CSE 331: Automata & Computability

Prepared By: KKP

Practice Sheet: Equivalence between Regular Expressions and Finite Automata

Converting DFAs to Regular Expressions Using State Elimination Method

Notes:

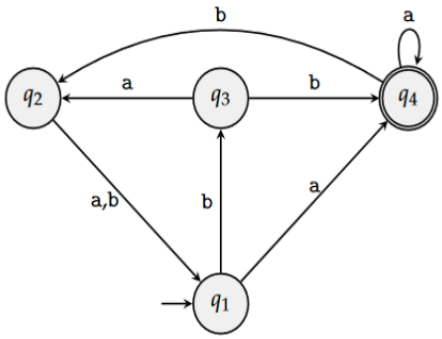
a. **If you use any other method apart from the State Elimination Method, you will be awarded 0 points**

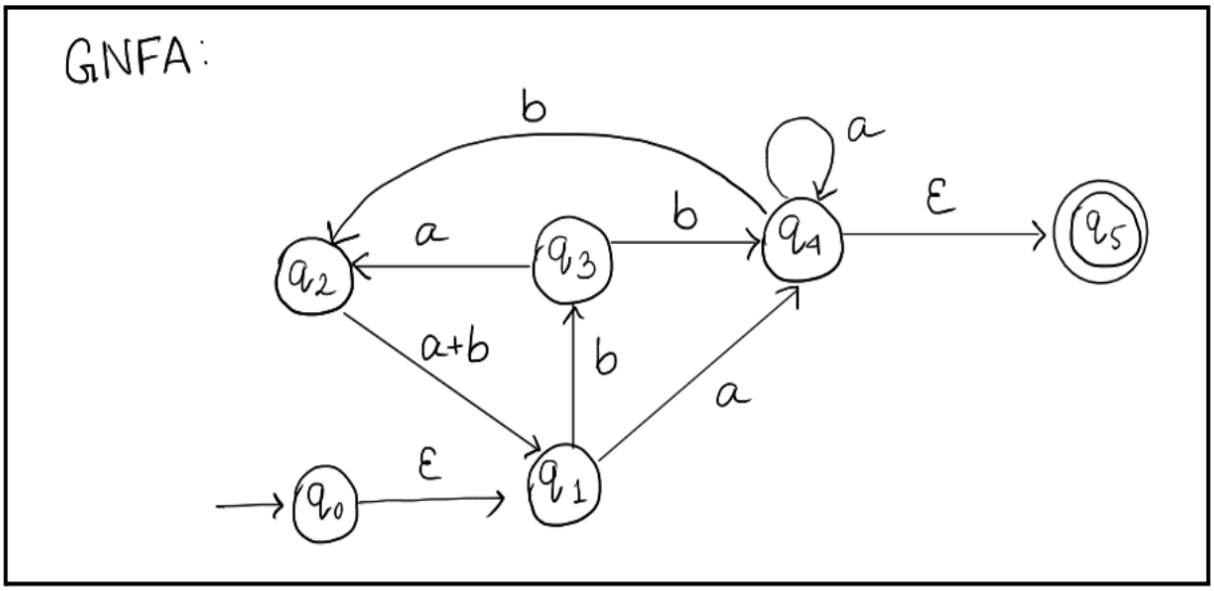
b. **If you don't follow the order of elimination mentioned in the question, you will be awarded 0 points.**

