QUIZ 1 & 2 SOLUTION (SEC 13)

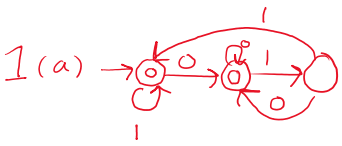
Thursday, July 24, 2025 8:22 PM

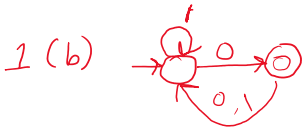
1. Two languages are defined as follows over the alphabets Σ = { 0 , 1 }:

   L_1 = { w | doesn't end with 01 }
   L_2 = { w | w ends with odd number of 0s }
   L_3 = { w | the number of '01' substrings is equal to the number of '10' substrings in w }

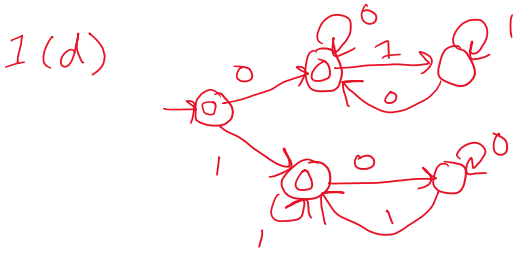
   a. Give a state diagram for that DFA that recognizes L_1 [4]
   b. Give a state diagram for that DFA that recognizes L_2 [4]
   c. If you were to use the "cross product" construction shown in class to obtain $L_1 \cup L_2$, how many states would it have? [3]
   d. Give a state diagram for that DFA that recognizes L_3 [3]

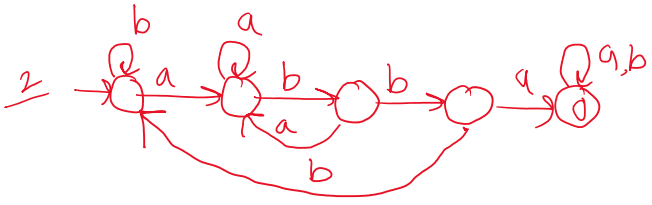
2. Design a DFA for the language L = { w | w contains "abba" as a substring } over the alphabets Σ = { a, b } [6]





1(c) 3x2 = 6





1. Let, Σ = {0, 1}. Give regular expressions for the following languages over Σ.

   a. { w : w doesn't end with 0 } [3]
   b. { w : w contains at least one 1 and odd 0's after the last 1 } [3]
   c. { w : w doesn't contain the substring "00" and starts and ends with 1} [3]
   d. { w : strings that have neither consecutive 1s, nor consecutive 0s } [3]
   e. { w : w is a string in which the number of 0s is odd. } [3]

2. Convert the following Regular expression into NFA: [5]

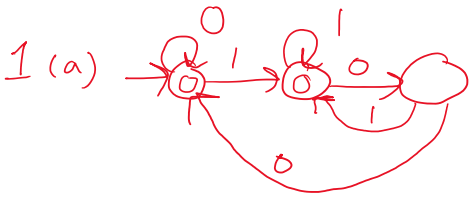
   (ab + ba)* | aba* (a + ba*)

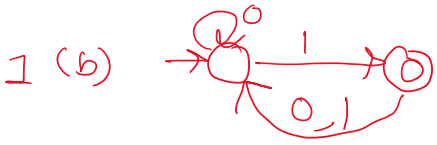
1(a) (0+1)* 1 + e
1(b) (0+1)* 1 (00)*0
1(c) 1 (1 | 01) * 1 + 1

1. Two languages are defined as follows over the alphabets Σ = { 0 , 1 }:
   L_1 = { w | doesn't end with 10 }
   L_2 = { w | w ends with odd number of 1s }
   L_3 = { w | the number of '01' substrings is not equal to the number of '10' substrings in w }

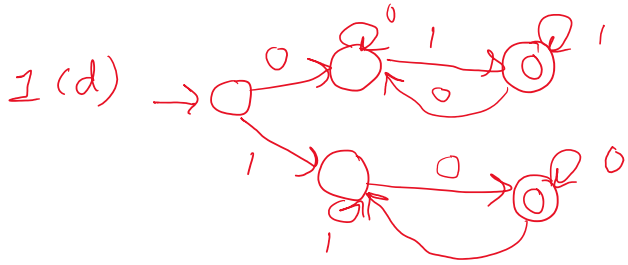
   a. Give a state diagram for the DFA that recognizes L_1 [4]
   b. Give a state diagram for the DFA that recognizes L_2 [4]
   c. If you were to use the "cross product" construction shown in class to obtain $L_1 \cup L_2$, how many states would it have? [3]
   d. Give a state diagram for the DFA that recognizes L_3 [3]

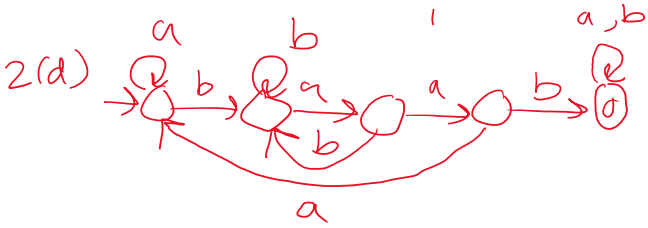
2. Design a DFA for the language L = { w | w contains "baab" as a substring} over the alphabets Σ = { a, b } [6]





1(c) ⇒ 3x2 = 6





1. Let, Σ = {0, 1}. Give regular expressions for the following languages over Σ.

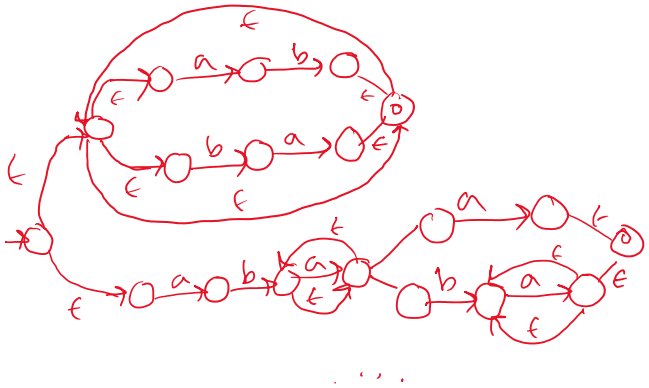
   a. { w : w doesn't start with 1 } [3]
   b. { w : w contains at least one 0 and odd 1's after the last 0 } [3]
   c. { w : w doesn't contain the substring "11" and starts and ends with 0} [3]
   d. { w : w contains only "01" and "10" substring in w } [3]
   e. { w : w is a string in which the number of 1's is even } [3]

2. Convert the following Regular expression into NFA: [5]

   ba*(a + ba)* | a* (ab* + b)

1(a) 0 (0+1)* + e
1(b) (0+1)* 0 (11)*1
1(c) 0 (0 | 10) * 0 + 0

1(a) (0+1)* 1 + e
1(b) (0+1)* 1 (00)*0
1(c) 1 (1 | 01) * 1 + 1
1(d) (10)*(1+e) + (01)* (0+e)
1(e) (1* 0 1* 0 1*)* 1* 01*



1(a) 0 (0+1)* + e
1(b) (0+1)* 0 (11)*1
1(c) 0 (0 | 10) * 0 + 0
1(d) (10)*(1+e) + (01)* (0+e)
1(e) (0* 1 0* 1 0*)* + 0*