the top two $\}$

ex $\begin{bmatrix}0\\0\\1\end{bmatrix}\begin{bmatrix}1\\0\\0\end{bmatrix}\begin{bmatrix}1\\1\\0\end{bmatrix} \begin{matrix}\to 3\\ \to 1\\ \to 4\end{matrix} \in B$ but $\begin{bmatrix}0\\0\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\end{bmatrix} \begin{matrix}\to 1\\ \to 0\\ \to 3\end{matrix} \notin B$

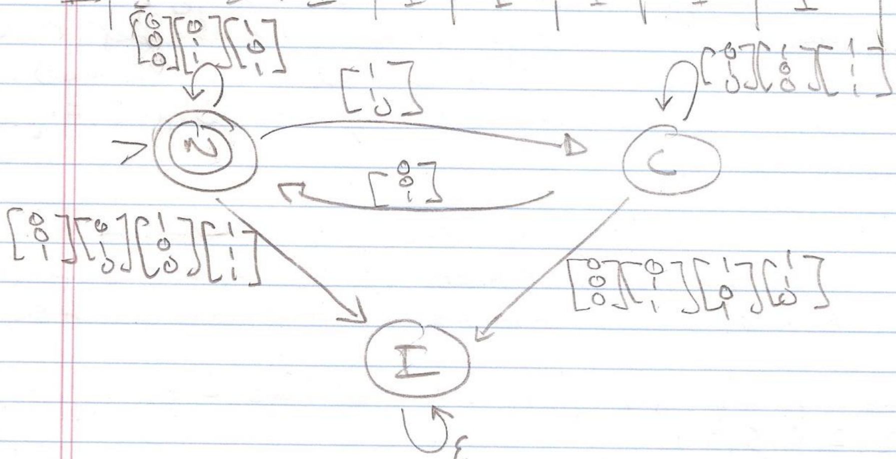
$3+1=4$ $\qquad$ $1+0 \neq 3$

Show B is regular

- ~~If~~ B is regular so is B^R.
- TA - DFA accepts B then B^R is regular

(1) No carry when adding string
(2) Carry Exists when adding up the string goes over the binary amt
(3) Impossible

δ	$\begin{bmatrix}0\\0\\0\end{bmatrix}$	$\begin{bmatrix}0\\0\\1\end{bmatrix}$	$\begin{bmatrix}0\\1\\0\end{bmatrix}$	$\begin{bmatrix}0\\1\\1\end{bmatrix}$	$\begin{bmatrix}1\\0\\0\end{bmatrix}$	$\begin{bmatrix}1\\0\\1\end{bmatrix}$	$\begin{bmatrix}1\\1\\0\end{bmatrix}$	$\begin{bmatrix}1\\1\\1\end{bmatrix}$
N	N	I	I	N	I	N	C	I
C	I	N	C	I	C	I	I	C
I	I	I	I	I	I	I	I	I



$M = (\{N, C, I\}, \Sigma_3, \delta, N, \{N\})$

This machine accepts B^R thus B^R is a regular language

4

1.36 Let $B_n = \{a^k \mid k$ is a multiple of $n\}$ Show that for each $n \geq 1$, the language B_n is regular.

$B_1 = \{a, aa, aaa, aaaa \dots\}$ a^1, a^2, a^3
$B_2 = \{aa, aaaa, aaaaaa, \dots\}$ $(aa)^1, (aa)^2, (aa)^3$
$B_3 = \{aaa, aaaaaa, aaaaaaaaa, \dots\}$ $(aaa)^1, (aaa)^2, (aaa)^3$

M has n states, $\{q_0, q_1, \dots, q_{n-1}\}$. The transition state $\delta(q_i, a) = q_{i+1 \bmod n}$ the starting and accepting state. If the machine stops ~~out~~ q_0 then it accepts Otherwise the machine will increment by 1 and advance to the next state.

1.37 Let $C_n = \{x \mid x$ is a binary number that is a multiple of $n\}$ Show that for each $n \geq 1$. the language is regular.

Looking at a binary number to determine if it is a remainder of n we take the modulo n. If it is 0 then it was a multiple, if not then it was not. So each state in M is represented by the current remainder of the input digit modulo n. The start and accept states are both q_0. Each time a new digit is read, from most significant digit to least, the remainder r is recalculated depending on if it was a 0 or 1. 0 is via ($2i \bmod n$) and 1 is ($2i+1 \bmod n$) then the state is from q_i to q_r. $\delta(q_i, 0) = q_{(2i \bmod n)}$ and $\delta(q_i, 1) = q_{(2i+1 \bmod n)}$.

1.41 For languages A and B let the Perfect Shuffle of A and B be the language $\{w \mid w = a_1 b_1 \dots a_k b_k$ where $a_1 \dots a_k \in A$ and $b_1 \dots b_k \in B$, each $a_i, b_i \in \Sigma\}$ Show that the class of regular languages is closed under perfect shuffle

We can combine the DFAs $M_A = (Q_A, \Sigma, \delta_A, s_A, F_A)$ and $M_B = (Q_B, \Sigma, \delta_B, s_B, F_B)$ to create $M = (Q, \Sigma, \delta, s, F)$
Let $Q = Q_A \times Q_B \times \{S_A, S_B\}$ where S_A and S_B are the next symbol to be read in whole in A or B
The start state in from A side. that is how our perfect shuffle begins
If the string is accepted then both M_A and M_B will be in their final accept states
The transition states should be checking each symbol one from A and then one from B
this is the only way that a string under perfect shuffle is accepted