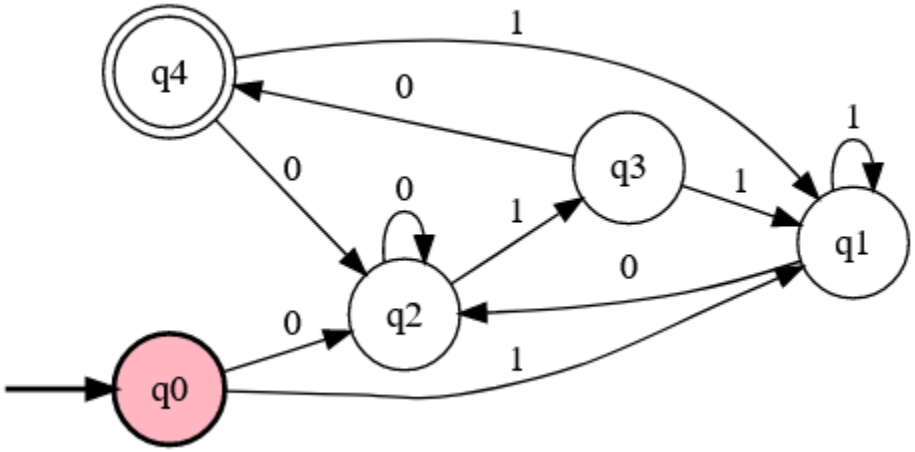
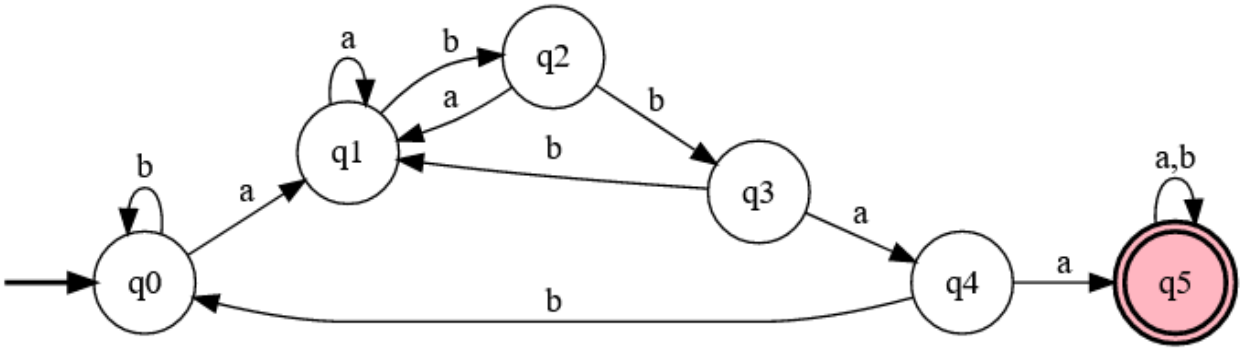
1. Construct the DFA for Σ = {0, 1} which accepts any string that ends with 010.



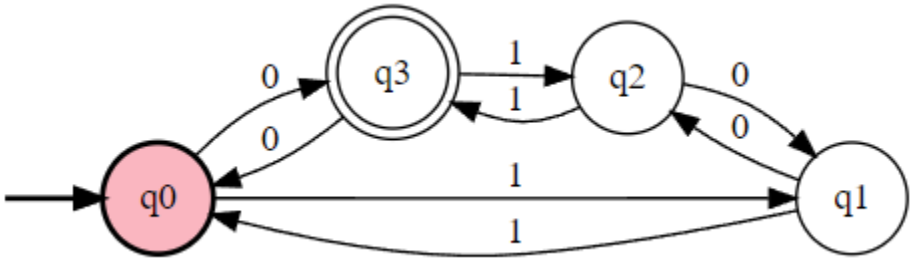
2. Construct the DFA for Σ = {a, b} that accepts any string with ‘abbaa’ as a substring. Write state set, start state, final state(s) and transition table.



