

1.a) < Automata theory story >

(2 marks)

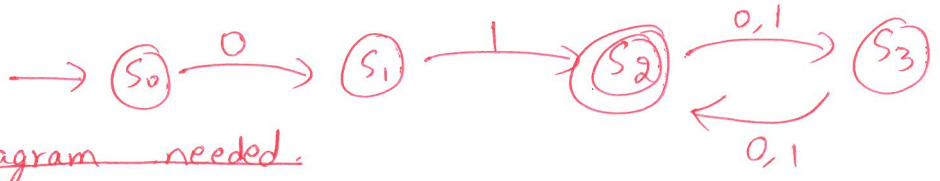
< 2-3 reasons >

(2 marks)

b) i)

First construct "begins with 01"

Notice that, after this 2 characters have been read, i.e. even. Every character after this can be 0/1 & flips even-odd



only diagram needed.
(3 marks)

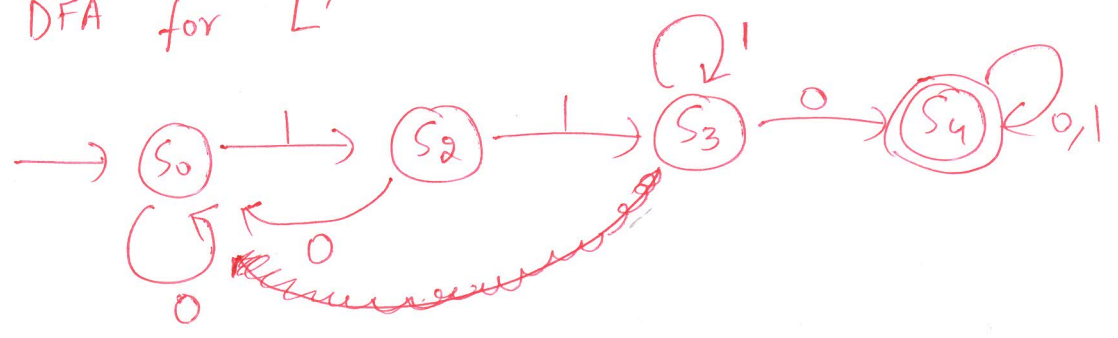
ii)

Either first construct DFA for opposite problem & flip accept-reject states, or draw directly; or draw NFA & convert
Opposite problem is:

$$L' = \{ w \mid w \text{ contains } 110 \}$$

if $s_i \in L'$ then $s_i \notin L$

DFA for L'

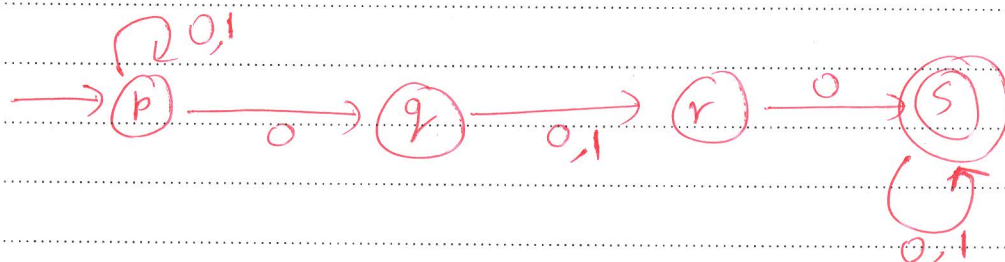


Now invert states to get answer:-



only diagram needed.
(3 marks)

2.a) First draw NFA to get clear picture



Let DFA start at state p:
then determine transitions w.r.t NFA

	0	1
p	{p, q}	{p}

~~The~~ created NFA would either go to p , or q
Let this be 1 state in DFA.

