UIU QUESTION BANK

MID-TERM QUESTION SOLUTIONS

THEORY OF COMPUTATION

CSE 2233

SOLUTION BY

NURUL ALAM ADOR

UPDATED TILL ***FALL 2023***

nurulalamador.github.io/**UIUQuestionBank**

Index

1. Design DFAs that accept the following languages:

   a) L = accepts any string that has an even number of 'a' or odd number of 'b' over alphabet {a, b}

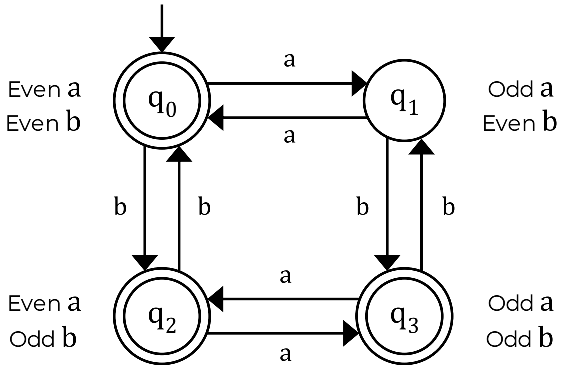
   b) L = accepts any string which starts with an odd number of 'r' and ends with 'brb' over the alphabet {b, r}

   c) L = accepts any string where the last two symbols are different over the alphabet {a, b}

Solution:

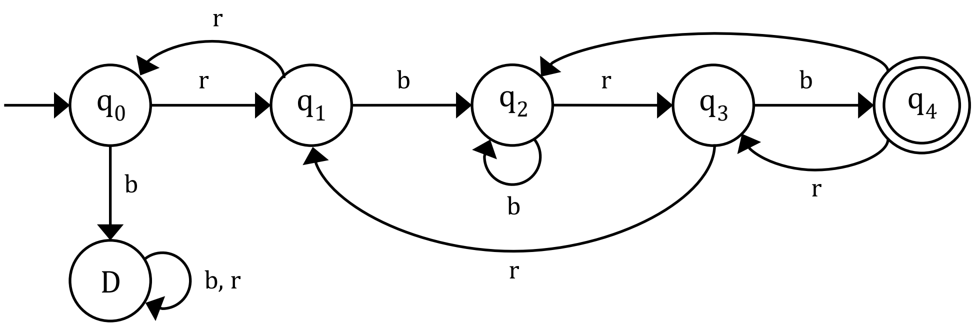
a) L = accepts any string that has an even number of 'a' or odd number of 'b' over alphabet {a, b}

The DFA has been designed below:



b) L = accepts any string which starts with an odd number of 'r' and ends with 'brb' over the alphabet {b, r}

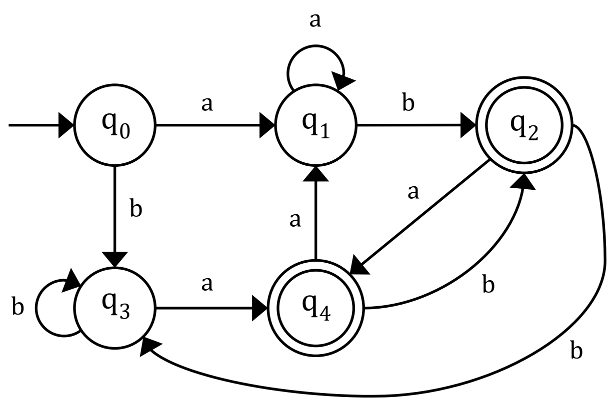
The DFA has been designed below:



c) L = accepts any string where the last two symbols are different over the alphabet {a, b}

The DFA has been designed below:

[P.T.O]



2. Design NFAs that accept the following languages:

a) L = { w | w starts with 'a' or 'b' and contains 'cca' and ends with 'b' or 'c' } | $\sum = \{a, b, c\}$

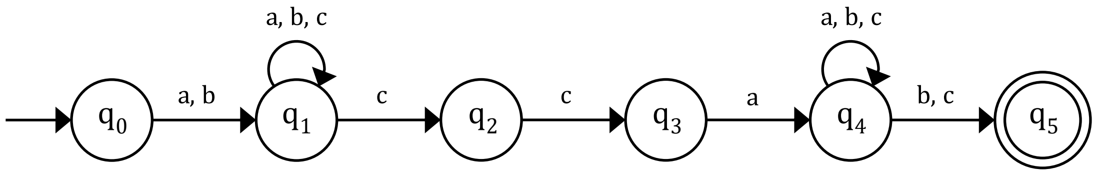
b) L = { w | w starts and ends with different symbols with total length of at least 2 } | $\sum = \{a, b, c\}$

c) L = { w | w contains 'xyz' or 'yzx' or 'zxx' and ends with 'yz' } | $\sum = \{x, y, z\}$

Solution:

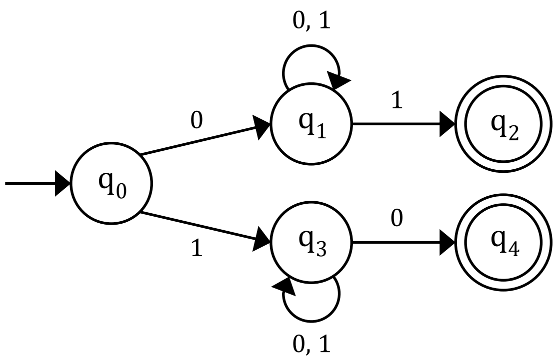
a) L = { w | w starts with 'a' or 'b' and contains 'cca' and ends with 'b' or 'c' } | $\sum = \{a, b, c\}$

The NFA has been designed below:



b) L = { w | w starts and ends with different symbols with total length of at least 2 } | $\sum = \{a, b, c\}$

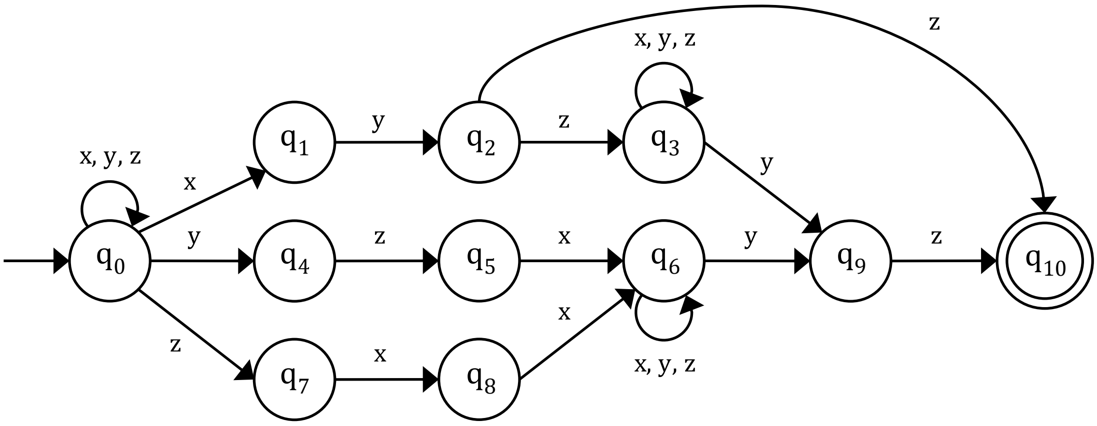
The NFA has been designed below:



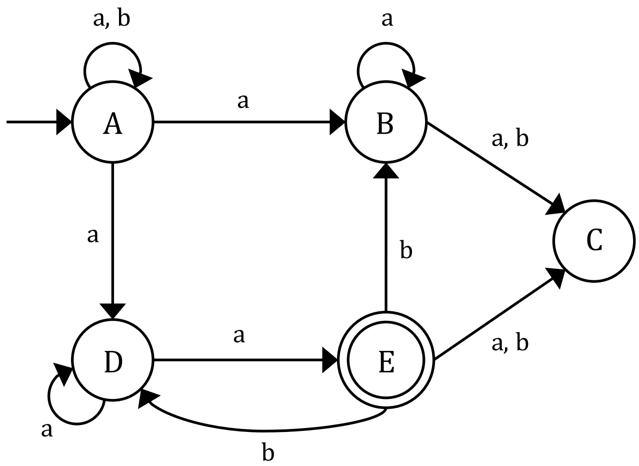
c) L = { w | w contains 'xyz' or 'yzx' or 'zxx' and ends with 'yz' } | $\sum = \{x, y, z\}$

The NFA has been designed below:

[P.T.O]



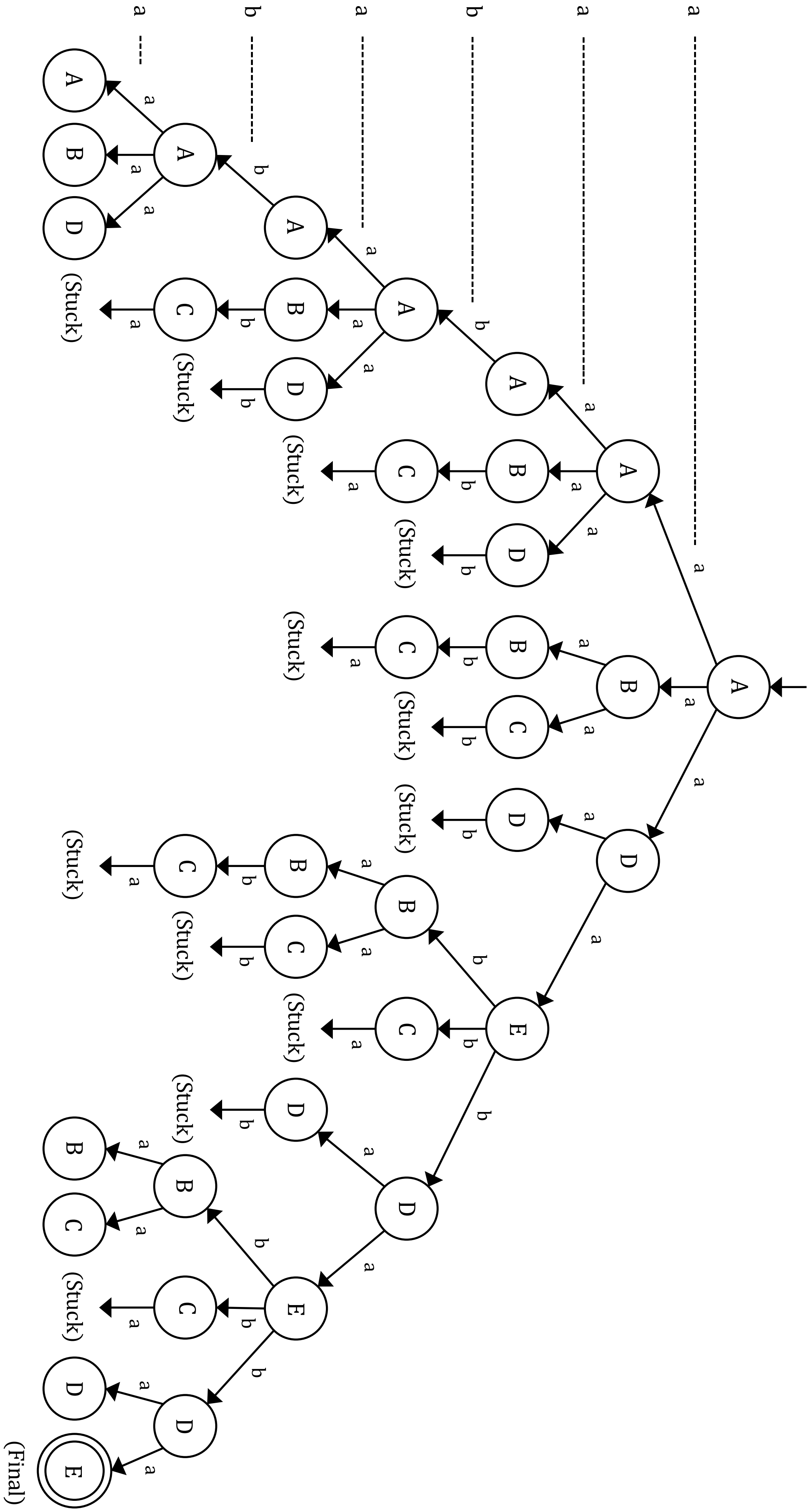
3. Consider the following NFA, and show with help of NFA-tree whether the string "aababa" is accepted.