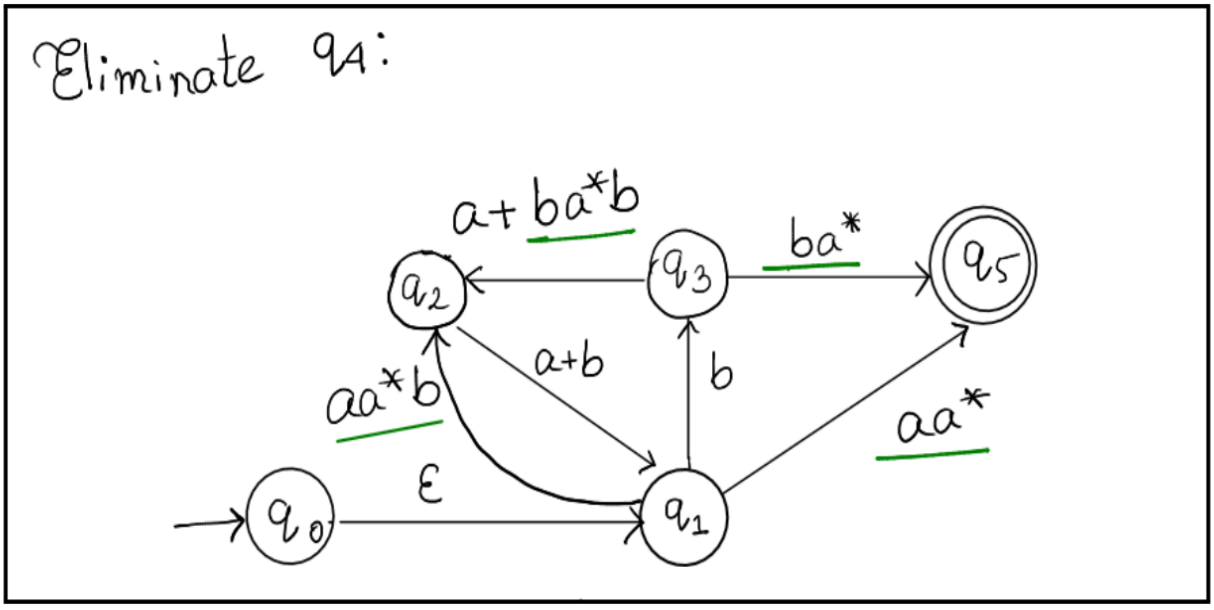
Practice

Problem 4 (CO2): Converting Finite Automata to Regular Expressions (10 points)

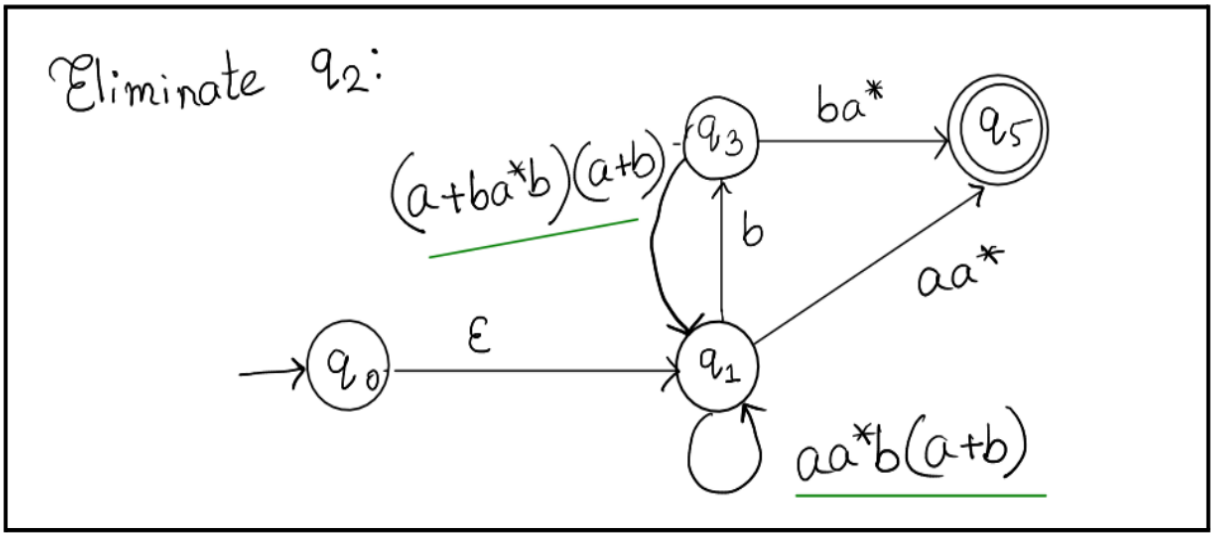
Convert the following DFA into an equivalent regular expression using the state elimination method. First eliminate q_4, then q_2, next q_3, and finally q_1. You must show work.