Now determine transitions from this

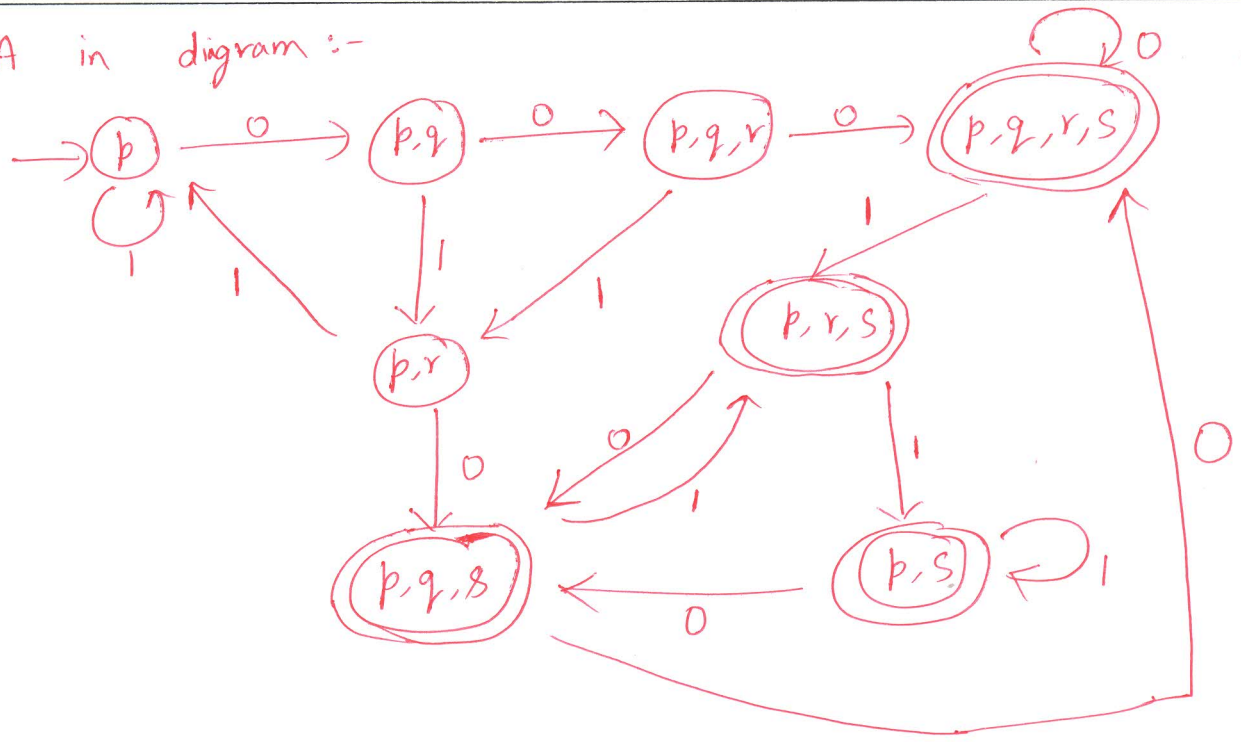
as :-

$$\delta_{DFA}(\{p, q\}, 0) = \delta_{NFA}(p, 0) \cup \delta_{NFA}(q, 0)$$

{p, q}	{p, q, r}	{p, r}
--------	-----------	--------

Now new states are
{p, q, r} & {p, r}
Determine transitions for this and fill table.

DFA in diagram :-



Either diagram or table
(5 marks)

Steps using union of δ
(1 mark)

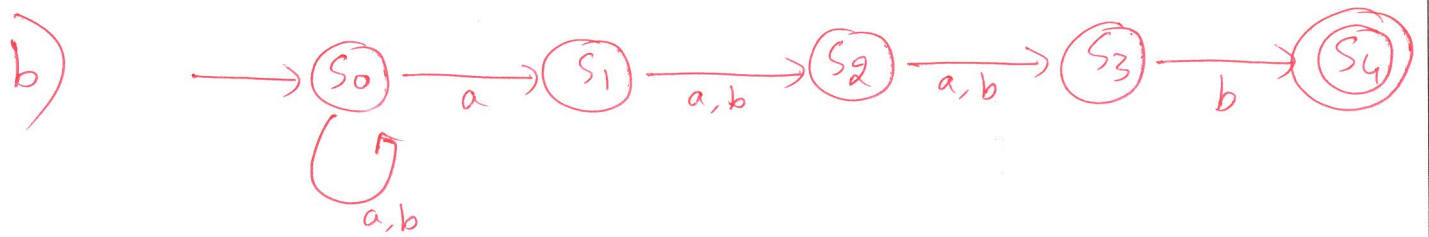


diagram or table
(4 marks)

3. a) i) $Eclosure(p) = \{p, q, r\}$

$Eclosure(q) = \{q\}$

$Eclosure(r) = \{p, q, r\}$

$Eclosure(s) = \{p, q, r, s\}$

(2 marks)

ii) $S_{DFA} = Eclosure(S_{NFA})$

$= Eclosure(p) = \{p, q, r\}$

Construct diagram or table
(3 marks)

b) Reg. Ex is short hand to DFA.
Reg Ex is an expression having operations on Regular Languages.