Solution:

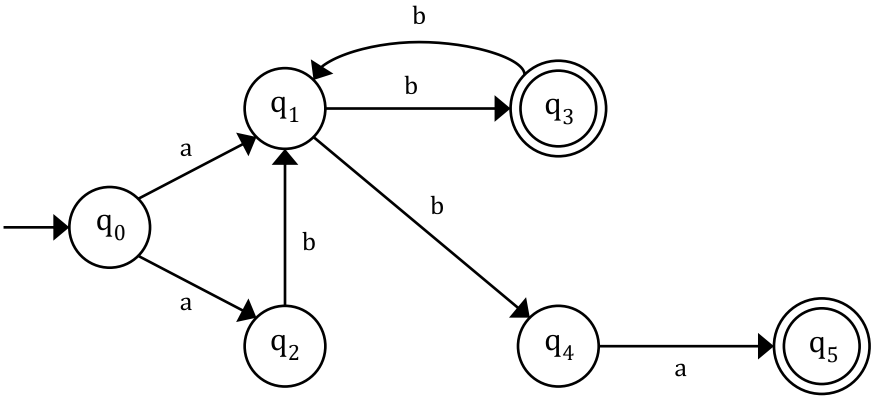
NFA Tree:

[P.T.O]


a
a
b
a
b
a
(Stuck)
(Final)

With help of NFA-tree, we can see the string "aababa" reach the final state.
∴ The string "aababa" is accepted.

4. Convert the following NFA over the alphabet $\sum = \{a, b\}$ to an equivalent DFA



Solution:

Transition Table of the given NFA:

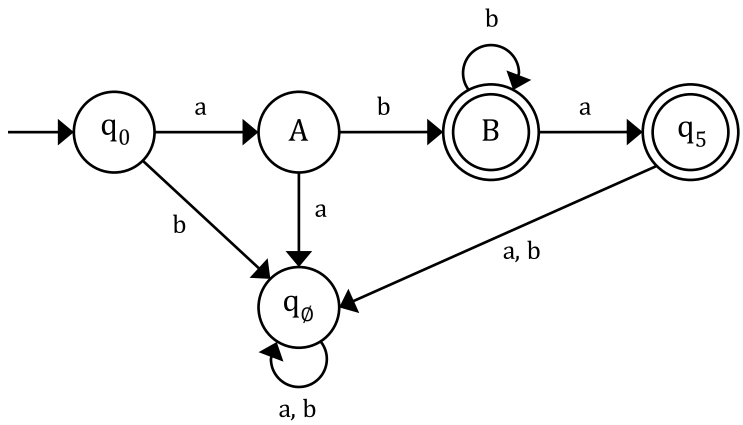
	a	b
→ q_0	q_1, q_2	Ø
q_1	Ø	q_3, q_4
q_2	Ø	q_1
∗ q_3	Ø	q_1
q_4	q_5	Ø
∗ q_5	Ø	Ø

Transition Table for the Equivalent DFA:

	a	b
→ q_0	$\{ q_1, q_2 \}$ = A	{ Ø } = $q_Ø$
A = $\{ q_1, q_2 \}$	{ Ø } = $q_Ø$	$\{ q_1, q_3, q_4 \}$ = B
∗ B = $\{ q_1, q_3, q_4 \}$	q_5	$\{ q_1, q_3, q_4 \}$ = B
∗ q_5	{ Ø } = $q_Ø$	{ Ø } = $q_Ø$
$q_Ø$ = { Ø }	{ Ø } = $q_Ø$	{ Ø } = $q_Ø$

Equivalent DFA Diagram:

[P.T.O]



5. a) Convert the following regular expressions to finite automata:

i) (ab)* + (a + ab)* b* (a + b)*

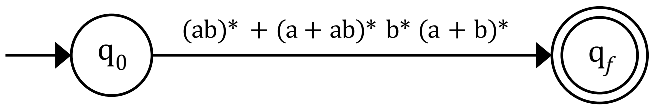
ii) [a + ba (a + b)]* a (ba)* b*

Solution:

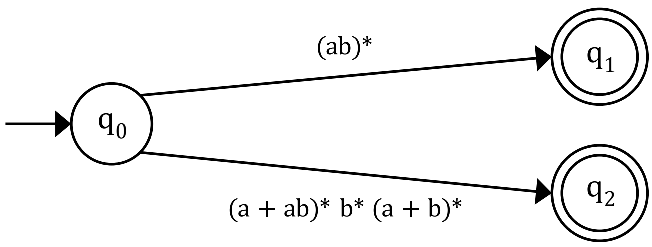
i) (ab)* + (a + ab)* b* (a + b)*

The equivalent finite automata for the following regular expression has been constructed below:

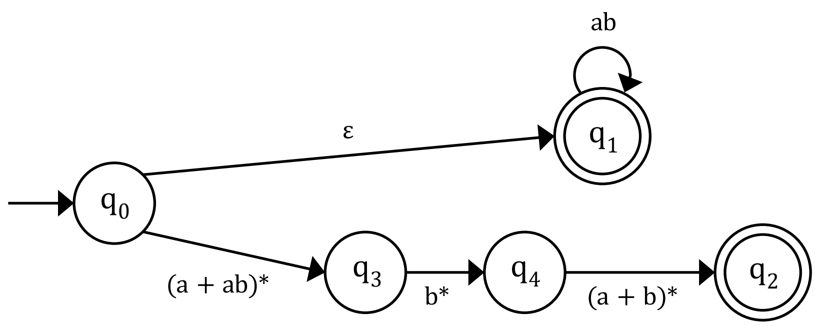
Step 1:



Step 2:

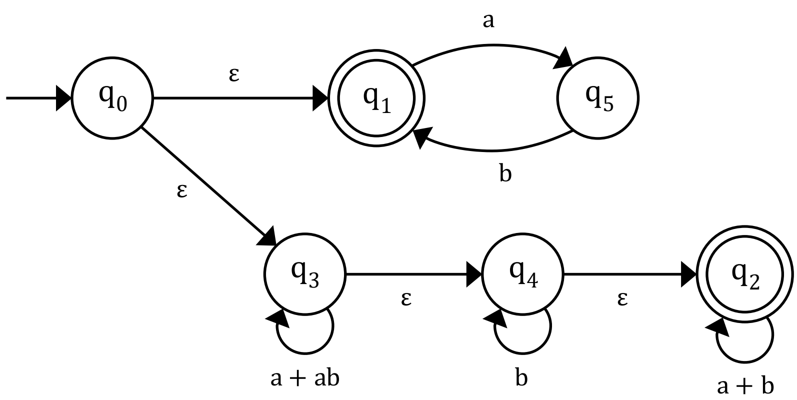


Step 3:

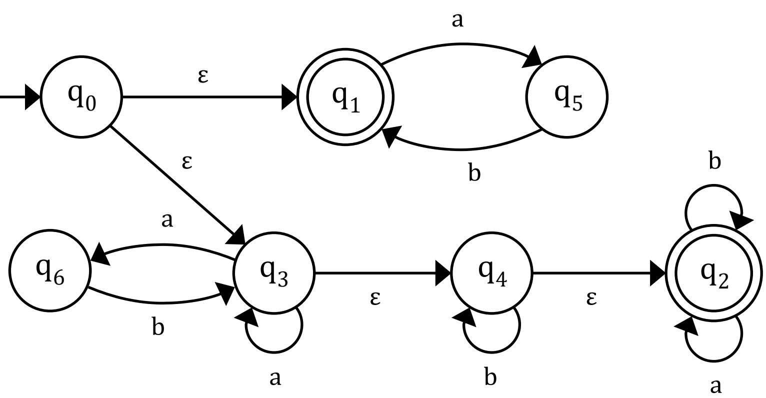


[P.T.O]

Step 4:



Step 5:

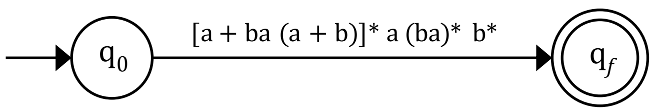


This is our final finite automata for following regular expression.

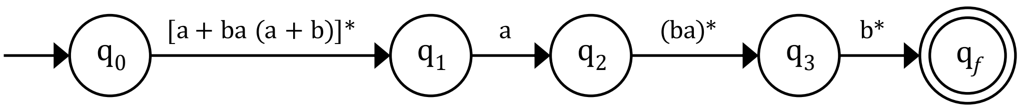
ii) [a + ba (a + b)]* a (ba)* b*

The equivalent finite automata for the following regular expression has been constructed below:

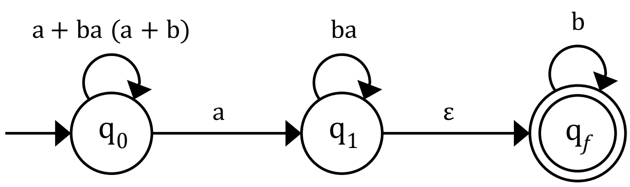
Step 1:



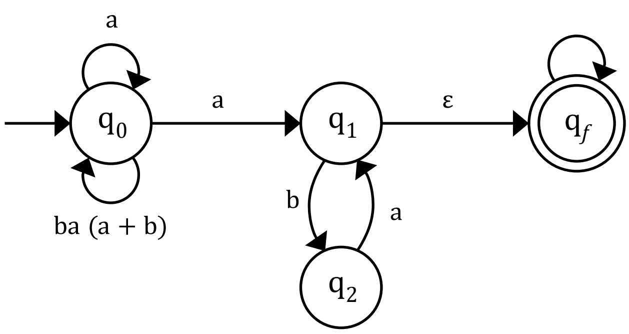
Step 2:



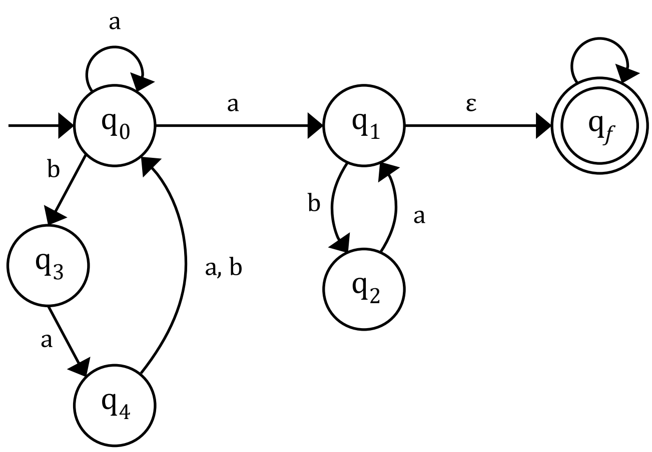
Step 3:



Step 4:



Step 5:



This is our final finite automata for following regular expression.

b) Convert the following regular expressions to finite automata:

i) L = { strings such that the 4th symbol from the right is b over the alphabet {a, b} }

ii) L = { strings such that they start and end with 'a' over the alphabet {a, b, c} }

Solution:

i) L = { strings such that the 4th symbol from the right is b over the alphabet {a, b} }

Regular Expression: (a | b)* a (a | b) (a | b) (a | b)

ii) L = { strings such that they start and end with 'a' over the alphabet {a, b, c} }

Regular Expression: a (a | b | c)* a | a

Summer 2023

1. Design **DFA**'s that accepts the following languages:

a) $L = \{ w \mid w \text{ does not contain '101'} \} \mid \Sigma = \{0,1\}$

b) $L = \{ w \mid w \text{ starts with an even number of 'a', contains 'ba' and ends with 'baa'} \} \mid \Sigma = \{a,b\}$