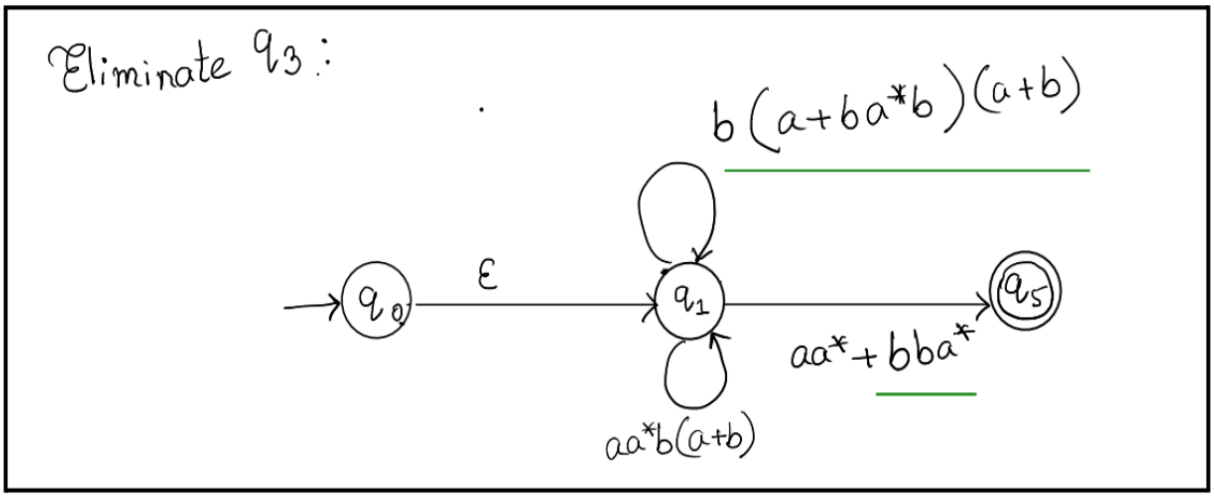


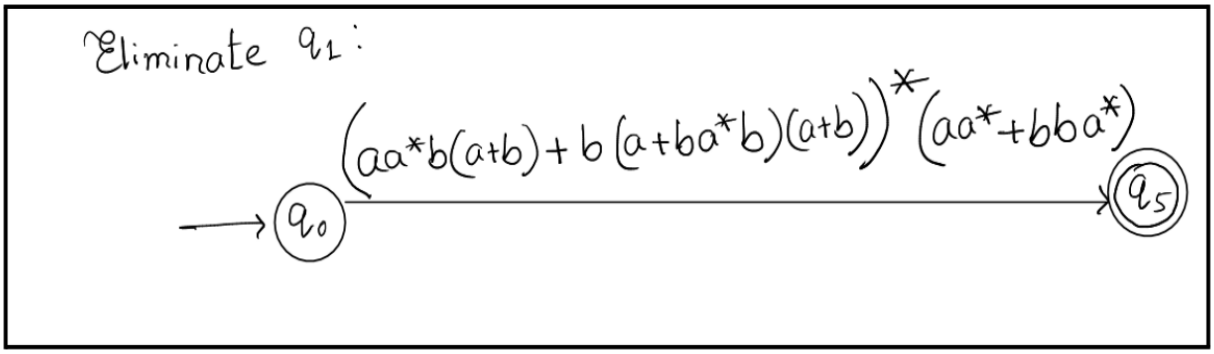




The parenthesis is a must (q3-q1)



