Problem Set 3

Fall 11

Due: 4th October, 2011, 11:00 am before review class begins

Please follow the homework format guidelines posted on the class web page:

`http://www.cs.uiuc.edu/class/fa11/cs373/`

1. [**Category**: Regular Expression Construction and Comprehension, **Points**: 20]

   For a and b, construct regular expressions defining the specified languages over the alphabet $\{a, b\}$. For c and d, answer True if the given pair of regular expressions defines the same language, and False otherwise.

   (a) All strings in which the number of times a appears is a multiples of 3.

   (b) All strings w such that every maximal substring of a's in w is of a length that is a multiple of 3 (E.g., *baaabaaabb*, *baaaaaabaaab*, *bbb* are in the language, while *baaaab*, *ba* are not).

   (c) $(ab)^*a$ and $a(ba)^*$

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

Solution:

(a) $(b^*ab^*ab^*ab^*)^*$

(b) $(aaa + b)^*$

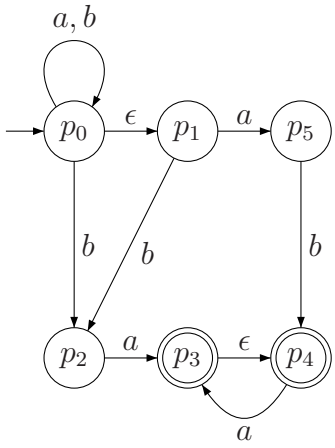
(c) True

(d) True

2. [**Category**: NFA to Regular Expression, **Points**: 20]

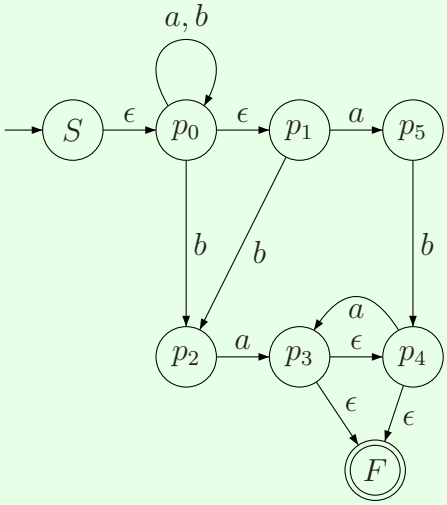
   Convert the following NFA M to a regular expression that describes the same language that M accepts. We require you to construct this regular expression using the systematic procedure outlined in class for converting NFAs to regular expressions using GNFAs.

   Show each step of the construction, as you eliminate a state using the algorithm, You must the intermediate steps to obtain full credit. Don't just provide the final regular expression.

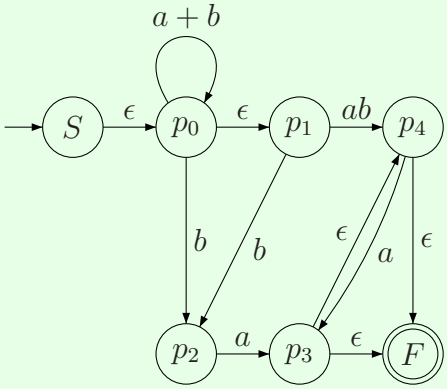


Solution:

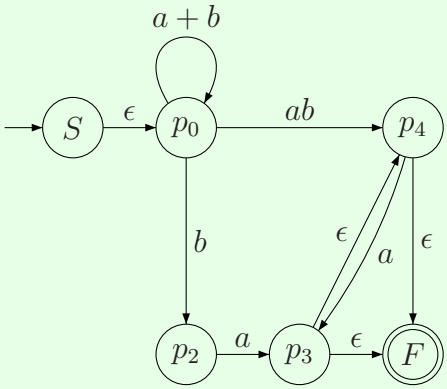
The GNFA with new initial stae and final state is



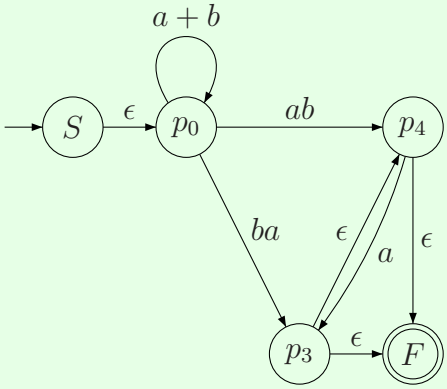
Eliminate p_5 by replacing some transition functions by regular expression.