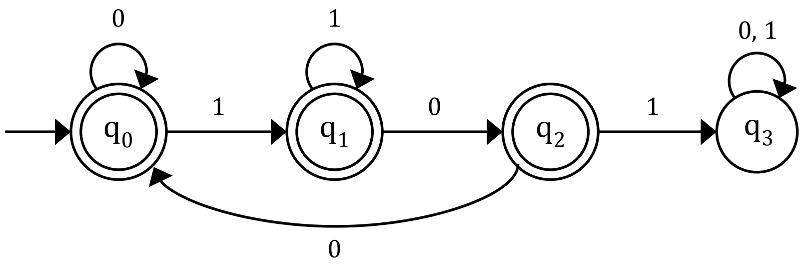
c) $L = \{ w \mid w \text{ is a palindrome with a max length of } 3 \} \mid \Sigma = \{0,1\}$

d) $L = \{ a^i b^j \mid i \geq 0, j \geq 0, \ i + j \text{ is an odd number} \} \mid \Sigma = \{a,b\}$

Solution:

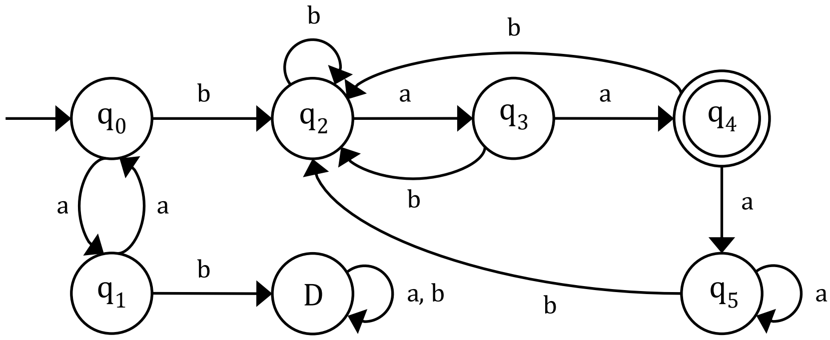
a) $L = \{ w \mid w \text{ does not contain '101'} \} \mid \Sigma = \{0,1\}$

The DFA has been designed below:



b) $L = \{ w \mid w \text{ starts with an even number of 'a', contains 'ba' and ends with 'baa'} \} \mid \Sigma = \{a,b\}$

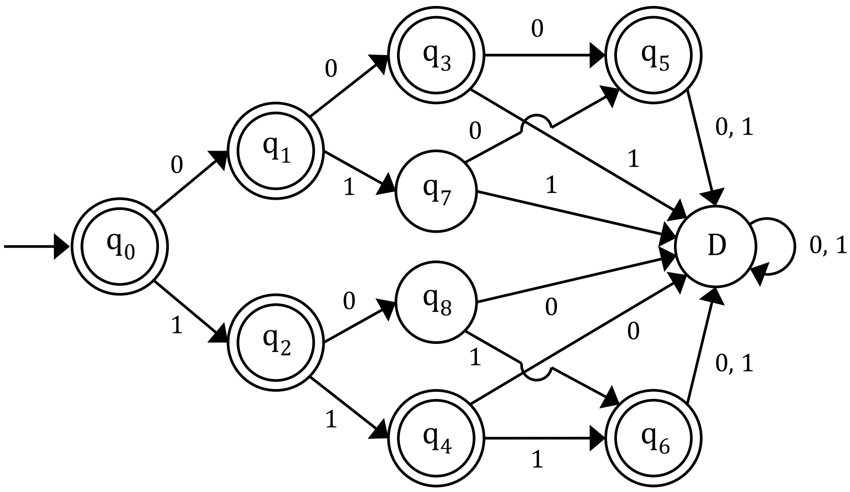
The DFA has been designed below:



c) $L = \{ w \mid w \text{ is a palindrome with a max length of } 3 \} \mid \Sigma = \{0,1\}$

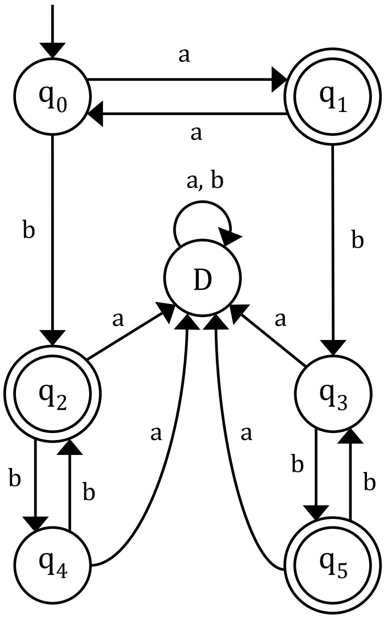
The DFA has been designed below:

[P.T.O]



d) $L = \{\, a^i b^j \mid i \geq 0, j \geq 0,\ i + j \text{ is an odd number} \,\} \mid \Sigma = \{a, b\}$

The DFA has been designed below:



We Know,

Even + Even = Even
Odd + Odd = Even
Even + Odd = Odd

2. Design **NFA**'s that accepts the following languages:

a) $L = \{\, w \mid w$ doesn't start with 'a' or 'b' and contains 'bbc' and ends with 'ab' $\} \mid \Sigma = \{a, b, c\}$

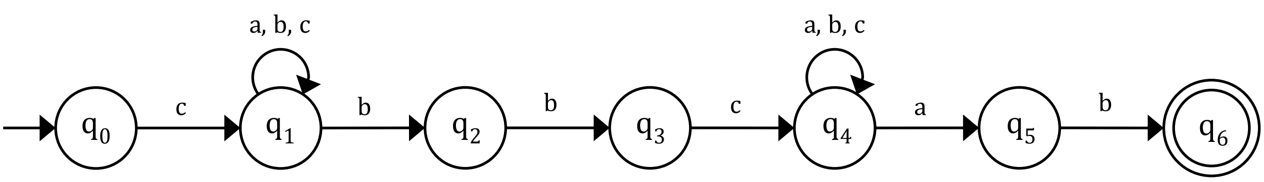
b) $L = \{\, w \mid w$ starts with '10' or '21' and contains '220' and ends with '112' $\} \mid \Sigma = \{0,1,2\}$

c) $L = \{\, w \mid w$ starts and ends with either 'xzy' or 'xy' $\} \mid \Sigma = \{x, y, z\}$

Solution:

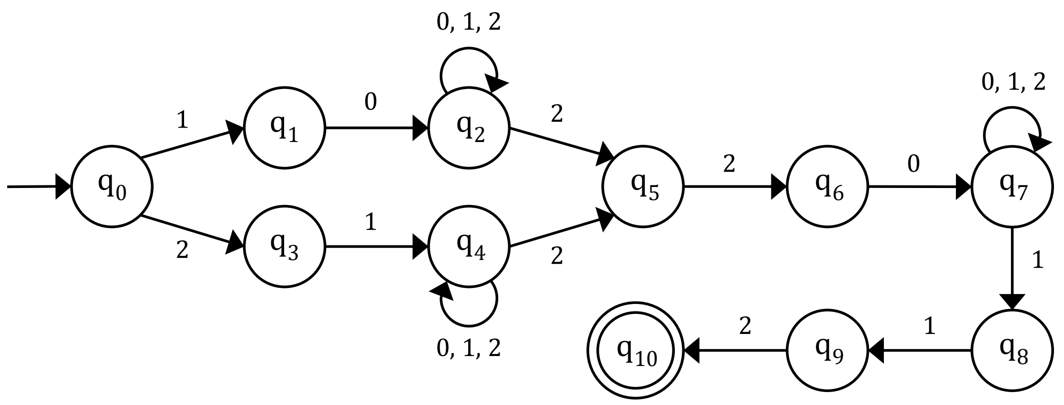
a) $L = \{\, w \mid w$ doesn't start with 'a' or 'b' and contains 'bbc' and ends with 'ab' $\} \mid \Sigma = \{a, b, c\}$

The NFA has been designed below:



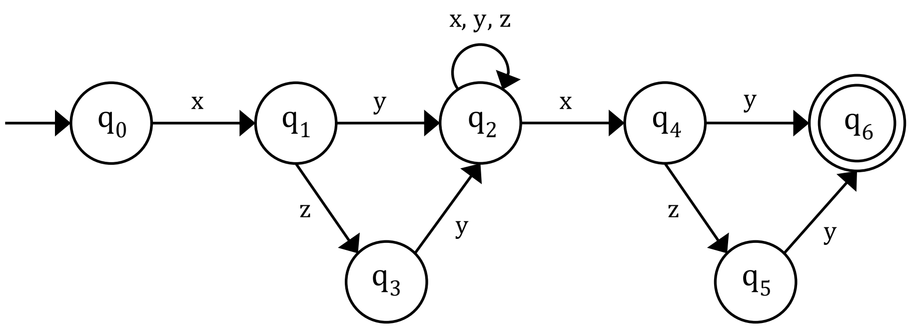
b) L = { w | w starts with '10' or '21' and contains '220' and ends with '112' } | $\sum = \{0,1,2\}$

The NFA has been designed below:

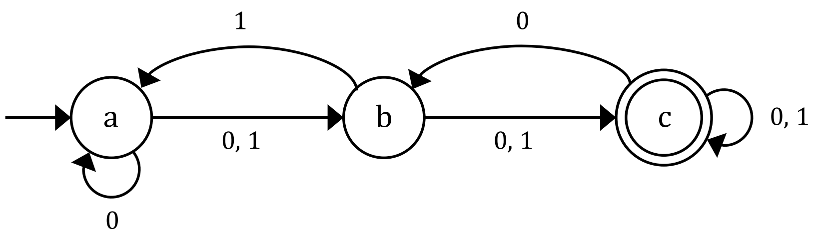


c) L = { w | w starts and ends with either 'xzy' or 'xy' } | $\sum = \{x, y, z\}$

The NFA has been designed below:



