Assignment 4 - Regexes

CS 234

Daniel Lee

1 Regexes on Paper

List the strings of length at most 4 for each of the following languages in shortlex order.

5.2 $0(0+1)^*0$

00, 000, 010, 0000, 0010, 0100, 0110

5.3 $(0+1)^*0(0+1)^*1(0+1)^*$

01, 001, 010, 011, 101, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1001, 1010, 1011, 1101

5.5 $(1+00)^*$

λ, 1, 00, 11, 001, 100, 111, 0000, 0011, 1001, 1100, 1111

Give regular expressions for the following languages:

5.6 Every string over $\{0,1\}$

$(0+1)^*$

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

5.10 Contains the substring aab

$(a+b)^*aab(a+b)^*$

5.11 Contains the characters a, a, b in that order but not necessarily next to one another

$(a+b)^*a(a+b)^*a(a+b)^*b(a+b)^*$

5.14 Strings of length divisible by 3

$((a+b)^3)^*$

5.17 Contains at most two bs

$a^* + a^*ba^* + a^*ba^*ba^*$

5.18 Contains at least one a and one b (in any order)

$(a+b)^*a(a+b)^*b(a+b)^* + (a+b)^*b(a+b)^*a(a+b)^*$

5.20 Does not contain the substring aa

$(b^* + (ab)^*)^*(\lambda + a)$

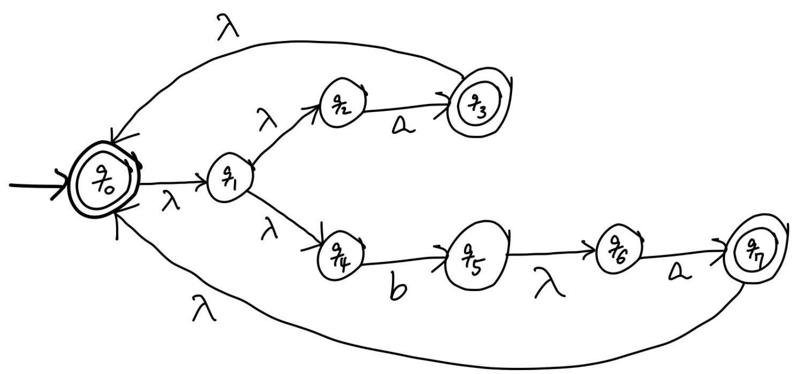
$$\Sigma = \{0-9, -\}$$

5.24 Valid integer (no leading 0s, but could be negative)

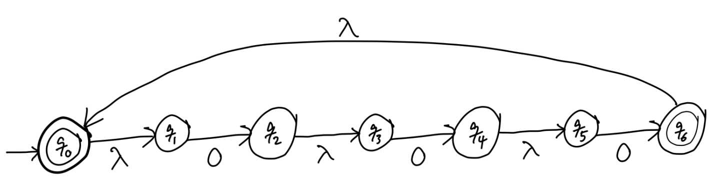
$(0+(1+2+3+4+5+6+7+8+9)(0+1+2+3+4+5+6+7+8+9)^*)+$
$(-(0+(1+2+3+4+5+6+7+8+9)(0+1+2+3+4+5+6+7+8+9)^*))$

Give λ-NFAs using the algorithm in this chapter for the regular expressions below:

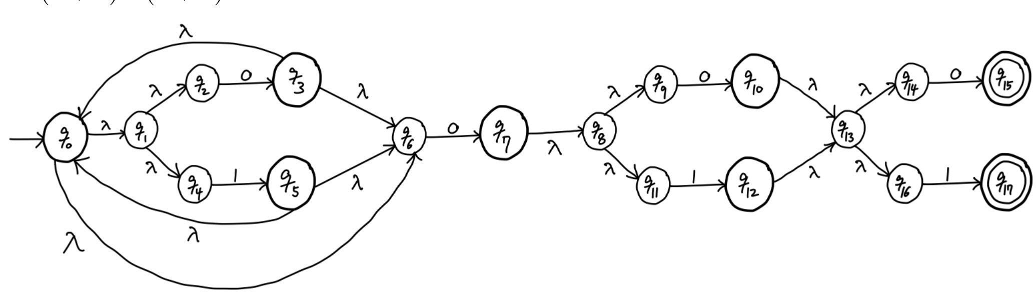
6.2 $(a+ba)^*$



6.6 $(000)^*$

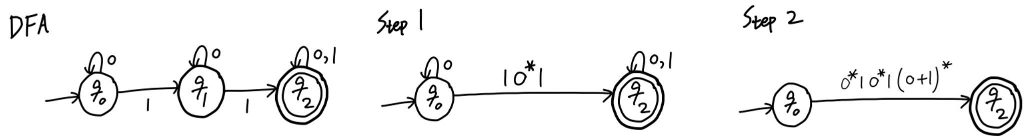


6.9 $(0+1)^*0(0+1)^2$



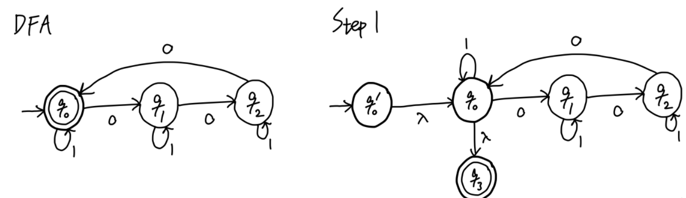
Construct DFAs for the following languages and use the state elimination algorithm in this chapter to create a regular expression for each. Show the intermediate steps as you eliminate each state along the way.