(1 mark)

i) $a?a?a?a?bbb^*$ or
 $(\epsilon + a + aa + aaa + aaaa + aaaaa)bbb^*$

ii) Either start with
 $(0+10)^*(\epsilon+1)$
Whenever we see a 1, next char must be 0 or end of string.

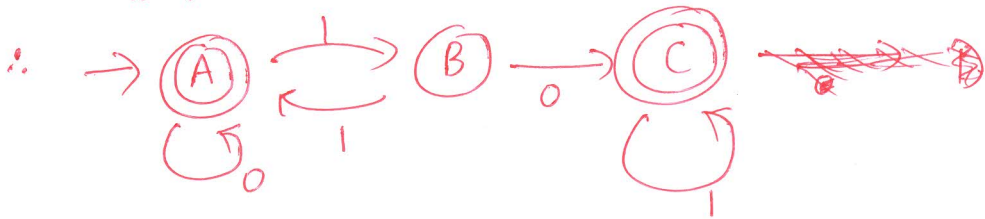
iii) $(a+b)^* a (a+b)(a+b)$

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

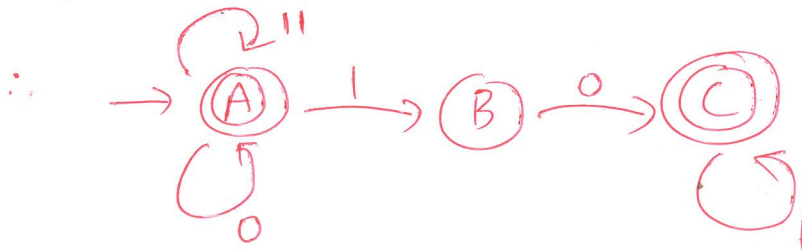
Whenever we see char, it must be in 3's.

(1x4 marks)

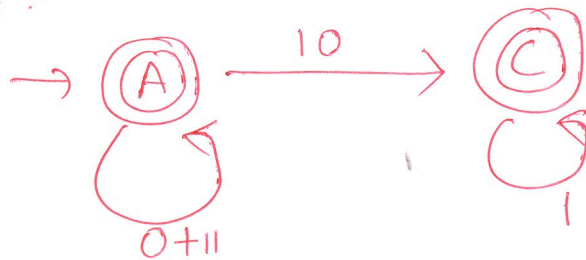
4.a) First eliminate D, as it is error state (no transition path from it to an accepting state)



Now, remove 1 transition from B to A.



