CS 3331a - Assignment 2 - Solutions

Paul Vrbik

October 19, 2009

Question 1- 18 marks

Let the language $L = \{a^i b^i | i \geq 0\}$.

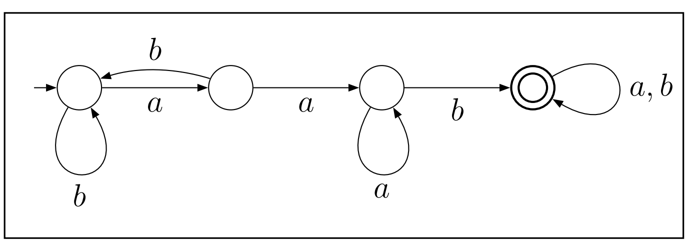
(a) $L_1 = L^R$ is the language consisting of the "reversals" of the words in L (or L's transposes). That is,
$$L_1 = \{b^i a^i | i > 1\}.$$

(b) $L_2 = L^2$ is the language consisting of words that are the concatenation of two words from L. That is,
$$L_2 = \{a^i b^i a^j b^j | i, j \geq 1\}.$$

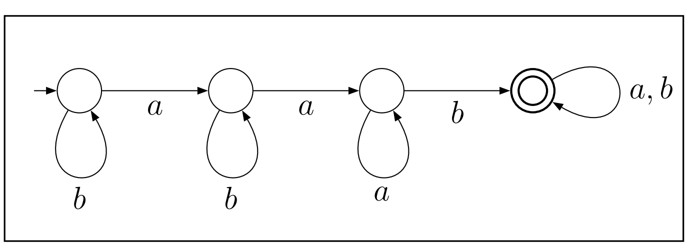
(c) $L_3 = L^*$ are those words that you can construct by concatenating as many (including zero) words from L. That is
$$L_3 = \{a^{i_1} b^{i_1} a^{i_2} b^{i_2} \cdots a^{i_k} b^{i_k} | i_1, \ldots, i_k \geq 1, k \geq 0\}$$

Question 2 - 18 marks

(a) The set of all words that have aab as a subword.

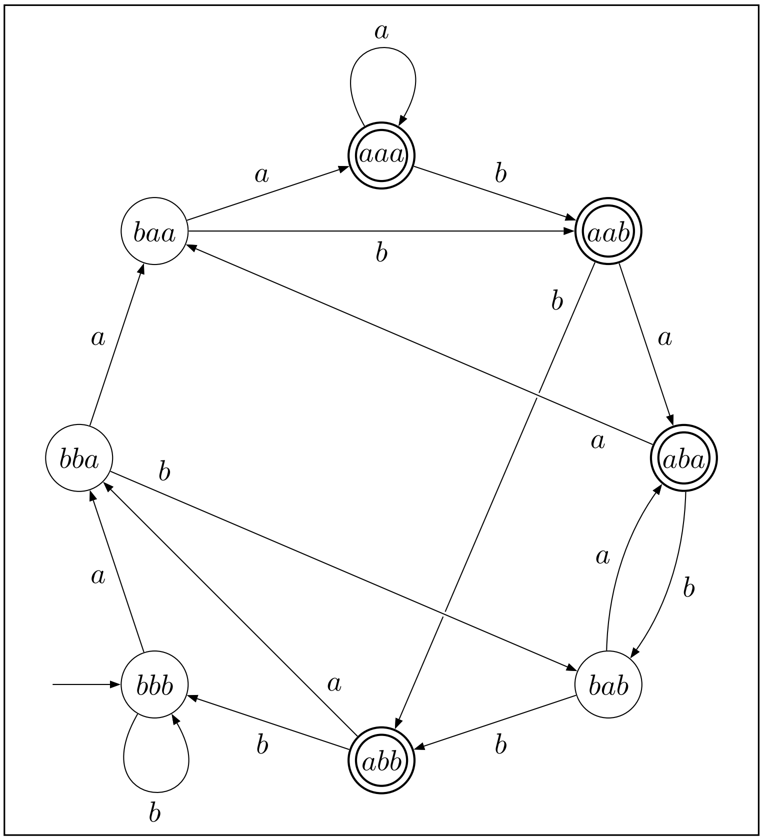


(b) The set of words that have aab as a scattered subword (also called a subsequence).



(c) The set of all words such that the third symbol from the right-end is an a.

There is a strategy which may make this question easier: create states for all possible length three substrings (there are eight of them) and draw rules for transitioning between them. The accepting states are those substrings that begin with 'a'. (I've labelled the states below to reflect this.) You may interpret the start state as being 'bbb' because you will at need at least three moves to get to an accepting state from there.



Question 3 - 15 marks

Nonnegative integers divisible by eight.

Explanation (not required for full credit):
This DFA describes all bit sequences that end in at least three zeros ($(01)^* \circ 000$). Converting this binary number into an integer would be done like this:

$$0 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + s_3 \times 2^3 + s_4 \times 2^4 + \cdots$$

where $s_i \in \{0, 1\}$. This number can be re-written as:

$$2^3 \left(s_3 \times 2^0 + s_4 \times 2^1 + \cdots\right)$$

which are precisely those **nonnegative integers divisible by eight** ($8\mathbb{Z}^+$).