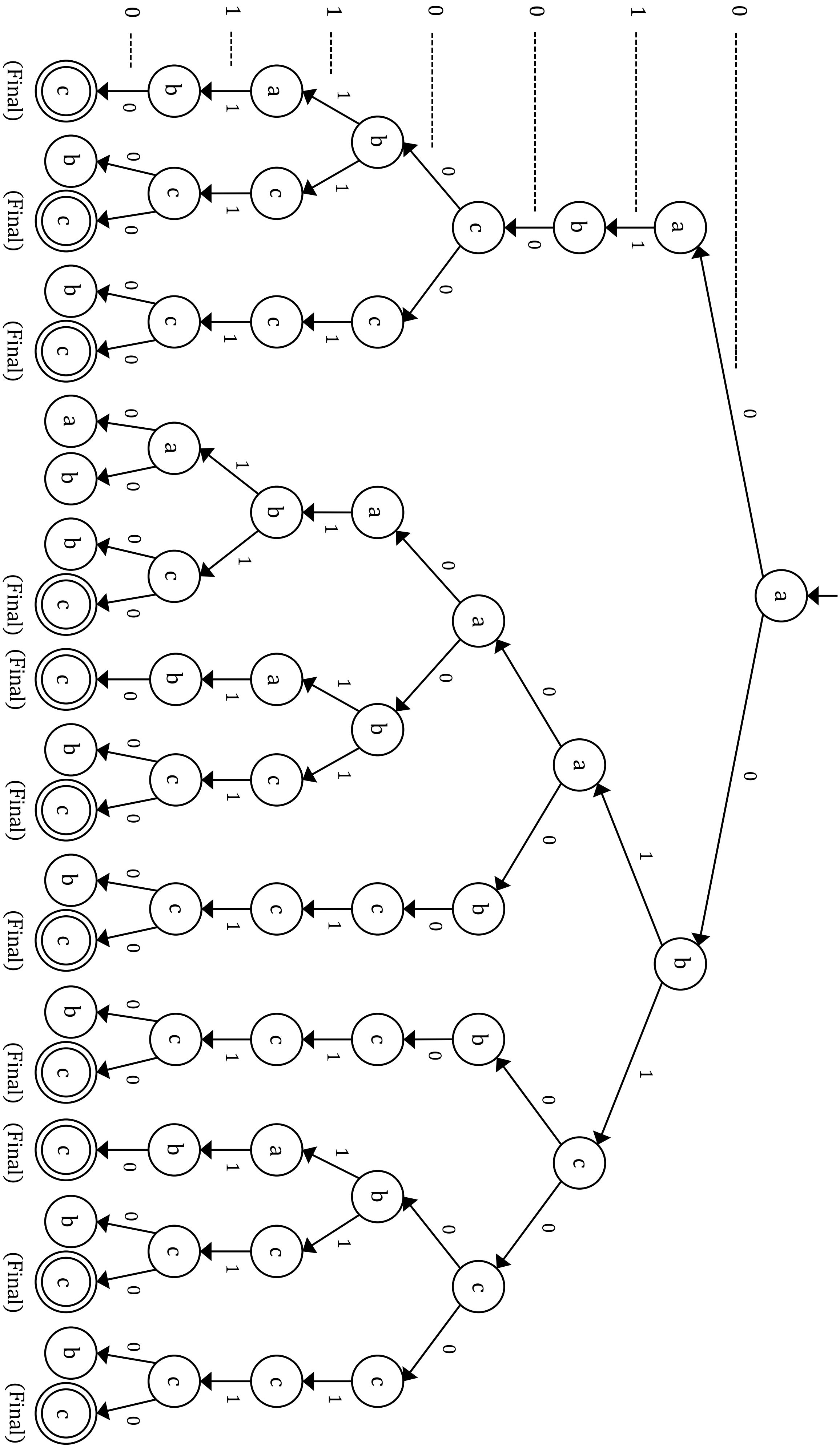
3. Consider the following NFA, and show with help of **NFA-tree** whether the string **"0100110"** is accepted.



Solution:

NFA Tree:

[P.T.O]



With help of NFA-tree, we can see the string "0100110" reach the final state.
∴ The string "0100110" is accepted.

4. Convert the following NFA over the alphabet $\sum = \{1,2,3\}$ to an equivalent DFA including the diagram.

	1	**2**	**3**
→ a	{ a, b, d }	{ a, c }	{ d }
b	∅	{ a, d }	{ a, e }
*** c**	{ a, b, c, d, e }	∅	{ b, c }
d	{ d, e }	{ d }	{ a, e }
*** e**	{ b, e }	∅	∅

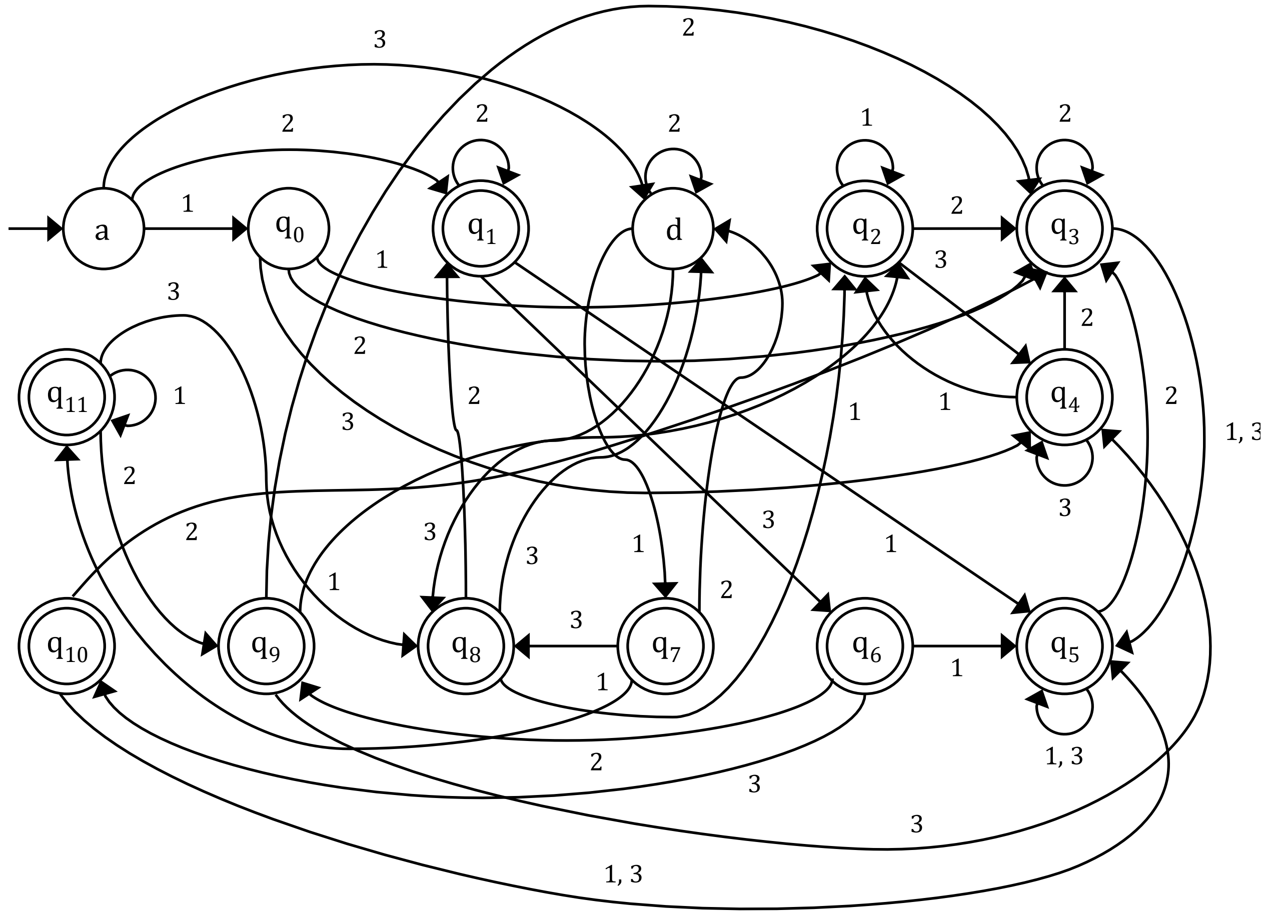
Solution:

Transition Table for the Equivalent DFA:

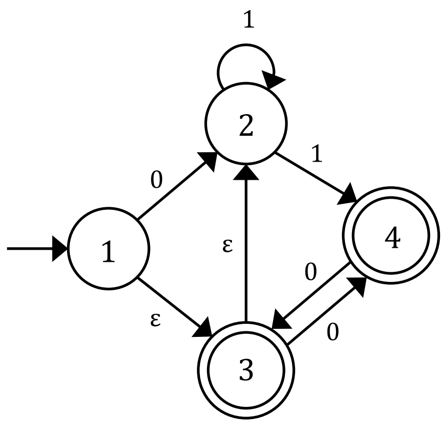
	1	**2**	**3**
→ a	{ a, b, d } = q_0	{ a, c } = q_1	d
q_0 = { a, b, d }	{ a, b, d, e } = q_2	{ a, c, d } = q_3	{ a, d, e } = q_4
*** q_1 = { a, c }**	{ a, b, c, d, e } = q_5	{ a, c } = q_1	{ b, c, d } = q_6
d	{ d, e } = q_7	d	{ a, e } = q_8
*** q_2 = { a, b, d, e }**	{ a, b, d, e } = q_2	{ a, c, d } = q_3	{ a, d, e } = q_4
*** q_3 = { a, c, d }**	{ a, b, c, d, e } = q_5	{ a, c, d } = q_3	{ a, b, c, d, e } = q_5
*** q_4 = { a, d, e }**	{ a, b, d, e } = q_2	{ a, c, d } = q_3	{ a, d, e } = q_4
*** q_5 = { a, b, c, d, e }**	{ a, b, c, d, e } = q_5	{ a, c, d } = q_3	{ a, b, c, d, e } = q_5
*** q_6 = { b, c, d }**	{ a, b, c, d, e } = q_5	{ a, d } = q_9	{ a, b, c, e } = q_{10}
*** q_7 = { d, e }**	{ b, d, e } = q_{11}	d	{ a, e } = q_8
*** q_8 = { a, e }**	{ a, b, d, e } = q_2	{ a, c } = q_1	d
q_9 = { a, d }	{ a, b, d, e } = q_2	{ a, c, d } = q_3	{ a, d, e } = q_4
*** q_{10} = { a, b, c, e }**	{ a, b, c, d, e } = q_5	{ a, c, d } = q_3	{ a, b, c, d, e } = q_5
*** q_{11} = { b, d, e }**	{ b, d, e } = q_{11}	{ a, d } = q_9	{ a, e } = q_8

Equivalent DFA Diagram:

[P.T.O]



5. Convert the following ε-NFA over the alphabet $\sum = \{0,1\}$ to an equivalent DFA.



Solution:

Transition Table of the given ε-NFA:

	0	**1**
→ 1	2	∅
2	∅	2, 4
∗ 3	4	∅
∗ 4	3	∅

ε-Closure of all state of the given ε-NFA:

ε-Closure (1) = { 1, 2, 3 }

ε-Closure (2) = { 2 }

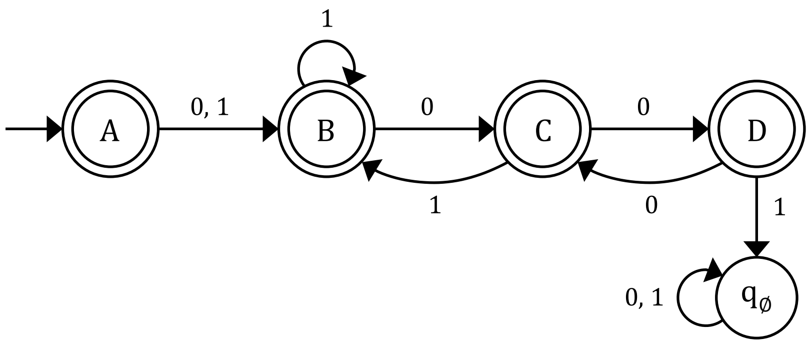
ε-Closure (3) = { 3, 2 }

ε-Closure (4) = { 4 }

Transition Table for the Equivalent DFA:

	0	**1**
→ ∗ A = { 1, 2, 3 }	{ 2, 4 } = B	{ 2, 4 } = B
∗ B = { 2, 4}	{ 2, 3 } = C	{ 2, 4 } = B
∗ C = { 2, 3}	{ 4 } = D	{ 2, 4 } = B
∗ D = { 4}	{ 2, 3 } = C	{ ∅ } = $q_∅$
$q_∅$ = { ∅ }	{ ∅ } = $q_∅$	{ ∅ } = $q_∅$

Equivalent DFA Diagram:



1. Design DFAs that accepts the following languages:

a) L = contains even number of ‘a’ and ends with ‘bc’ | $\sum = \{a, b, c\}$

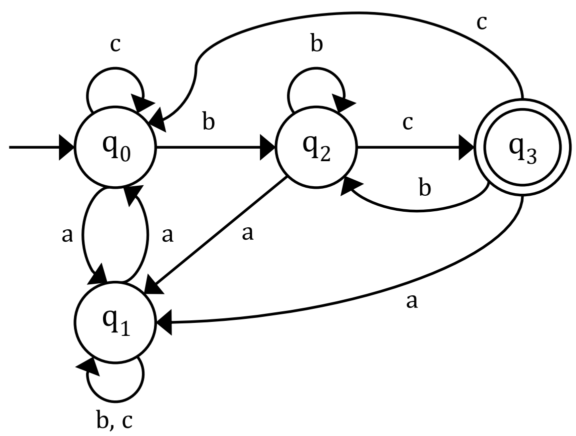
b) L = does not contain ‘mnm’ | $\sum = \{m, n, w\}$

c) L = starts with ‘gh’ and contains ‘kgh’ and ends with ‘gh’ | $\sum = \{g, h, k\}$