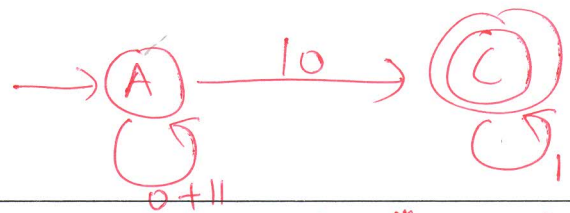
Removing B & adding :-



Now, consider only 1 as accepting state at a time.
if A is accepting



Reg Ex :- $(0+11)^*$



Reg Ex :- $(0+11)^* 10 (1)^*$

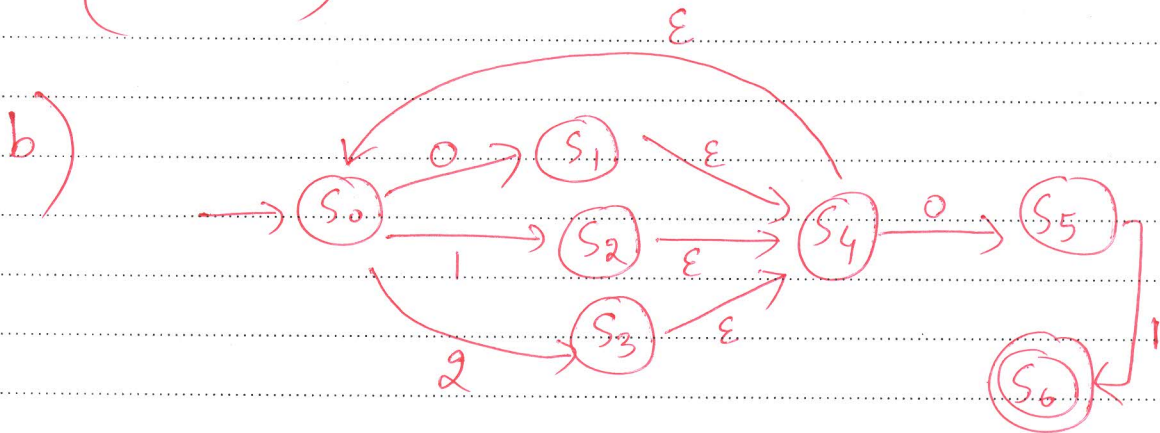
∴ Overall, or The 2 Reg Ex
 $(0+11)^* + (0+11)^*10(1)^*$

Final Answer:

(3 marks)

Steps

(2 marks)



$(0+1+2)$ requires branching
 * requires looping back:

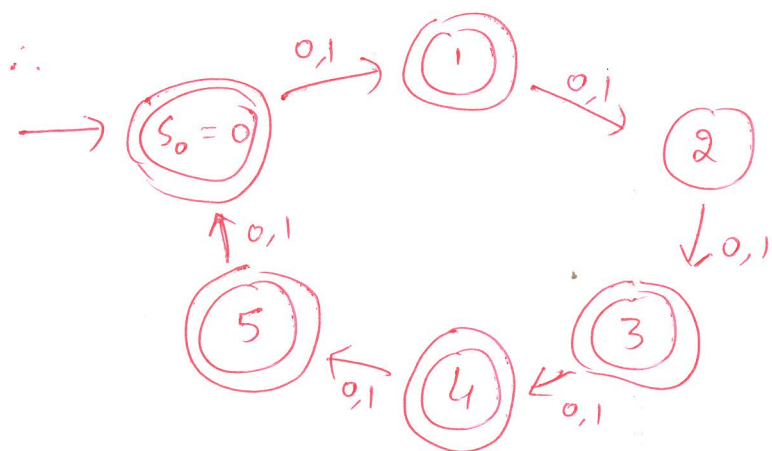
diagram with extra ϵ or without

(5 marks)

5.a) < Proof >
(6 marks)

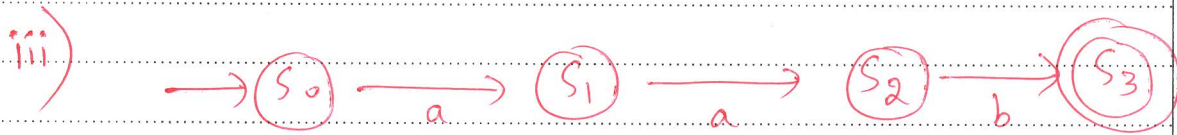
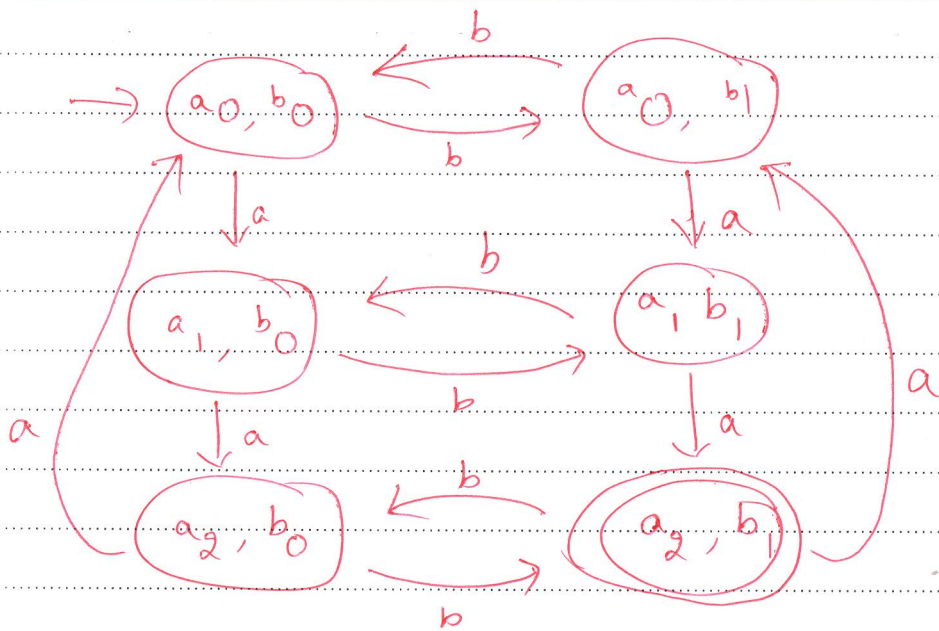
b) < differences >
(4 marks)