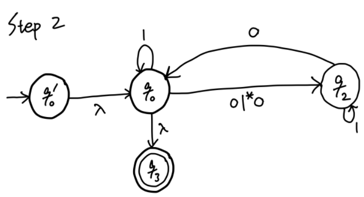
6.13 $\{w \in \{0,1\}^* : w$ has at least two 1 s$\}$

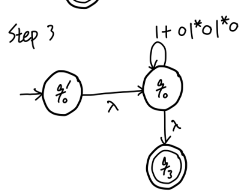


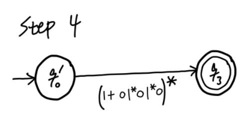
$\therefore\ 0^*10^*1(0+1)^*$

6.15 $\{w \in \{0,1\}^* :$ the number of 0 s in w is divisible by 3$\}$



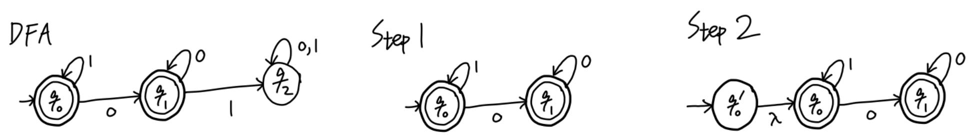


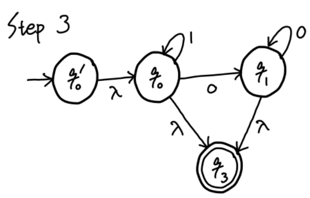


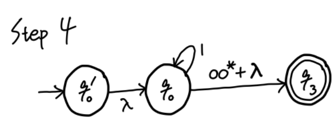


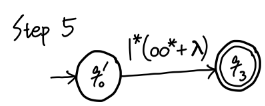
$\therefore\ (1+01^*01^*0)^*$

6.16 $\{w \in \{0,1\}^* : w$ does not have 01 as a substring $\}$









$\therefore\ 1^*(00^*+\lambda)$

7.2 Show that if x and y are odd length strings and z is an even length string that xyz is an even length string.

Let x and y be odd length of strings and z be an even length string.
WTS that xyz is an even length of string.
By the def. 7.2 in the text book, we know that there exist some integers p, q ≥ 0 such that $|x| = 2p + 1, |y| = 2q + 1$.
Also, by the def. 7.1 in the text book, we know that there exists some integer r ≥ 0 such that $|z| = 2r$.
Then, the length of the string xyz is the sum of the lengths of x, y, and z, or $2p + 1 + 2q + 1 + 2r = 2(p + q + r + 1)$. [def. 7.1]
Since we can write the length of xyz as $2s$ (where $s = p + q + r + 1 \geq 0$ is an integer), this means that it follows by def 7.1 that xyz is an even length string. $Q.E.D.$

7.8 Show that for all $n \geq 3, 4n^2 + 6n \leq 2n^3$.

Let us assume that $n \geq 3$. Then we can write

$$\begin{aligned} 4n^2 + 6n &\leq 4n^2 + 2n^2 \qquad \left[\text{ as } 6n \leq 2n^2 \text{ since } 3 \leq n\right] \\ &= 6n^2 \quad [\text{ math }] \\ &\leq 2n^3 \quad [\text{ as } 3 \leq n], \end{aligned}$$

which shows that $4n^2 + 6n \leq 2n^3$. Thus, $4n^2 + 6n$ is at most $2n^3$ for all $n \geq 3$. $Q.E.D.$

7.10 Define the NOR operation as $\text{NOR}\,(L_1, L_2) = \{x : x \notin L_1 \wedge x \notin L_2\}$. Show that regular languages are closed under the NOR operator.

Suppose L_1 and L_2 are regular languages.
WTS $\text{NOR}\,(L_1, L_2) = \{x : x \notin L_1 \wedge x \notin L_2\}$ is regular.
By the proof of theorem which was demonstrated in class that regular languages are closed under union, we know that $\{x : x \in L_1\} \cup \{x : x \in L_2\} = \{x : x \in L_1 \vee x \in L_2\}$ is regular.
Also by the proof of theorem which was demonstrated in class that regular languages are closed under complement, we know that $\{x : x \in L_1 \vee x \in L_2\}^c = \{x : x \notin L_1 \wedge x \notin L_2\}$ is regular. [De Morgan's laws]
Thus, regular languages are closed under the NOR operator by the theorems above. $Q.E.D.$

2 References

I have used the following external resource in purpose of learning general mechanism and procedure of state elimination algorithm for 5.20, 5.24, 6.13, 6.15, 6.16

`https://courses.cs.washington.edu/courses/cse311/14sp/kleene.pdf`