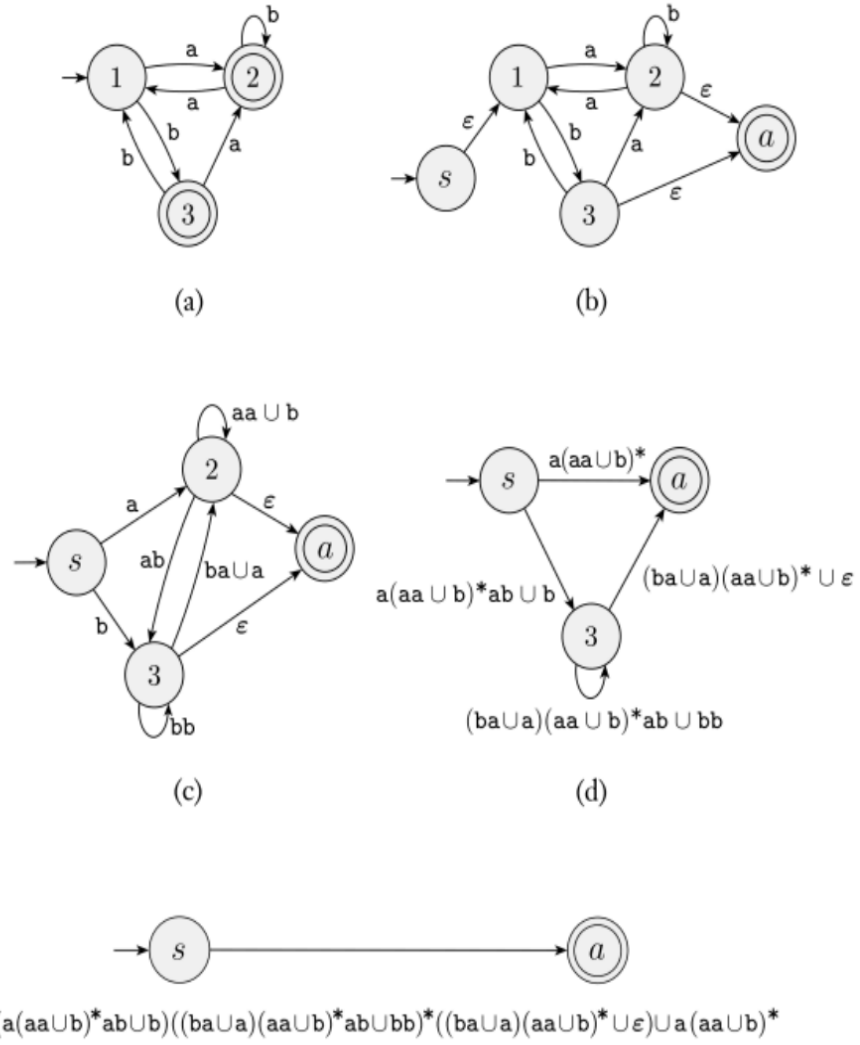




Book's Example

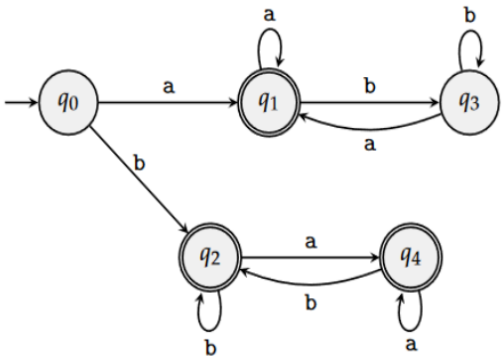
EXAMPLE 1.68

In this example, we begin with a three-state DFA. The steps in the conversion are shown in the following figure.



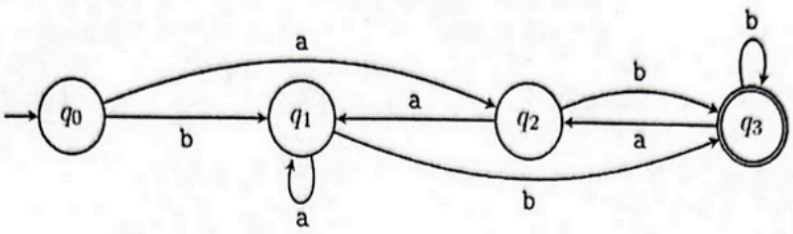
Problem 1:

Convert the following DFA into an equivalent regular expression using the state elimination method. First eliminate q_1, then q_2, next q_3 and then finally q_4. You must show work.



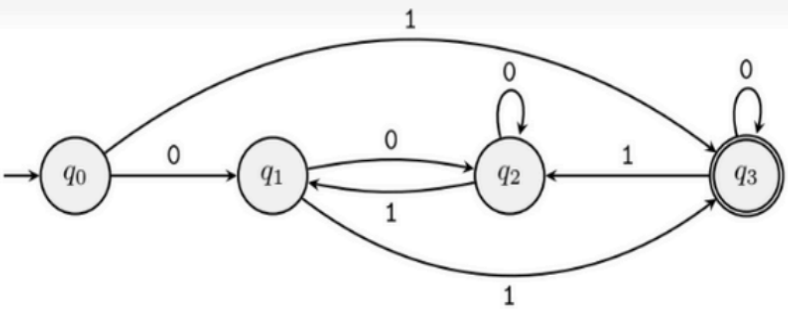
Problem 2:

Convert the following DFA into an equivalent regular expression using the **state elimination method**. First eliminate q2, q1 and finally q3. You must show work.



Problem 3:

Convert the following DFA into an equivalent regular expression using the **state elimination method**. First eliminate q3, q1 and finally q2. You must show work.



Converting Regular Expressions to NFAs

Problem 4:

$$(a^*b^* + (ac + b^*c)\ b^*a)^*$$

Problem 5:

$$a(bca^*)^* + (ba + ca)\ a^+(ab + ac) + ((c^*a)\ ab)^*b^*$$