Solution:

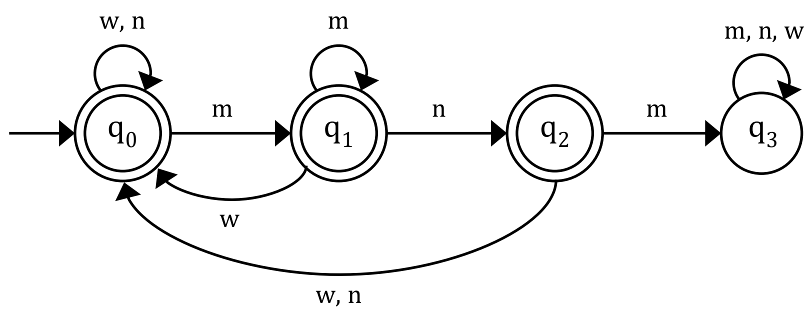
a) L = contains even number of ‘a’ and ends with ‘bc’ | $\sum = \{a, b, c\}$

The DFA has been designed below:



b) L = does not contain ‘mnm’ | $\sum = \{m, n, w\}$

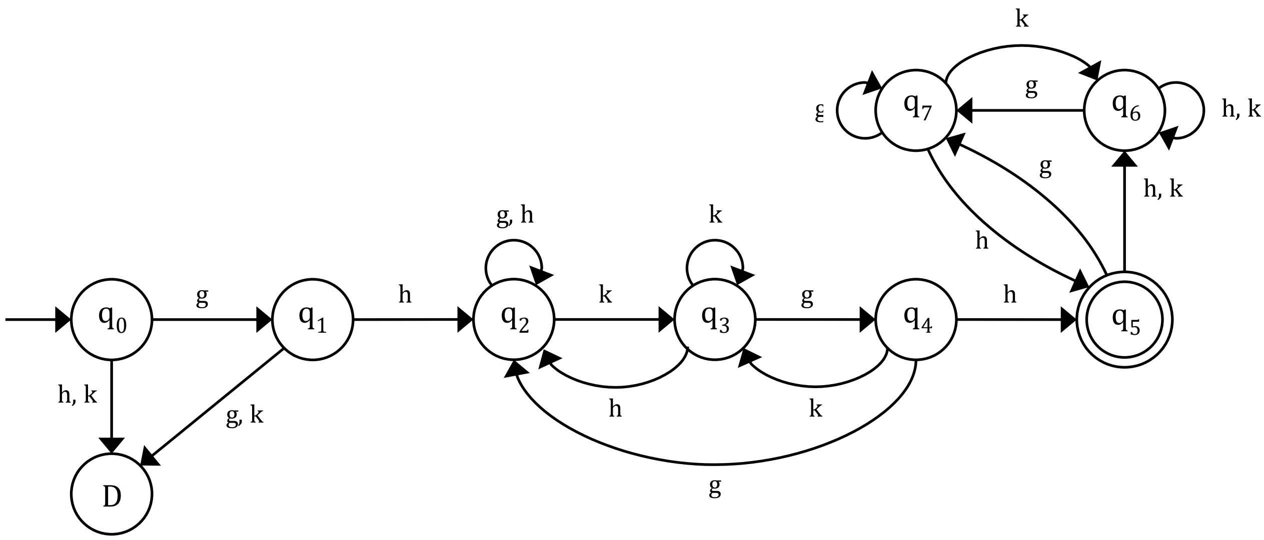
The DFA has been designed below:



c) L = starts with ‘gh’ and contains ‘kgh’ and ends with ‘gh’ | $\sum = \{g, h, k\}$

The DFA has been designed below:

[P.T.O]



2. Design NFAs that accepts the following languages:

a) L = starts with ‘p’, and contains ‘rqp’, and ends with ‘qr’ | $\sum = \{p, q, r\}$

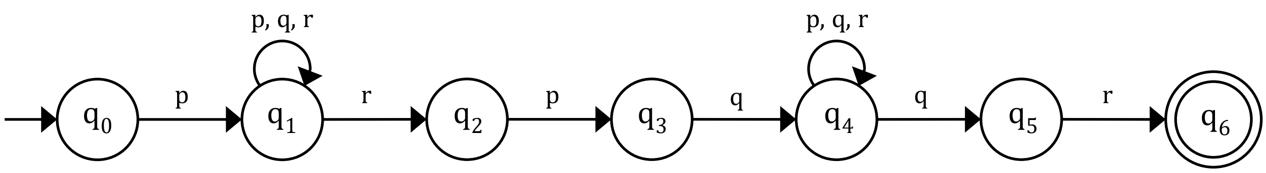
b) L = starts with ‘11’ or ‘21’ and contains ‘210’ and ends with ‘101’ | $\sum = \{0,1,2\}$

c) L = starts with ‘xyz’ and contains ‘yyz’ or ‘zyx’ and ends with ‘zy’ | $\sum = \{x, y, z\}$

Solution:

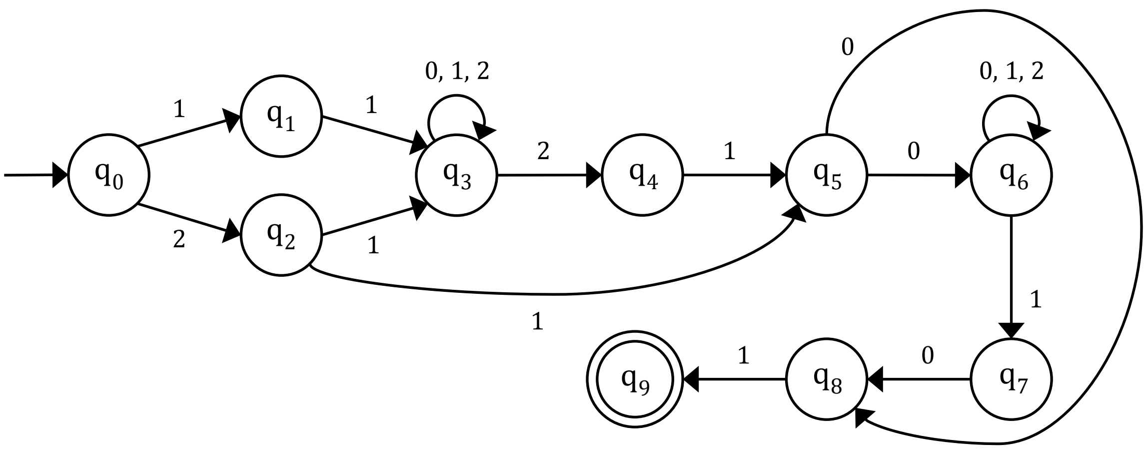
a) L = starts with ‘p’, and contains ‘rqp’, and ends with ‘qr’ | $\sum = \{p, q, r\}$

The NFA has been designed below:



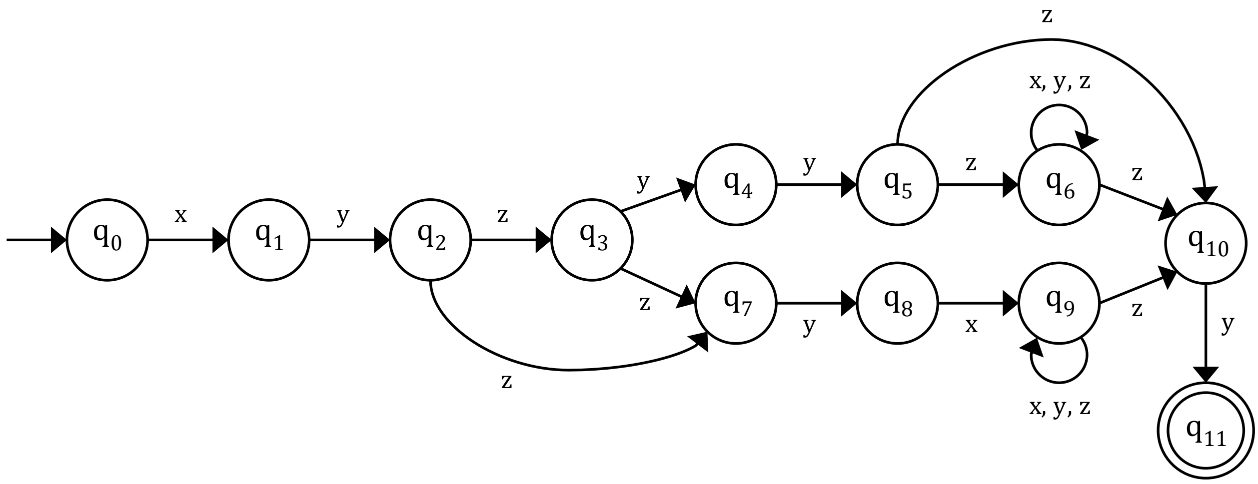
b) L = starts with ‘11’ or ‘21’ and contains ‘210’ and ends with ‘101’ | $\sum = \{0,1,2\}$

The NFA has been designed below:

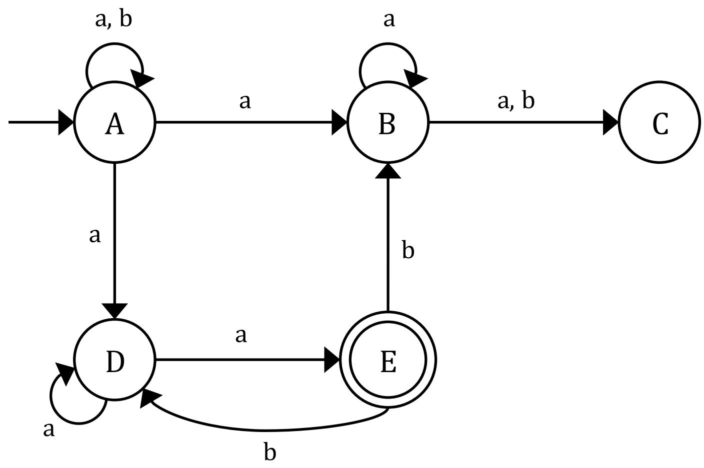


c) L = starts with ‘xyz’ and contains ‘yyz’ or ‘zyx’ and ends with ‘zy’ | $\sum = \{x, y, z\}$

The NFA has been designed below:



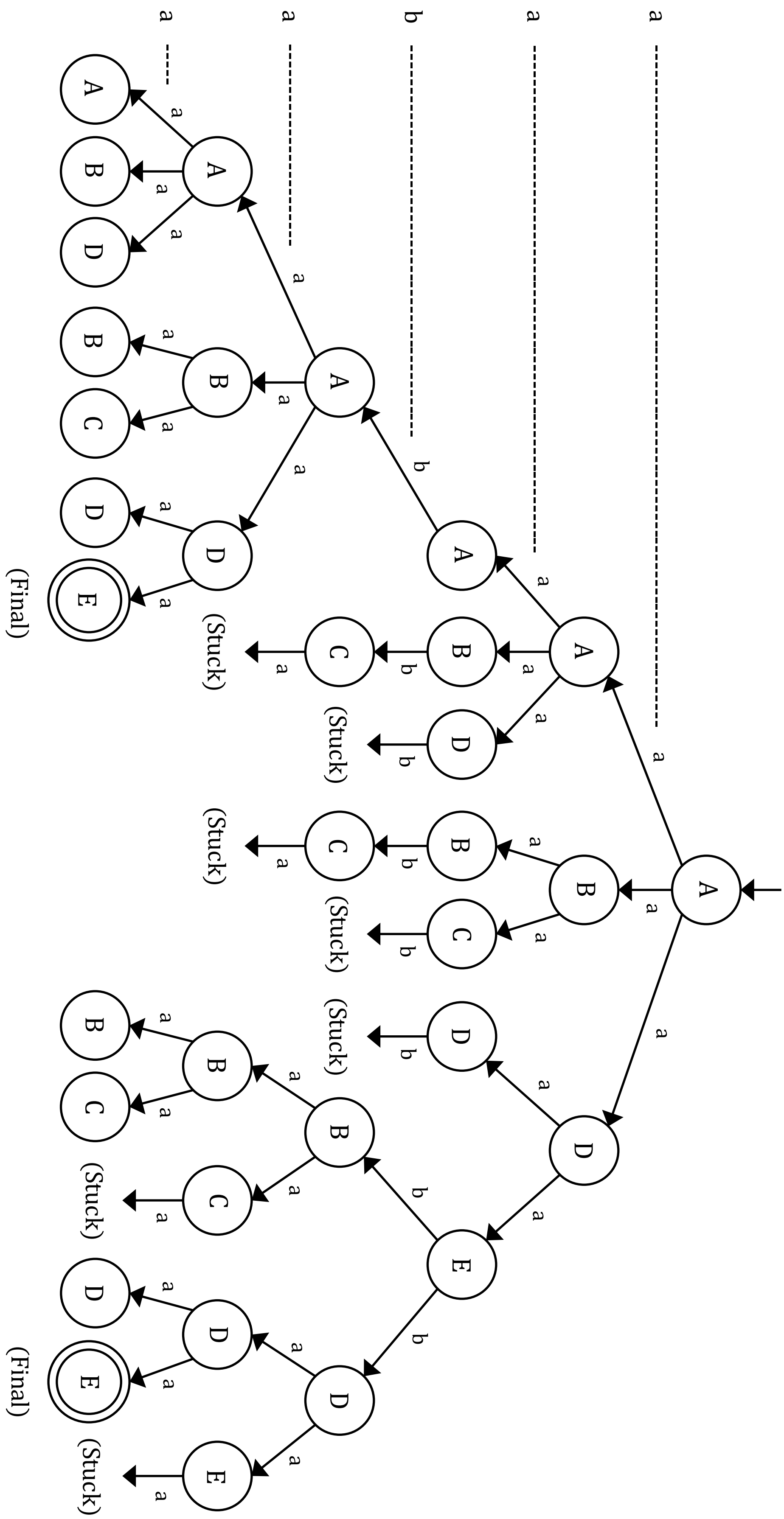
3. Consider the following NFA, and show with help of NFA-tree whether the string "aabaa" is accepted or not.



Solution:

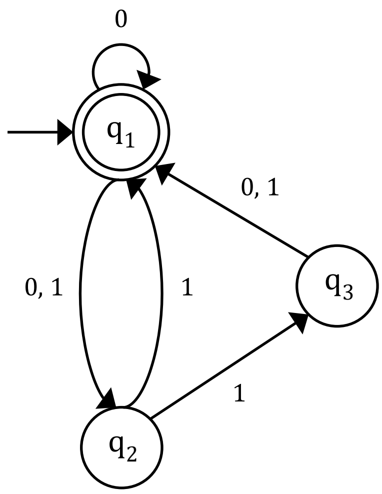
NFA Tree:

[P.T.O]



With help of NFA-tree, we can see the string “aabaa” reach the final state.
∴ The string “aabaa” is accepted.

4. Convert the following **NFA** over the alphabet $\Sigma = \{0,1\}$ to an equivalent **DFA**.



Solution:

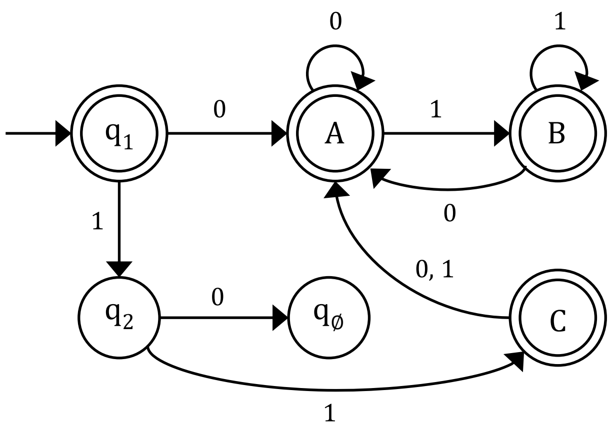
Transition Table of the given NFA:

	0	**1**
→ ∗ q_1	q_1, q_2	q_2
q_2	∅	q_1, q_3
q_3	q_1	q_1

Transition Table for the Equivalent DFA:

	0	**1**
→ ∗ q_1	$\{q_1, q_2\}$ = A	q_2
∗ A = $\{q_1, q_2\}$	$\{q_1, q_2\}$ = A	$\{q_1, q_2, q_3\}$ = B
q_2	{ ∅ } = $q_\emptyset$	$\{q_1, q_3\}$ = C
∗ B = $\{q_1, q_2, q_3\}$	$\{q_1, q_2\}$ = A	$\{q_1, q_2, q_3\}$ = B
∗ C = $\{q_1, q_3\}$	$\{q_1, q_2\}$ = A	$\{q_1, q_2\}$ = A

Equivalent DFA Diagram:



5. Design Regular Expression for the following languages where $\sum = \{a, b\}$

a) All strings $\boldsymbol{w}$ having even length strings and starting with $\mathbf{a}$ or odd length strings starting with $\mathbf{b}$.

b) All strings $\boldsymbol{w}$ which begins and ends with $\mathbf{b}$.

c) All strings $\boldsymbol{w}$ where every $\mathbf{a}$ is followed by at least one $\mathbf{b}$.

Solution:

a) All strings $\boldsymbol{w}$ having even length strings and starting with $\mathbf{a}$ or odd length strings starting with $\mathbf{b}$.

Regular Expression: $(a\,(a \mid b)\ ((a \mid b)\ (a \mid b))^*)^* \ \mid\ b\,((a \mid b)\ (a \mid b))^*$

b) All strings $\boldsymbol{w}$ which begins and ends with $\mathbf{b}$.

Regular Expression: $b\,(a \mid b)^*\,b\ \mid\ b$

c) All strings $\boldsymbol{w}$ where every $\mathbf{a}$ is followed by at least one $\mathbf{b}$.

Regular Expression: $b^*\,(ab\,b^*)^*$

Fall 2022

1. Design DFAs that accepts the following languages:

a) L = { w | w starts with 'ab' and contains 'bba' and ends with 'bb' }
| $\sum = \{a, b\}$

b) L = { w | w contains the set of all strings that has length exactly 3 and its third symbol is from the left side is 'a' }
| $\sum = \{a, b\}$

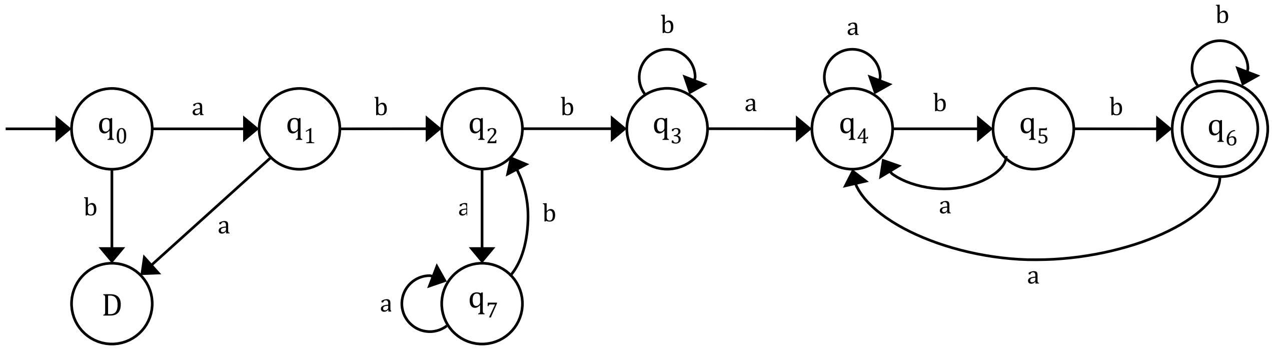
c) L = { w | w contains the set of all strings that has neither '00' nor '11' as substring }
| $\sum = \{0,1,2\}$

d) L = { w | w contains the set of all strings whose length always returns remainder 2 when divided by 4 }
| $\sum = \{0,1\}$

Solution:

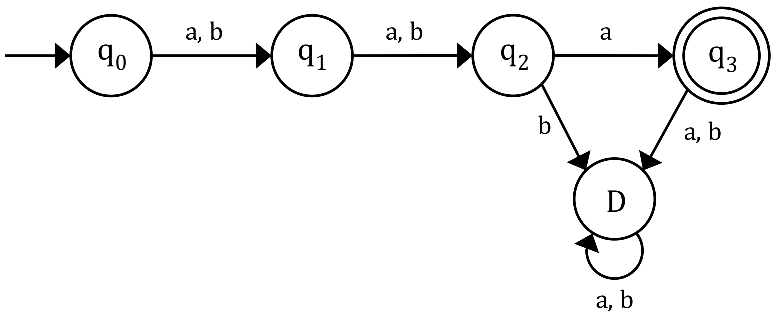
a) L = { w | w starts with 'ab' and contains 'bba' and ends with 'bb' } | $\sum = \{a, b\}$

The DFA has been designed below:



b) L = { w | w contains the set of all strings that has length exactly 3 and its third symbol is from the left side is 'a' } | $\sum = \{a, b\}$

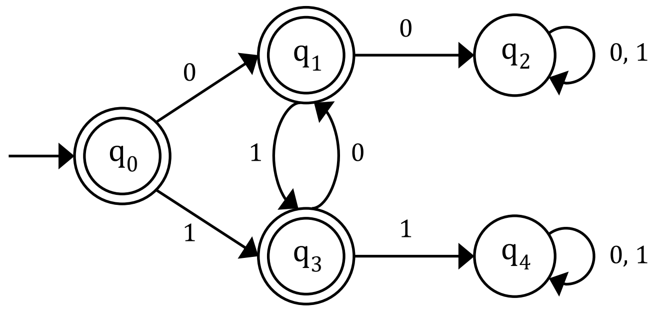
The DFA has been designed below:



c) L = { w | w contains the set of all strings that has neither '00' nor '11' as substring }
| $\sum = \{0,1,2\}$

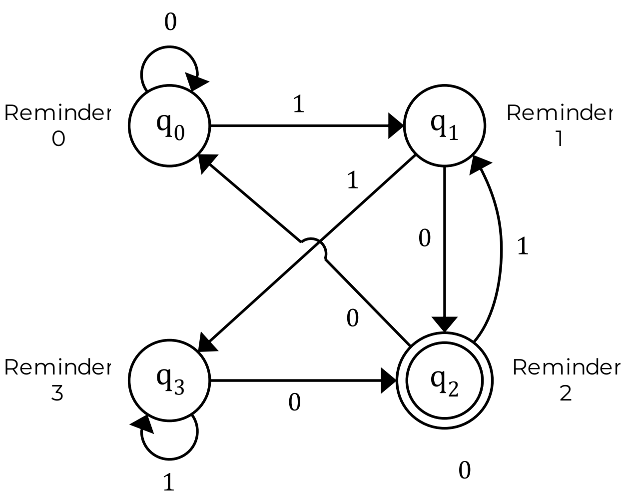
The DFA has been designed below:

[P.T.O]



d) $L = \{ w \mid w$ contains the set of all strings whose length always returns remainder 2 when divided by 4 $\} \mid \Sigma = \{0,1,2\}$

The DFA has been designed below:



Decimal	Binary	Reminder
0	0	0
1	1	1
2	10	2
3	11	3
4	100	0
5	101	1
6	110	2
7	111	3
8	1000	0

2. Design NFAs that accepts the following languages:

a) L = ends with ‘x’ and contains ‘yxz’ and starts with ‘xy’
$\mid \Sigma = \{x, y, z\}$