6.a) i) for any prefix s_i of w ;
 $|s_i| \bmod 6$ can take values $0, 1, 2, 3, 4, 5$.
 Accept if input gets over and value is not 2.

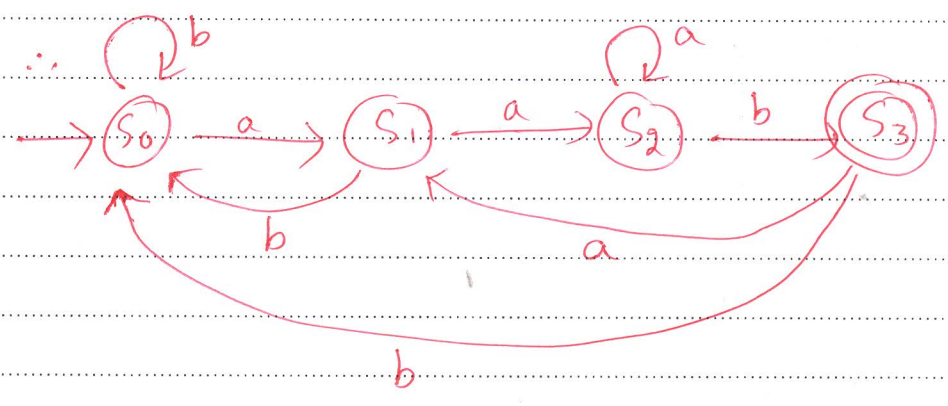


ii) 2 events are occurring
 $\#_a(s)$ can be $0, 1, 2$ and changes each time another 'a' is read.
 $\#_b(s)$ can be 0 or 1 .

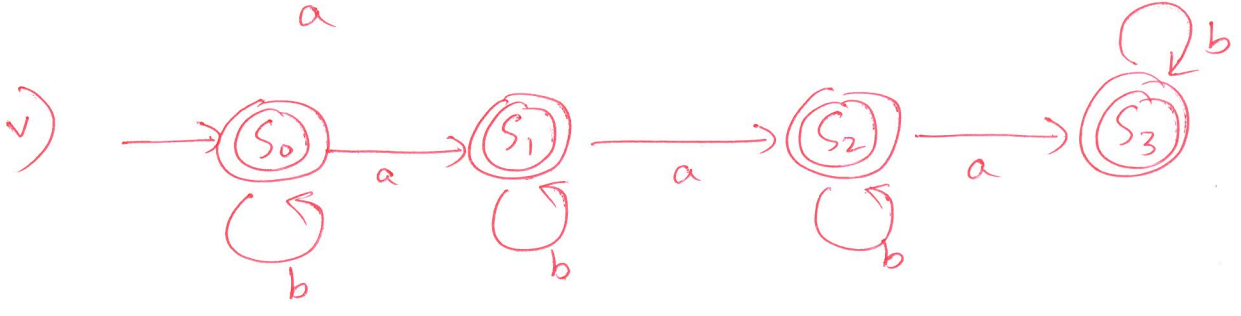
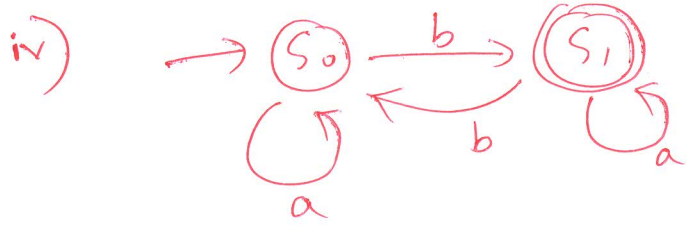




first draw transition for shortest string for which DFA should say yes. Then add transition for other alphabets at each state.

