

Assignment #5

Gustavo Estrela de Matos
CSCE 433: Formal Languages and Automata

Monday 2nd May, 2016

Question 1.

- Eliminate the starting symbol from the right hand side

- Eliminate ϵ rules

$$S \rightarrow AB \mid aB \mid B$$

$$A \rightarrow aab$$

$$B \rightarrow bbA \mid bb$$

- Eliminate unity rules

$$S \rightarrow AB \mid aB \mid bbA \mid bb$$

$$A \rightarrow aab$$

$$B \rightarrow bbA \mid bb$$

- Eliminate rules with more than 2 symbols in the right hand side

$$S \rightarrow AB \mid aB \mid bY \mid bb$$

$$A \rightarrow aZ$$

$$B \rightarrow bY \mid bb$$

$$Y \rightarrow bA$$

$$Z \rightarrow ab$$

- Create unity rules for terminals

$$S \rightarrow AB \mid XB \mid VY \mid VV$$

$$A \rightarrow XZ$$

$$B \rightarrow VY \mid VV$$

$$Y \rightarrow VA$$

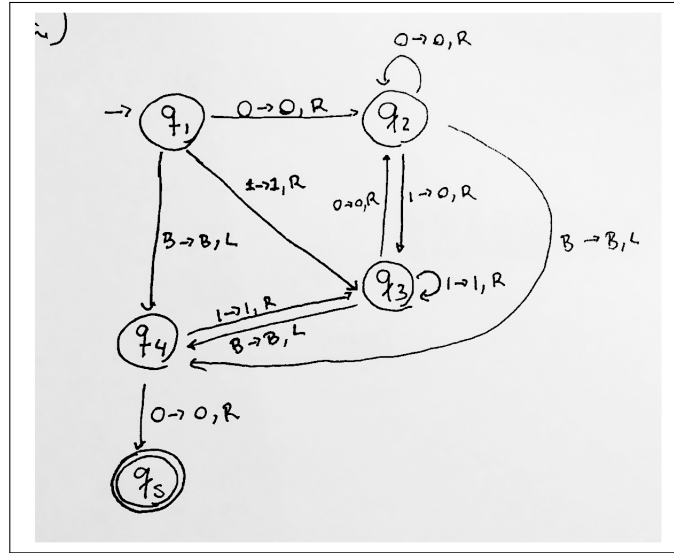
$$Z \rightarrow VX$$

$$V \rightarrow b$$

$$X \rightarrow a$$

Question 2.

(a)



(b)

$q_1 00110 \vdash 0q_2 0110 \vdash 00q_2 110 \vdash 000q_3 10 \vdash 0001q_3 0 \vdash 00010q_2 \vdash 0001q_4 0 \vdash 00010q_5$

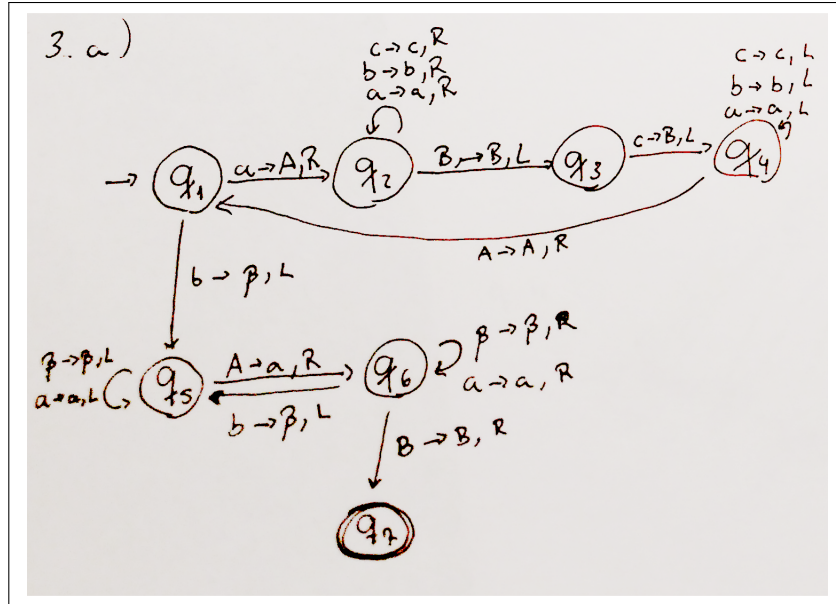
(c)

$q_1 111010 \vdash 1q_3 11010 \vdash 11q_3 1010 \vdash 111q_3 010 \vdash 1110q_2 10 \vdash 11100q_3 0 \vdash 111000q_2 \vdash 11100q_4 0 \vdash 111000q_5$

Question 3.

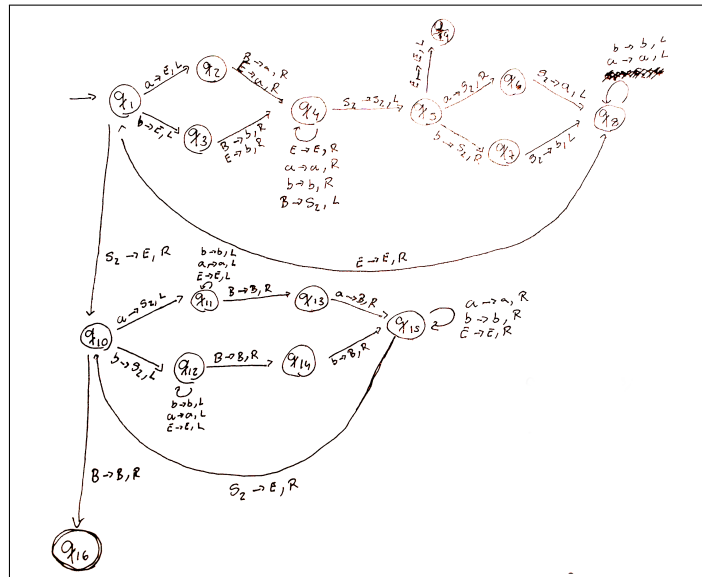
(a)

The idea implemented in this TM is to match a's from left to right (replacing with A's) with the c's from right to left, and when this is done it should match b's from left to right with the A's from right to left.



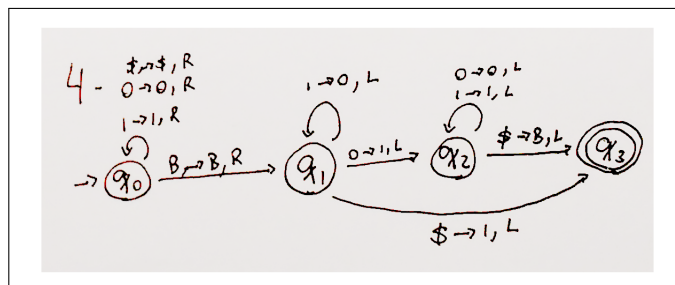
(b)

We can split this machine in two other problems. First we divide the string in the middle, then we match every character of the two produced strings. To divide the string in two we are going to shift left the beginning of the string and the shift right the end of the string until the symbols we use for keeping track of the shifting meet in the middle of the first string. The second part is basically going to the beginning of the two strings and comparing the symbols.



Question 4.

This machine does the algorithm we normally do by hand, from right to left, bit by bit. The key idea is that we are summing N with 0 and starting the carry with 1. The carry is 1 when we are in the state q_1 and it is 0 when we are in the state q_2 .



Question 5.

As we saw in class, the symbol w_i is such that $1w_i = i_2$. Therefore, since $37 = (100101_2)$ we have that $1w_{37} = 100101$ and $w_{37} = 00101$.

Question 6.

