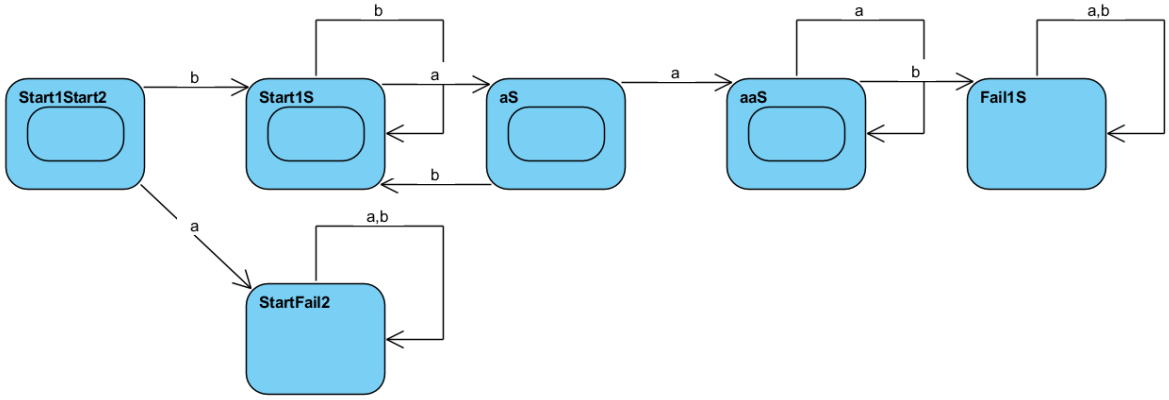
Michael Wang
CS 181 HW 3

1)



2a)
FSL – the intersection of two FSL is an FSL

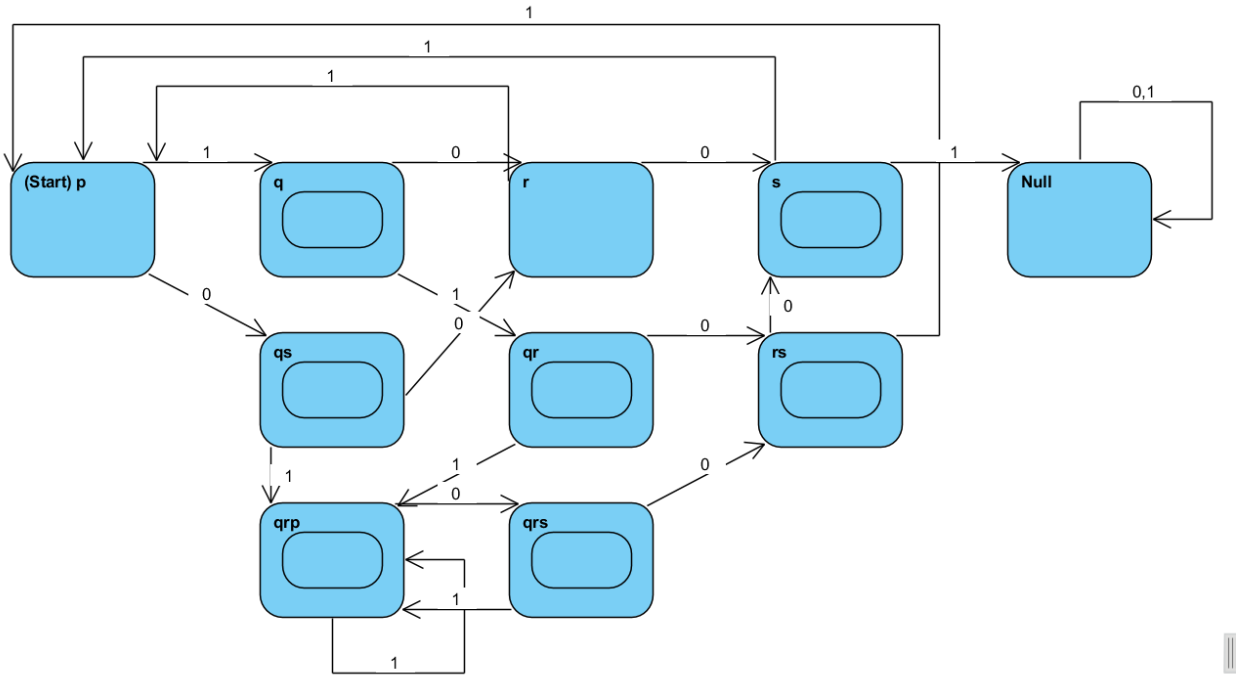
2b)
Unknown – no information is provided of C. If C is a Non-FSL then definitely not since the intersection of a FSL and non-FSL is not an FSL.