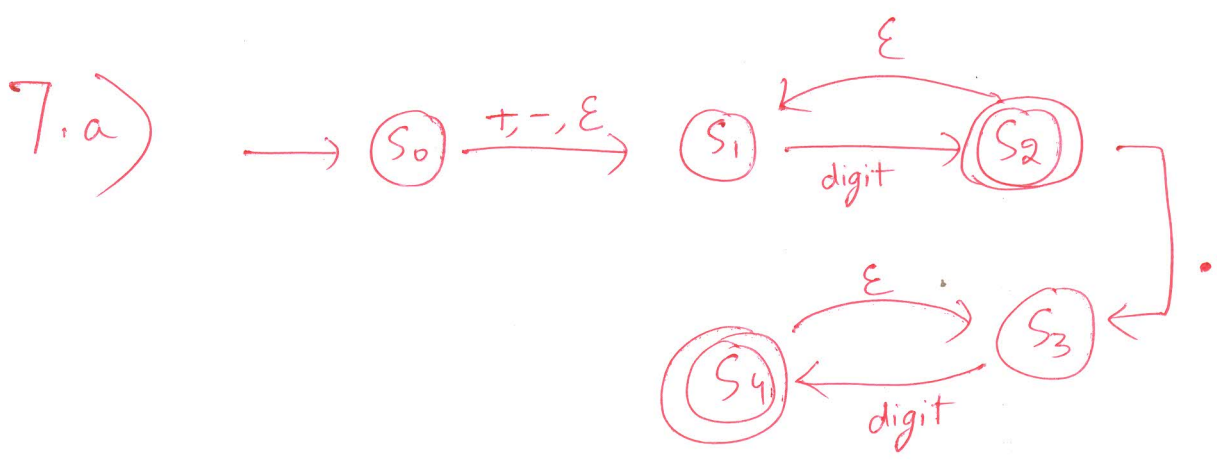


Reset properly if full aab not seen or if seen, input did not end.



Diagrams enough.

(2 x 5 = 10 marks)



Straight forward.

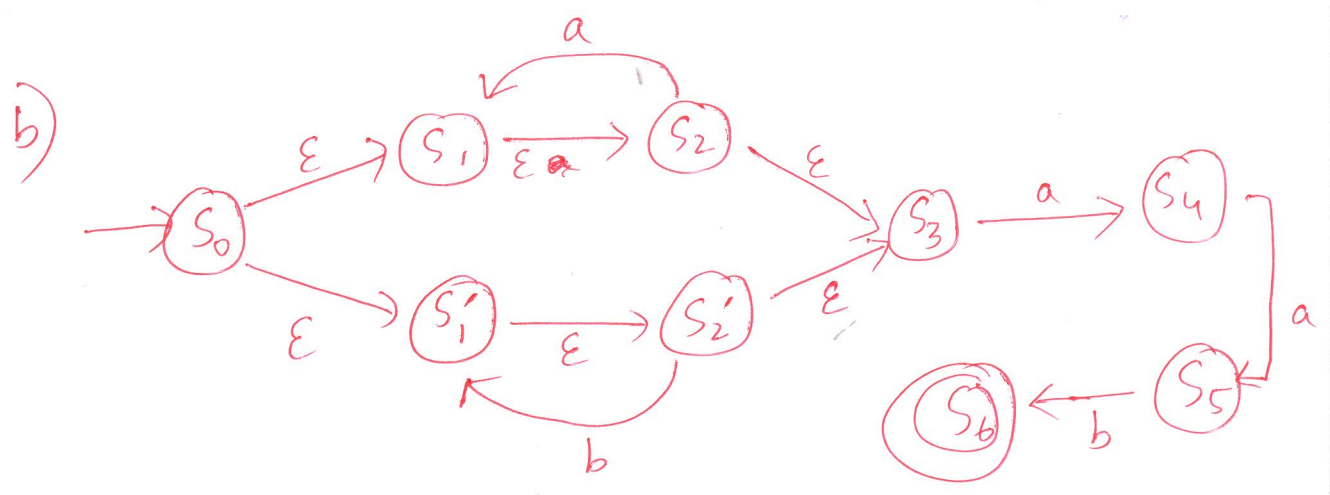


Diagram enough
(2 x 5 marks)