- Transitions code:

$\delta(q_1, 0) = (q_3, 0, R)$	$C_1 = 01010001010$
$\delta(q_1, 1) = (q_5, 1, R)$	$C_2 = 010010000010010$
$\delta(q_1, B) = (q_5, B, L)$	$C_3 = 010001000001000100$
$\delta(q_3, 0) = (q_3, 0, R)$	$C_4 = 0001010001010$
$\delta(q_3, 1) = (q_4, 0, R)$	$C_5 = 000100100001010$
$\delta(q_3, B) = (q_5, B, L)$	$C_6 = 00010001000001000100$
$\delta(q_4, 0) = (q_3, 0, R)$	$C_7 = 00001010001010$
$\delta(q_4, 1) = (q_4, 1, R)$	$C_8 = 00001001000010010$
$\delta(q_4, B) = (q_5, B, L)$	$C_9 = 000010001000001000100$
$\delta(q_5, 0) = (q_2, 0, R)$	$C_{10} = 00000101001010$
$\delta(q_5, 1) = (q_4, 1, R)$	$C_{11} = 000001001000010010$

- The machine code:

$C_11C_211C_311C_411C_511C_611C_711C_811C_911C_{10}11C_{11}$

Question 7.

(a)

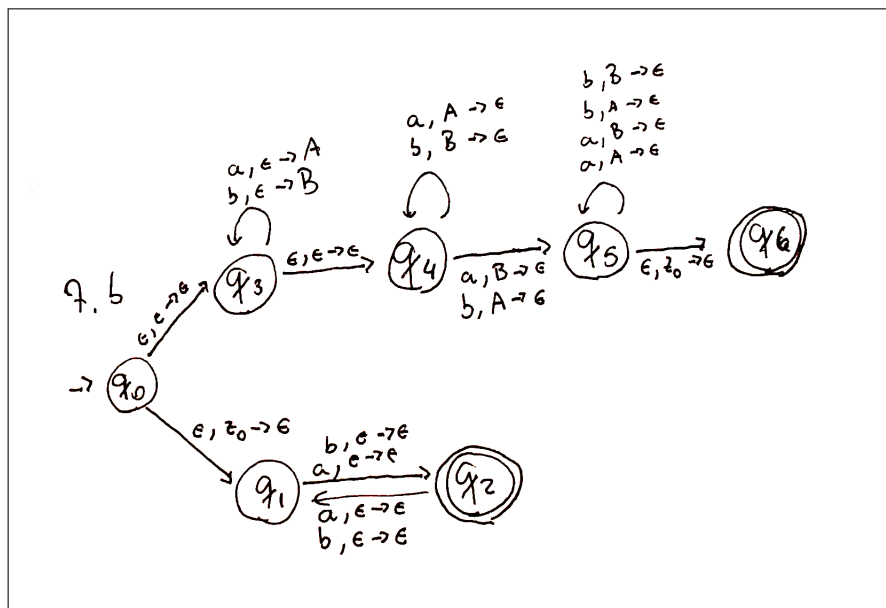
L is CFL. First, we prove that it's not regular using the Pumping Lemma. Assume that L is regular with pumping constant p and consider the string $s = a^p b^{2p} a^p \in L$. According to the Pumping Lemma, we can write $s = xyz$ where $|xy| \leq p$ and $|y| \geq 1$, and also $s' = xy^0z$ should be in L . Since $|xy| \leq p$ we have that $y = a^i$, therefore $s' = a^{p-i} b^{2p} a^p$ and this string should not be in L because $s' \neq reverse(s')$, which contradicts the pumping lemma. Since we have a contradiction, we conclude that our assumption is false and L is not regular.

To prove that this language is a Context Free Language we can construct a grammar for it: $S \rightarrow \epsilon \mid aaSaa \mid baSab \mid abSba \mid bbSbb$.

(b)

\bar{L} is CFL. First we prove that it's not regular using closure properties. Assume that \bar{L} is regular, then L should also be regular. Note that $L = \{w \mid w = reverse(w) \text{ and } |w| \text{ is divisible by } 2\}$, therefore $L \cap ((a+b)(a+b)(a+b)(a+b))^*$ is regular and is the same as $\{w \mid w = reverse(w) \text{ and } |w| \text{ is divisible by } 4\}$, the language that we proved in a) that is not regular, a contradiction. Therefore the assumption that \bar{L} is regular is false.

We show that \bar{L} is regular by building a PDA. This PDA accepts either strings with odd size or strings with even size but not in the format ww^R



(c)

L is regular. We prove this by giving a regular expression for this language: $(b^*ab^*ab^*ab^*ab^*)^*ab^*$