Question 4 - 20 marks

The language $L = \{a^i b^j a^k | i, j, k \geq 0 \text{ and } j < k\}$ is not accepted by any DFA.

Proof. Towards a contradiction assume there is a DFA

$$M = (Q, \{a, b\}, \delta, s, F)$$

such that $L = L(M)$. Let $n = \#Q$ (the number of states). Consider the accepting configuration for $b^n a^{n+1}$,

$$\underbrace{s_0 b^n a^{n+1} \vdash s_1 b^{n-1} a^{n+1} \vdash \cdots \vdash s_n a^{n+1}}_{n+1 \text{ states required}} \vdash \cdots s_{2n+1}$$

where $s_0 = s$ and $s_n \in F$. Note that $n + 1$ states are required to read b^n (as indicated). However, by our assumption there is only n states; therefore by Pigeonhole principle two of these states must be the same. That is, there is $s_i = s_j$ such that $0 \leq i < j \leq n$.

Now consider the accepting configuration,

$$\begin{aligned} b^{n+(j-i)} a^{n+1} &\vdash \cdots \vdash s_j b^{n-i} a^{n+1} \\ &\vdash s_i b^{n-i} a^{n+1} \vdash \cdots \vdash s_{2n+1} \end{aligned}$$

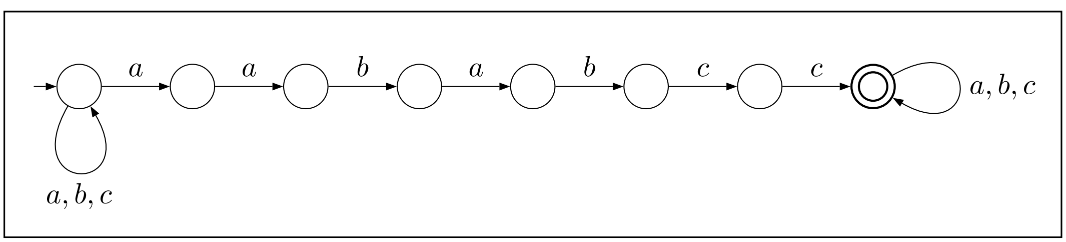
which shows that $b^{n+(j-i)} a^{n+1} \in L$ as well. Since $j - i > 0$ this is a contraction.

□

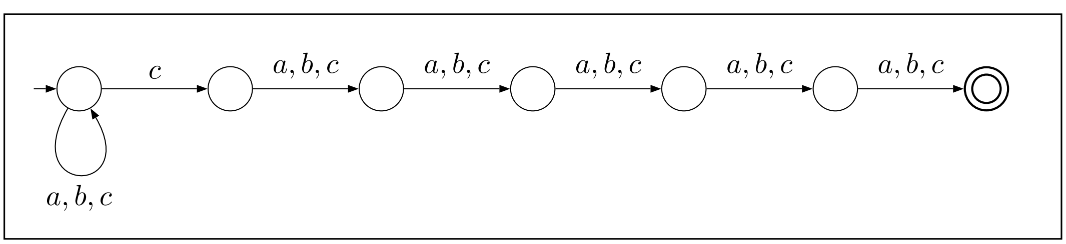
Note: a similar proof was given on page 53 of the lecture notes.

Question 5 - 16 marks

(a) The set of all words that have the subword *aababcc*.



(b) The set of all words such that the sixth symbol from the right-end is c.

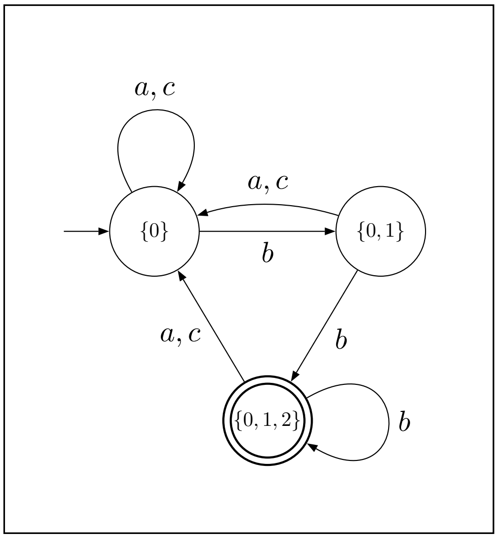


Note: consider how hard it would be to create a DFA to do this (as in Question 2 c); such a machine would require $3^6 = 729$ states for each length six substring of $\{a, b, c\}^*$. Yet, a NFA is no more powerful than a DFA (theoretically) as every DFA can be converted to a NFA (next question). Isn't that amazing!?

Question 6 - 13 marks

We use the iterative subset construction (lecture notes page 65), sets in bold indicate first appearance. *This table is required for full credit*

states	a	b	c
$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$\{0\}$	$\{0\}$	$\{\mathbf{0}, \mathbf{1}\}$	$\{0\}$
$\{0,1\}$	$\{0\}$	$\{\mathbf{0}, \mathbf{1}, \mathbf{2}\}$	$\{0\}$
$\{0,1,2\}$	$\{0\}$	$\{0,1,2\}$	$\{0\}$



Typeset by: Paul Vrbik Compiled on October 23, 2009