2c)
Unknown – no information is provided of C. If C is a Non-FSL then definitely not since the union of a FSL and non-FSL is not an FSL.

2d)
Non-FSL – the union of two FSL is an FSL. Since A is a FSL and the union is not, X cannot be an FSL

2e)
Unknown – the empty set is clearly a FSL, but X could also be “intersection of B with some Non-FSL”

3)



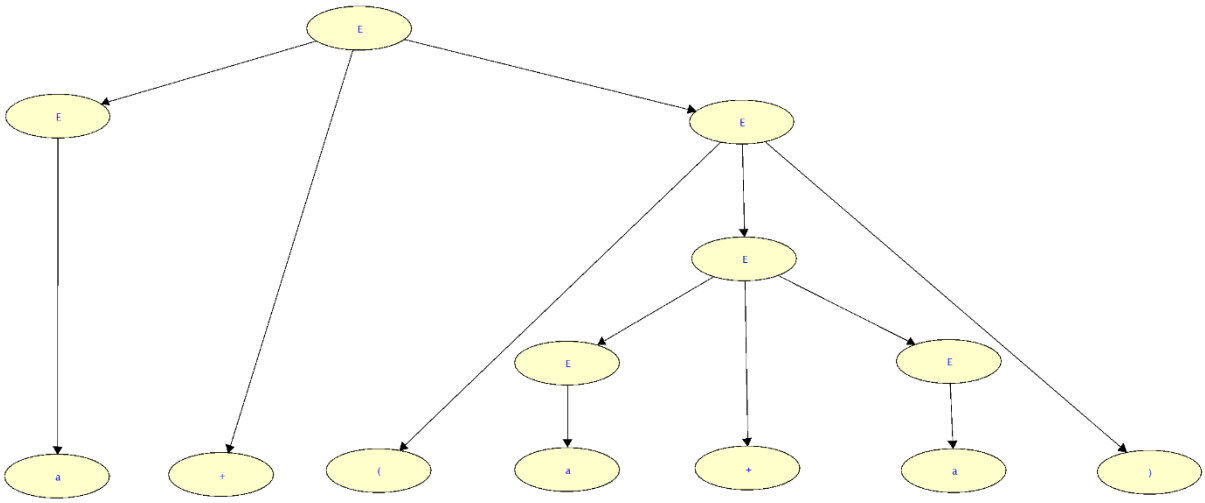
The powerset of the states of N are Q_D = {p, q, r, s, pq, pr, ps, qr, qs, rw, pqr, pqs, prs, qrs, pqrs, Null}. The start state of D is p. The accept states are all states in Q_D containing q or s. Starting from p in Q_N, I computed all states reachable using 0 and all states reachable using, taking their respective unions and adding the transition to δ'. I repeated this process until all states in the DFA had two outputs meaning the DFA had been fully specified.