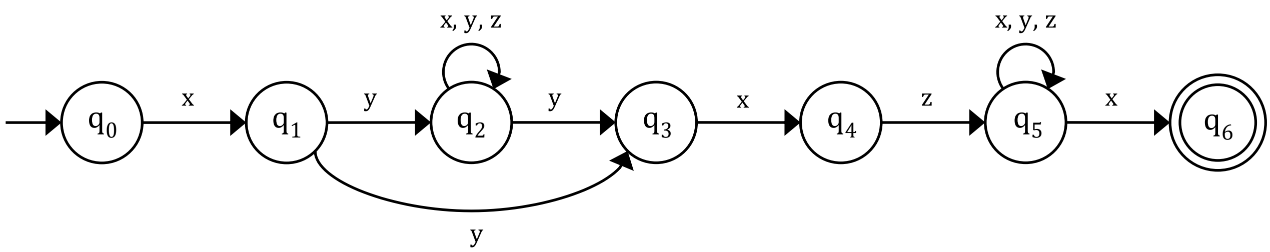
b) L = starts with ‘pq’ or ‘qr’ and contains ‘pqp’ or ‘qrr’ and ends with ‘qqr’
$\mid \Sigma = \{p, q, r\}$

c) L = starts with ‘211’ and contains ‘112’ or ‘321’ and ends with ‘1’
$\mid \Sigma = \{1,2,3\}$

Solution:

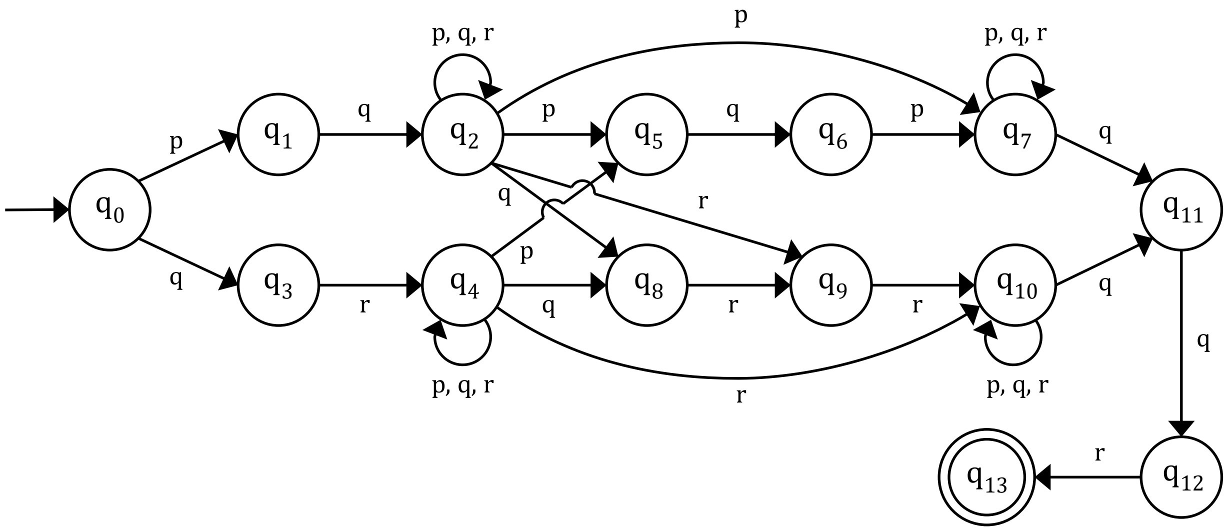
a) L = ends with ‘x’ and contains ‘yxz’ and starts with ‘xy’ $\mid \Sigma = \{x, y, z\}$

The NFA has been designed below:



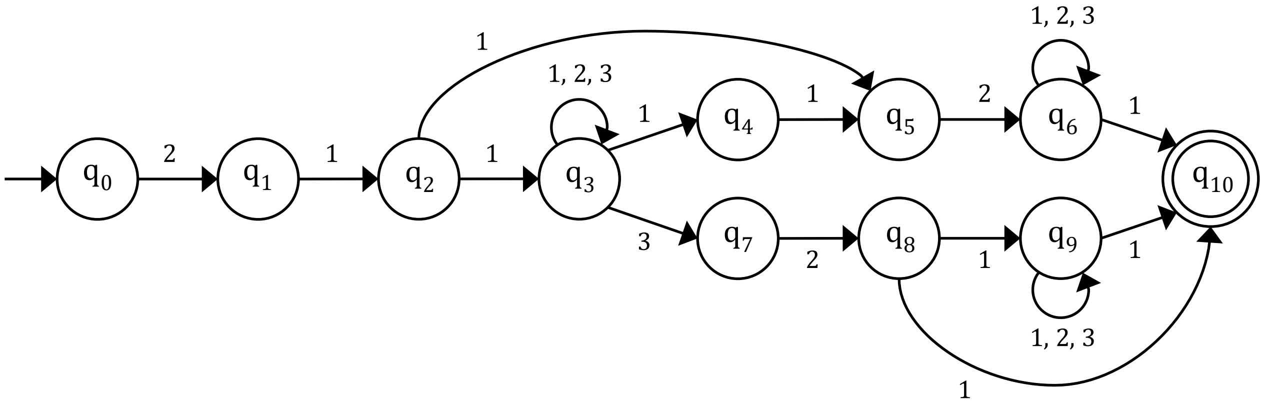
b) L = starts with ‘pq’ or ‘qr’ and contains ‘pqp’ or ‘qrr’ and ends with ‘qqr’ $\mid \Sigma = \{p, q, r\}$

The NFA has been designed below:

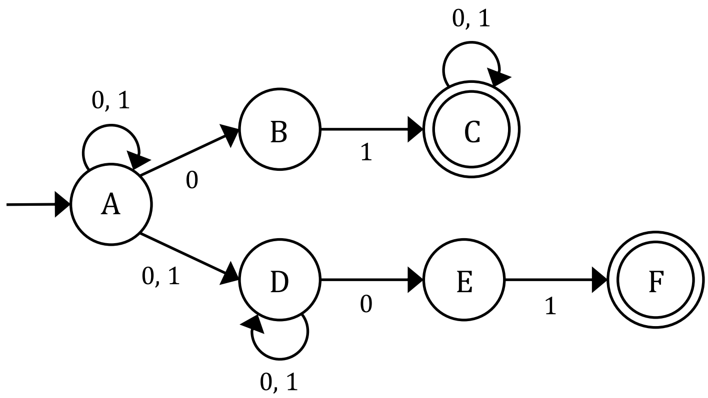


c) L = starts with ‘211’ and contains ‘112’ or ‘321’ and ends with ‘1’ | $\sum = \{1,2,3\}$

The NFA has been designed below:



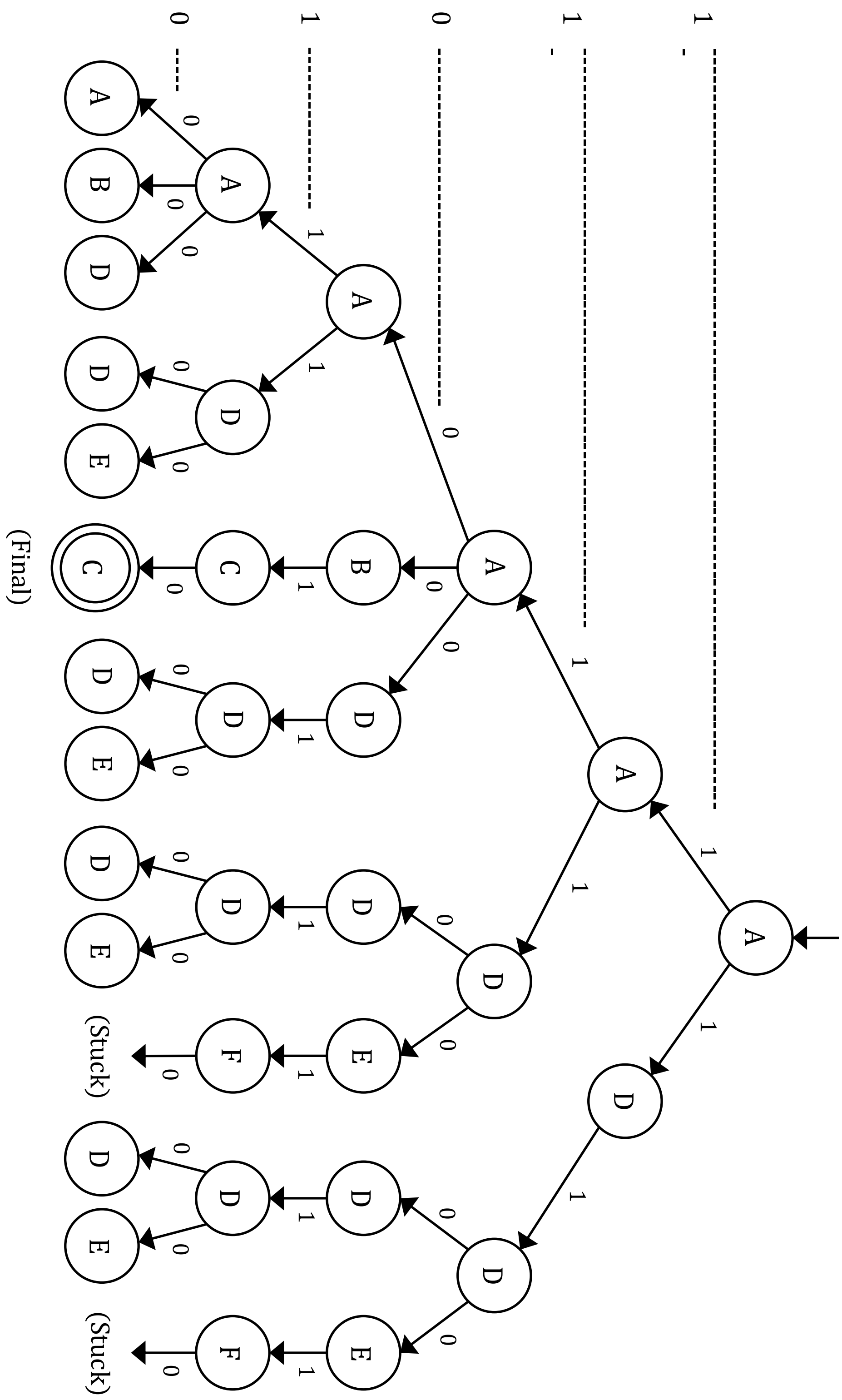
3. Consider the following NFA, and show with help of NFA-tree whether the string “11010” is accepted or not.



Solution:

NFA Tree:

[P.T.O]

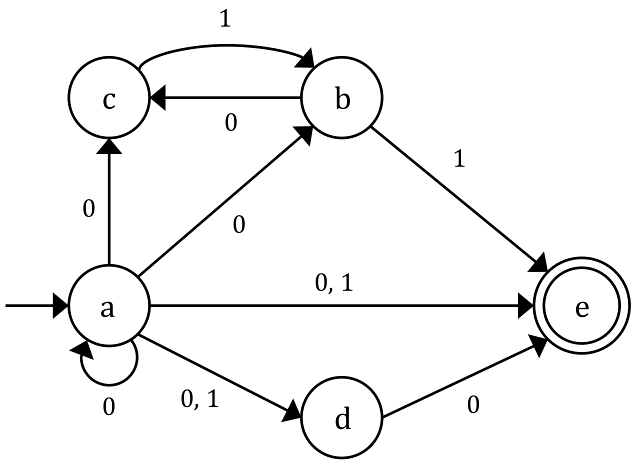


With help of NFA-tree, we can see the string "aabaa" reach the final state.

∴ The string "11010" is accepted.