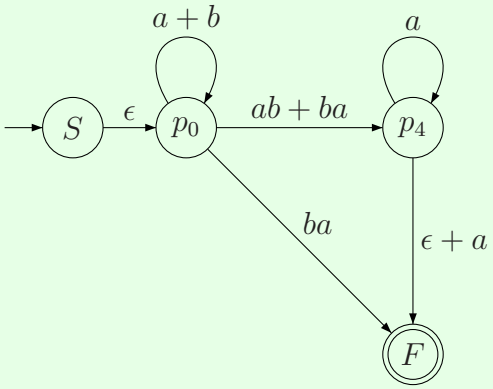
Then eliminate p_1:



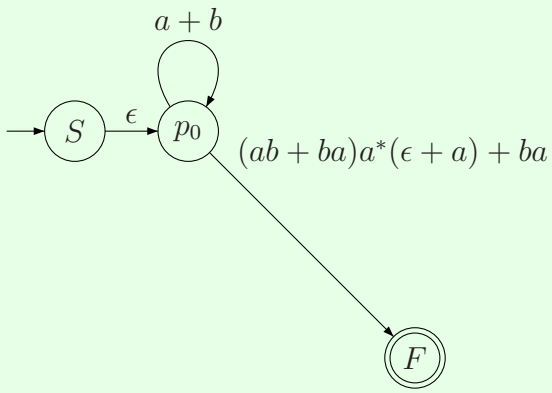
Then eliminate p_2.



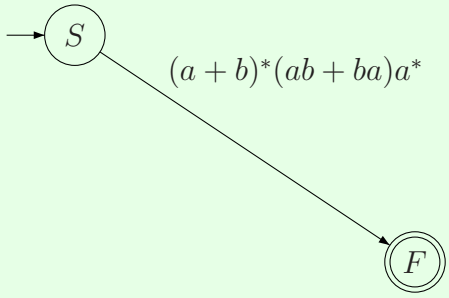
Then eliminate p_3



Then, eliminate p_4.



Finally, eliminate $p0$



Thus, the corresponding regular expression is $(a+b)^*(ab+ba)a^*$.

3. [**Category**: Regular Expression to NFA, **Points**: 20]

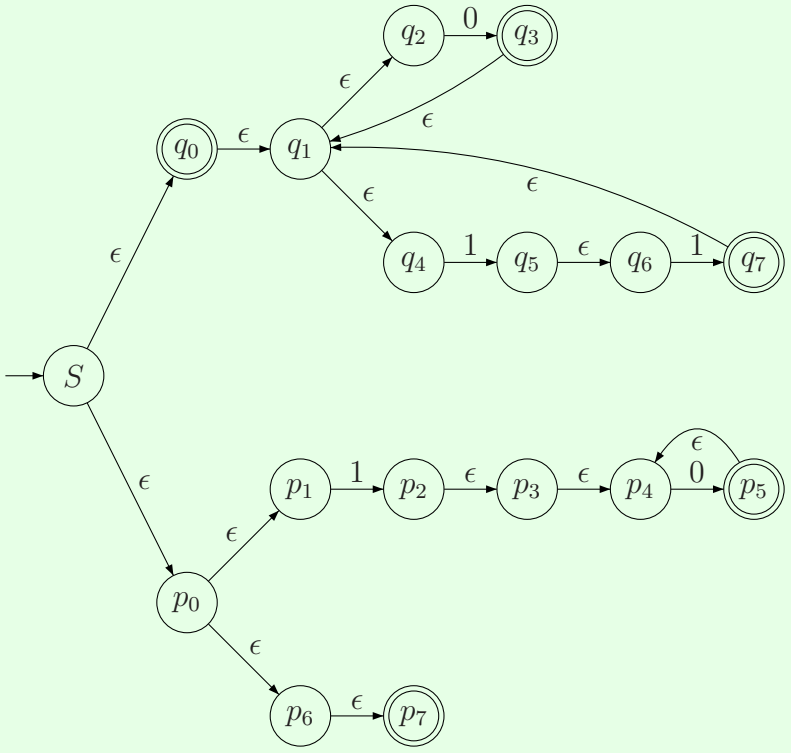
Construct an NFA for the follwoing regular expression, $(0 + 11)^* + (10^* + \epsilon)$.

Don't reduce your NFA. Follow the steps in textbook on constructing NFAs from regular expressions (using closure properties of NFAs under union, *, and concatenation). That is, use ϵ transitions and extra initial state and final state appropriately.

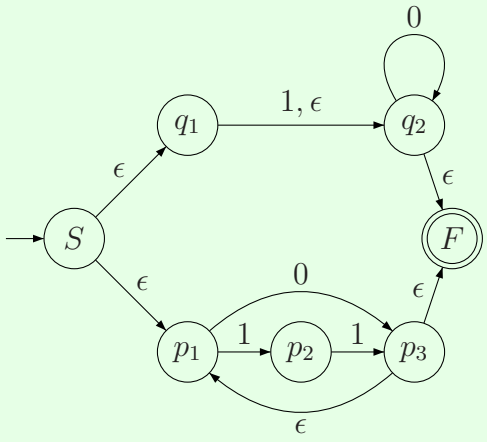
You don't need to draw the trap state for this NFA.

Solution:

The NFA based on the algorithm in the textbook without any simplification is



The somewhat simplified NFA is


0
q_1
$1, \epsilon$
q_2
ϵ
ϵ
S
F
ϵ
0
ϵ
p_1
1
p_2
1
p_3
ϵ