4a)
Yes, go q2 on ab* using a, go to q1 on a* using null, go to q_{accept} on b* using b

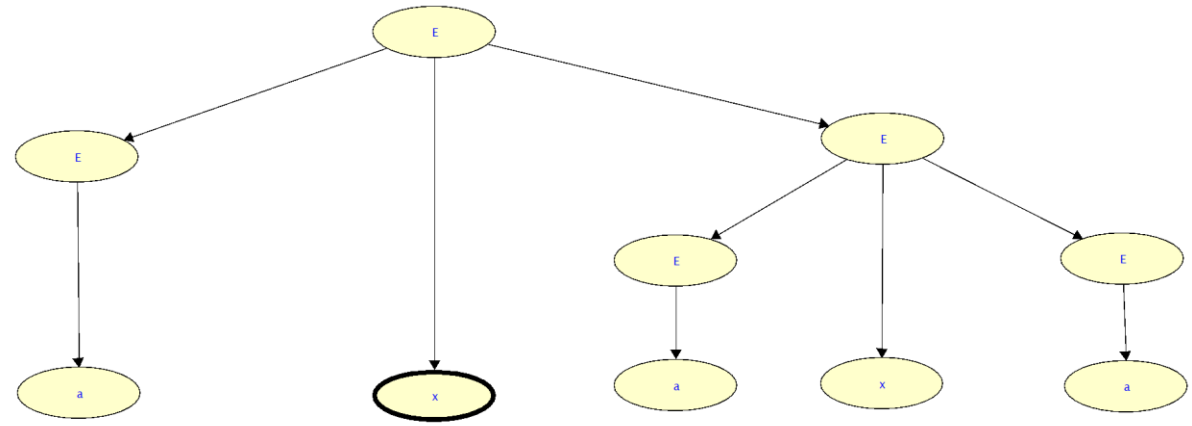
4b)
Yes, go to w2 on ab* using a, go to q1 on a* using a, go to q_{accept} on b* using bb

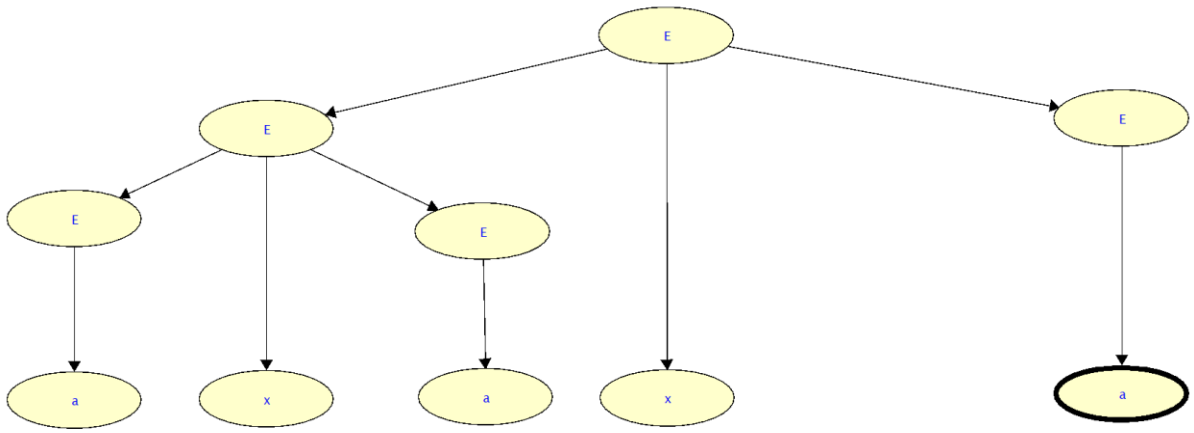
5)
(aUb)*a(aUb)a(aUb) U (aUb)*b(aUb)b(aUb)

6a)



6b)





6c)
Left tree: E → ExE → E+ExE → a+ExE → a+axE → a+axa
Right tree: E → E+E → a+E → a+ExE → a+axE pp> a+axa

7)
E → 01 | 10 | 0R1 | 1R0 | 0E0 | 1E1
R → 0R0 | 1R1 | 00 | 11
(Start symbol is E, variables are E and R)

8)
The given language is a FSL as demonstrated by the following NFA: