Regular Expressions and Finite State Automata

SOLUTIONS

CSC 28 Fall 2020

Jagan Chidella

1. a. The regular expression with the alphabet **{c,d}** for all strings:

"that begin with **c** and end with **c** and there is no more than one **d**" is:

(2 points)

(i) cdc
(ii) c (c+d)* c
(iii) c^*dc^*
(iv) $c^+ dc^+$

ANSWER:

The Solution to this problem does not exist in the choices.

The "closest choice" that exists is

$c^+ dc^+$

However, the correct answer is:

$C^+ (\lambda + d) C^+$ (here the + in λ+ d has the meaning λ or d)

since no more than one **d** also has the meaning that it includes no **d**'s at all, hence the λ.

This can also be written as: **$C^+ + C^+ d C^+$**

b. Provide two different possible example strings that the regular expression $r = a(a + b)^*$ generates in L(r). In this case, the + symbol has the meaning of Union symbol U. Notice that + is not a superscript. **(2 points)**

ANSWERS:

- aaabbaaabaabbbbba
- a

c. For the Regular Expression (RE) $r = a^*(\lambda + b)^*a^*$ Here the + has the meaning of Union symbol U. Notice that + is not a superscript.

Mark True or False next to each string if the string is generated (TRUE) and not generated (FALSE) by the RE r. **(8 points)**

(i) abb (TRUE)
(ii) bba (TRUE)
(iii) aa (TRUE)
(iv) b (TRUE)
(v) λ (TRUE)
(vi) ba (TRUE)
(vii) bbbb (TRUE)
(viii) a (TRUE)

d. Express in English the meaning of the following Regular Expressions: Here the meaning of a + b is same as a OR b or in other words, a Union b. **(10 points)**

(i) a* an empty string or a string of a's

(ii) a(bb)* a or a followed by even number of b's

(iii) (aa)*(bb)* nothing or a string of even number of a's followed by even number of b's or a string of even number of a's or a string of even number of b's (the same can be expressed in many ways)