{q0, q1, q2, q3, q4, q5}
q0
q5

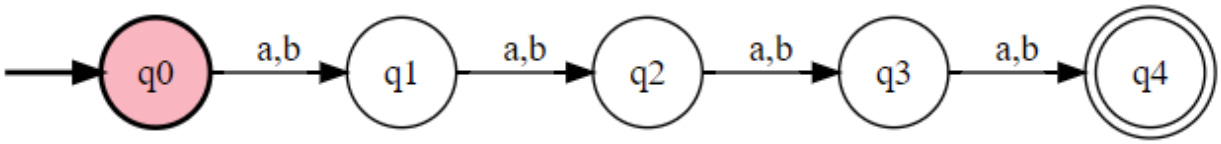
	a	b
q0	q1	q0
q1	q1	q2
q2	q1	q3
q3	q4	q1
q4	q5	q0
q5	q5	q5

3. Construct the DFA for Σ = {0, 1}, which accepts all strings that have an even number of 1's and an odd number of 0's. Write state set, start state, final state(s) and transition table.



#states
q0
q1
q2
q3
q4
q5
#initial
q0
#accepting
q5
#alphabet
a
b
#transitions
q0:a>q1
q0:b>q0
q1:a>q1
q1:b>q2
q2:a>q1
q2:b>q3
q3:a>q4
q3:b>q0
q4:a>q5
q4:b>q0
q5:a>q5
q5:b>q5

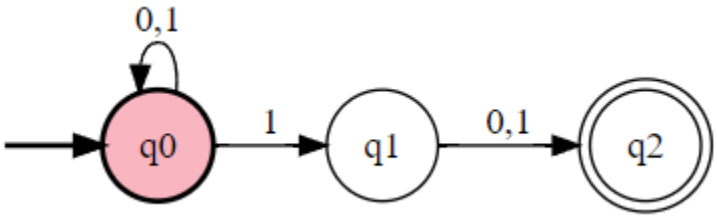
4. Construct the NFA for a language L = {set of all strings over (a,b) with length 4}.



#states
s0
s1
s2
s3
s4
#initial
s0
#accepting
s4
#alphabet
a
b
#transitions
s0:a>s1
s0:b>s1
s1:a>s2
s1:b>s2
s2:a>s3
s2:b>s3
s3:a>s4
s3:b>s4

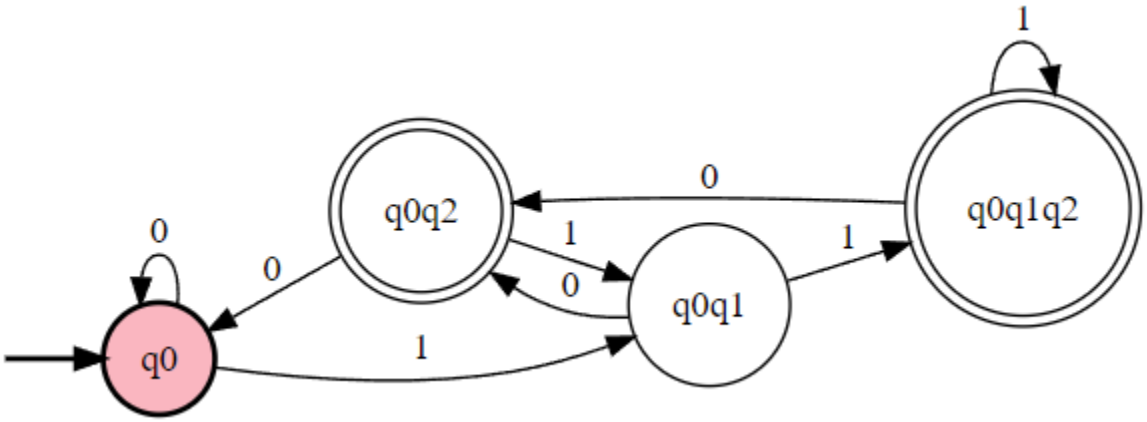
5. Construct the NFA for a language L = {set of all strings over (0,1) with second last digit '1'}. Convert it to its equivalent DFA.

NFA:



#states
q0
q1
q2
q3
q4
#initial
q0
#accepting
q2
#alphabet
0
1
#transitions
q0:0>q0
q0:1>q0,q1
q1:0>q2
q1:1>q2

DFA:



#states
q0
q0q1
q0q2
q0q1q2
#initial
q0
#accepting
q0q2
q0q1q2
#alphabet
0
1
#transitions
q0:0>q0
q0:1>q0q1
q0q1:0>q0q2
q0q2:1>q0q1
q0q2:0>q0
q0q1:1>q0q1q2
q0q1q2:1>q0q1q2
q0q1q2:0>q0q2

6. Aa

7.