(iv) (a + b)* any string of a's and b's

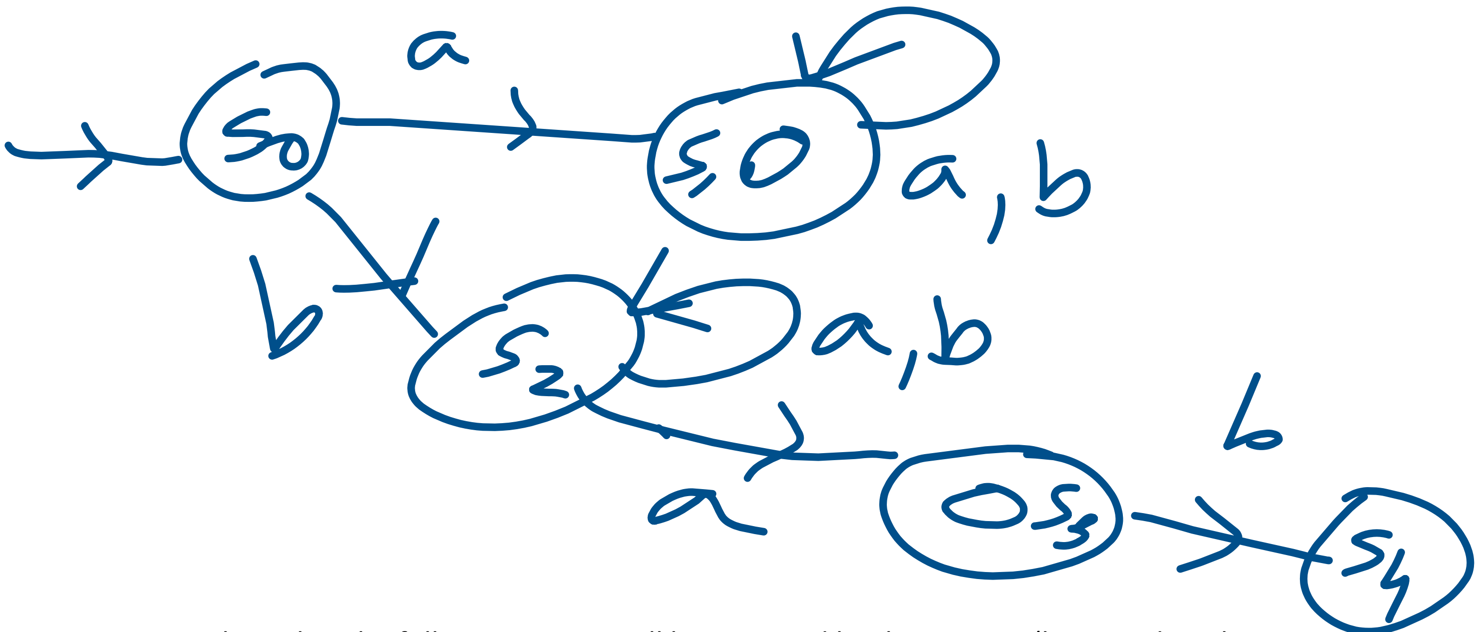
(v) (b)*b strings with one or more b's

2. (a) Define and draw a finite State Machine (FSA) M1 for the following patterns. Mark your Final States clearly.

   all strings that begin OR end with an **a**.

   Note that the alphabet A = {a, b} (in other words contains letters a and b only [Hint: first write the regular expression to make it easy for you]. **(6 points)**

Solution: r = a (a+b)* + b(a+b)*a (FSA is below for L(r); note that the transition from S3 to S4 is superfluous, this case can be handled by the transitions on S2).



Show that the following strings will be accepted by the FSA M1 (by providing the sequence of states and the type of the end state): **(5 points)**

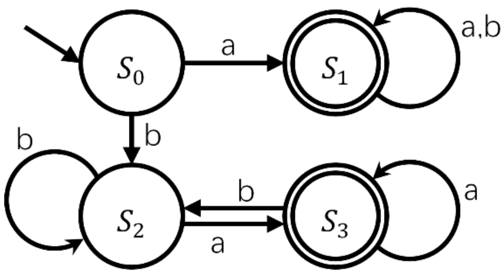
- abb S0 a S1 b S1 b S1 (S1 is a final state)
- bba S0 b S2 b S2 a S3 (S3 is a final state)
  Please note that after **bb** the next **a** can also go to S2 which is not a final state but the automata must see if there are other paths to a final state.
- abba S0 a S1 b S1 b S1 a S1 S1 is a final state)
- a S0 a S1 (S1 is a final state)
- aa S0 a S1 a S1 (S1 is a final state)

Show that the following strings will not be accepted by the FSA M1 (by providing the sequence of states and the type of the end state): **(5 points)**

- baab S0 b S2 a S2 a S2 b S2 (S2 is not a final state)
- bb S0 b S2 b S2 (S2 is not a final state)
- bbaab S0 b S2 b S2 a S2 a S2 n S2 (S2 is not a final state)
- b S0 b S2 (S2 is not a final state)
- λ S0 not a final state.

ADDENDUM for 2(a)

Please note that 2(a) may have an alternative FSA that looks like this one (where S_0 is the start state, S_1 and S_3 are final states). This is also called a deterministic FSA as there are no alternate choices for a transition on an element of the alphabet.



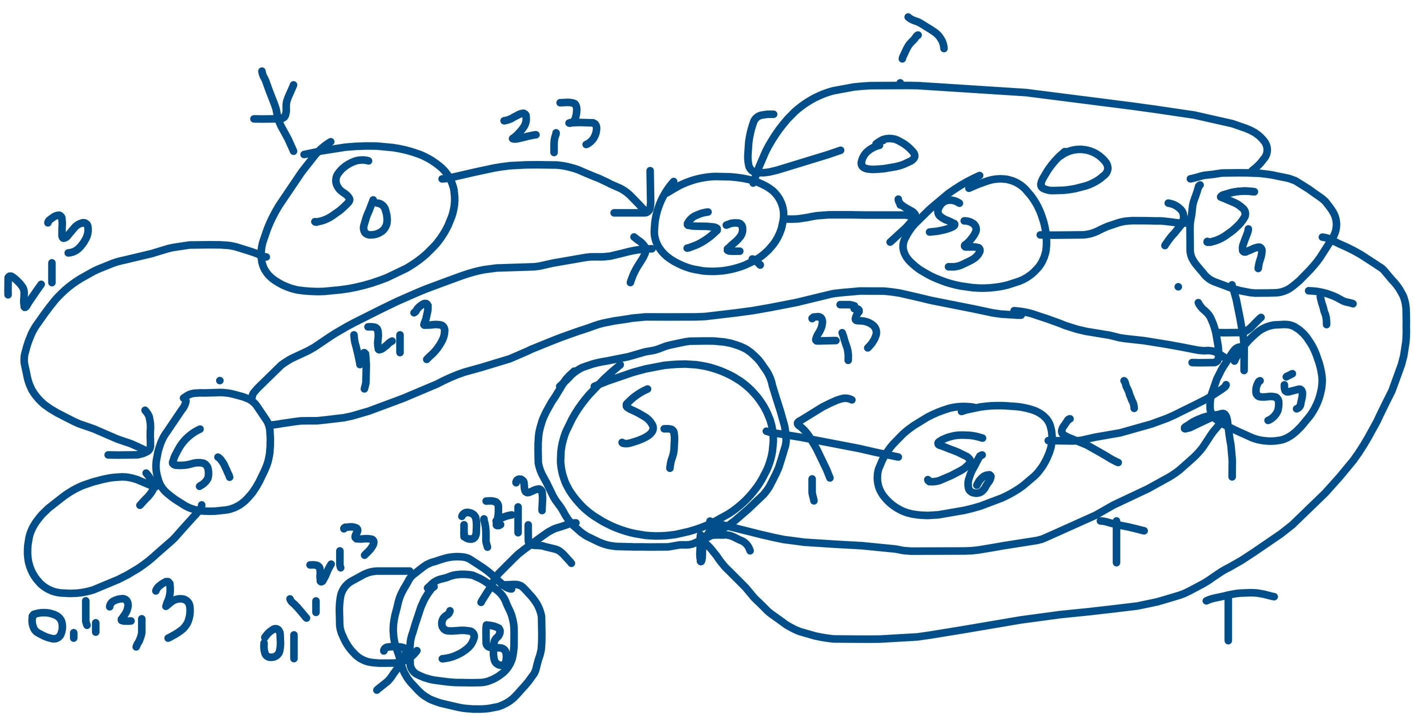
(b) Define and draw a finite State Machine (FSA) M2 for the following patterns. Mark your Start State and Final States clearly.

all strings that have even number of 0's followed immediately by even number if 1s. The string may start with anything other than 0 or 1 from the alphabet {0, 1, 2, 3}

6 points)

Solution:

This is like a research problem. This is a particularly hard problem! Never seen this before!!! 😊



S0 is the start state and S5, S7 are final states. Above, the transitions on **λ** are called **lambda-transitions.**

The regular expression is:
(2+3) (0+1+2+3)*((1+2+3)(00)$^+$(11)* + (2+3)(11)$^+$) (0+2+3) (0+1+2+3)* (Note the + on top of 11, since no zeros can imply 11). Of course, you may say zero number of 11s is also even number of 11s, hence the * on top of 11! The (1+2+3) before (00) and

Show that the following strings will be accepted by M2: (by providing the sequence of states and the type of the end state): **(3 points)**

- 2001111 S0 2 S2 0 S3 0 S4 λ S5 1 S6 1 S7 λ S5 1 S6 1 S7 (S7 is a final state)
- 333322211 S0 3 S1 3 S1 3 S1 3 S1 2 S1 2 S1 2 S5 1 S6 1 S7 (S7 is a final state)
- 323200 S0 3 S1 2 S1 3 S1 2 S2 0 S3 0 S4 λ S5 λ S7 (S7 is a final state)

Show that the following strings will not be accepted by M2: (by providing the sequence of states and the type of the end state): **(3 points)**

- 0011 No transition on 0 (S0 is not a final state)
- 20111 S0 2 S1 0 S1 1 S1 1 S1 1 S1 (S1 is not a final s
- 33320101 S0 3 S1 3 S1 3 S1 2 S2 0 S3 (no transition on 1, S3 is not a final state)