4. Convert the following NFA over the alphabet $\sum = \{0,1\}$ to an equivalent DFA.

[P.T.O]



Solution:

Transition Table of the given NFA:

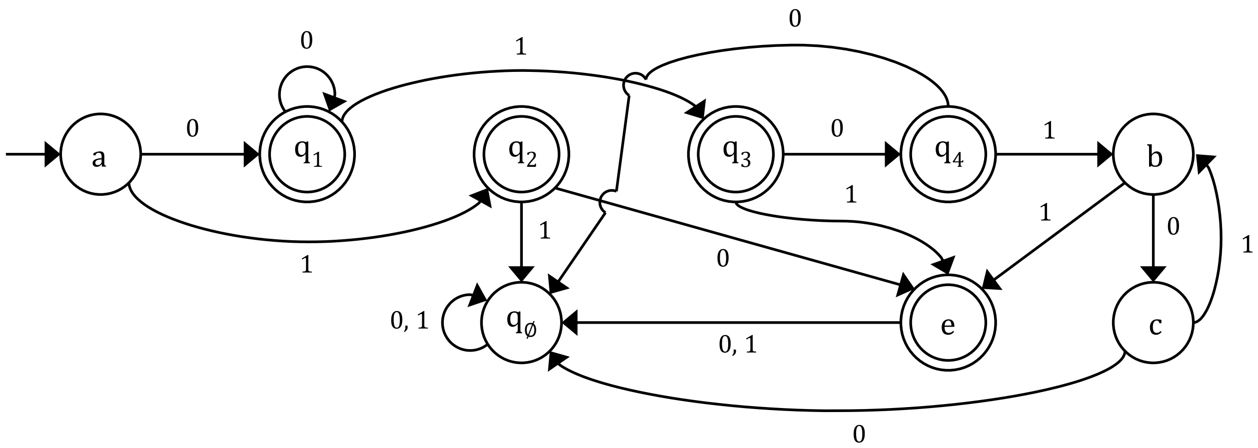
	0	1
→ a	a, b, c, d, e	d, e
b	c	e
c	∅	b
d	e	∅
∗ e	∅	∅

Transition Table for the Equivalent DFA:

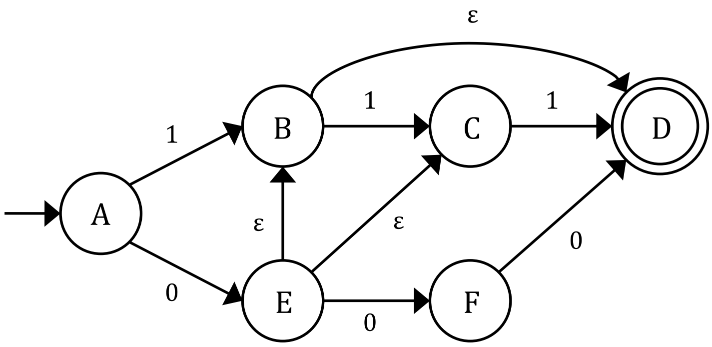
	0	1
→ a	{ a, b, c, d, e } = q_1	{ d, e } = q_2
∗ q_1 = { a, b, c, d, e}	{ a, b, c, d, e } = q_1	{ b, d, e } = q_3
∗ q_2 = { d, e}	e	{ ∅ } = $q_\emptyset$
∗ q_3 = { b, d, e}	{ c, e } = q_4	e
∗ e	{ ∅ } = $q_\emptyset$	{ ∅ } = $q_\emptyset$
∗ q_4 = { c, e}	{ ∅ } = $q_\emptyset$	b
b	c	e
c	{ ∅ } = $q_\emptyset$	b
∗ $q_\emptyset$ = { ∅ }	{ ∅ } = $q_\emptyset$	{ ∅ } = $q_\emptyset$

Equivalent DFA Diagram:

[P.T.O]



5. Convert the following ε-NFA over the alphabet $\sum = \{0,1\}$ to an equivalent DFA.



Solution:

Transition Table of the given ε-NFA:

	0	**1**
→ A	E	B
B	∅	C
C	∅	D
*** D**	∅	∅
E	F	∅
F	D	∅

ε-Closure of all state of the given ε-NFA:

ε-Closure (A) = { A }

ε-Closure (B) = { B, D }

ε-Closure (C) = { C }

ε-Closure (D) = { D }

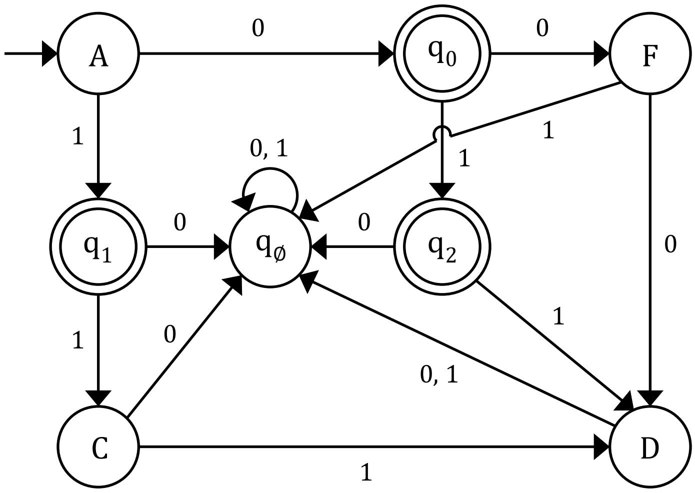
ε-Closure (E) = { E, B, C, D }

ε-Closure (F) = { F }

Transition Table for the Equivalent DFA:

	0	**1**
→ A	{ B, C, D, E } = q_0	{ B, D } = q_1
∗ q_0 = { B, C, D, E}	F	{ C, D } = q_2
∗ q_1 = { B, D}	{ Ø } = $q_\emptyset$	C
F	D	{ Ø } = $q_\emptyset$
∗ q_2 = { C, D}	{ Ø } = $q_\emptyset$	D
C	{ Ø } = $q_\emptyset$	D
∗ D	{ Ø } = $q_\emptyset$	{ Ø } = $q_\emptyset$
$q_\emptyset$ = { Ø }	{ Ø } = $q_\emptyset$	{ Ø } = $q_\emptyset$

Equivalent DFA Diagram:



Summer 2022

1. Design DFAs that accepts the following languages:

a) $L = \{ w \mid w \text{ starts and ends with different symbols and the length of w is even} \} \mid \Sigma = \{0,1\}$

b) $L = \{ w \mid w \text{ contains at least two 'a's and at most one 'b'} \} \mid \Sigma = \{a, b\}$

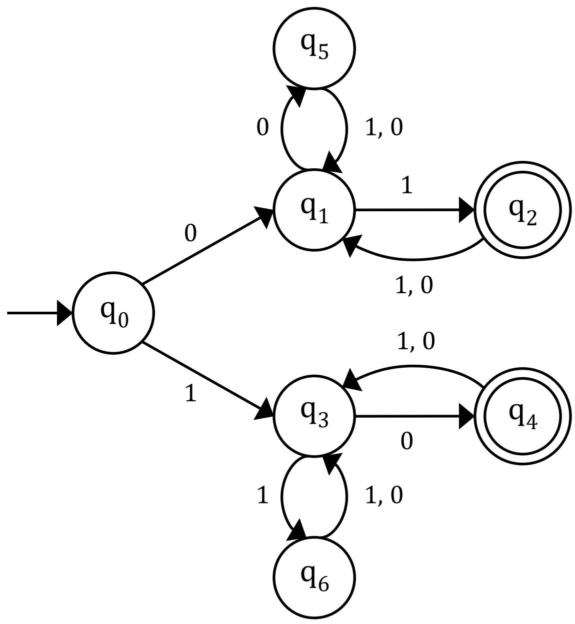
c) $L = \{ w \mid w \text{ contains even number of 0's or odd number of 2's.} \}$ over $\Sigma = \{0,1,2\}$

d) $L = \{ w \mid w \text{ contains all the binary number which is divisible by 3 or ends with with '011'} \}$ $\mid \Sigma = \{0,1\}$

Solution:

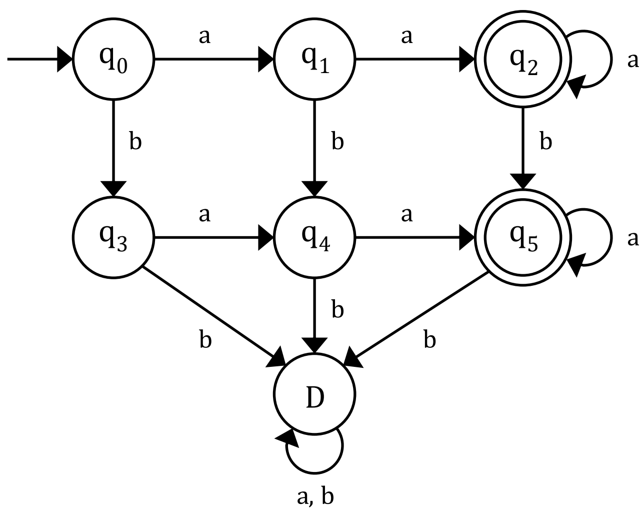
a) $L = \{ w \mid w \text{ starts and ends with different symbols and the length of w is even} \} \mid \Sigma = \{0,1\}$

The DFA has been designed below:



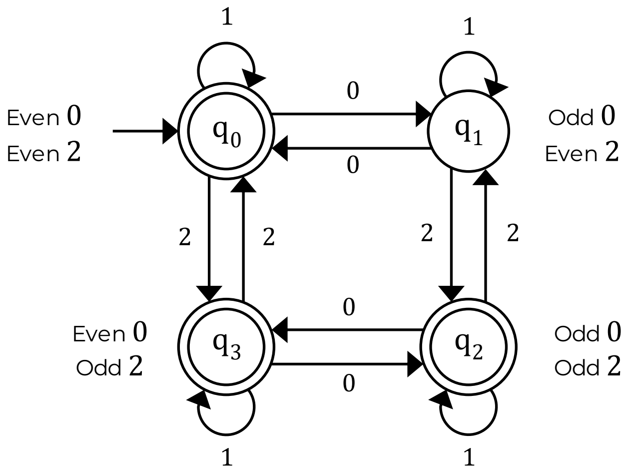
b) $L = \{ w \mid w \text{ contains at least two 'a's and at most one 'b'} \} \mid \Sigma = \{a, b\}$

The DFA has been designed below:



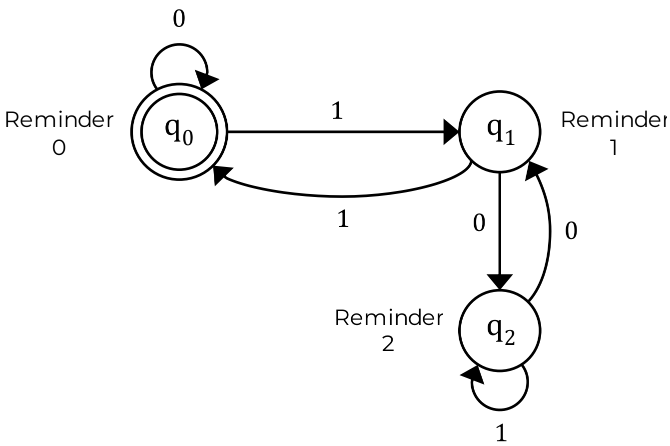
c) $L = \{\ w \mid w \text{ contains even number of 0's or odd number of 2's.}\}$ over $\sum = \{0,1,2\}$

The DFA has been designed below:



d) $L = \{\ w \mid w \text{ contains all the binary number which is divisible by 3 or ends with with '011'}\ \}$

The DFA has been designed below:



Decimal	Binary	Reminder
0	0	0
	1	1
2	10	2
3	11	0
4	100	1
5	101	2
6	110	0

2. Design NFAs that accepts the following languages:

a) $L = \text{ends with 'b' and contains 'bbcb' and starts with 'aacd'} \mid \sum = \{a, b, c, d\}$

b) $L = \text{contains 'bba' or 'abb' or 'acc' and starts with 'ab' or 'bc'} \mid \sum = \{a, b, c\}$

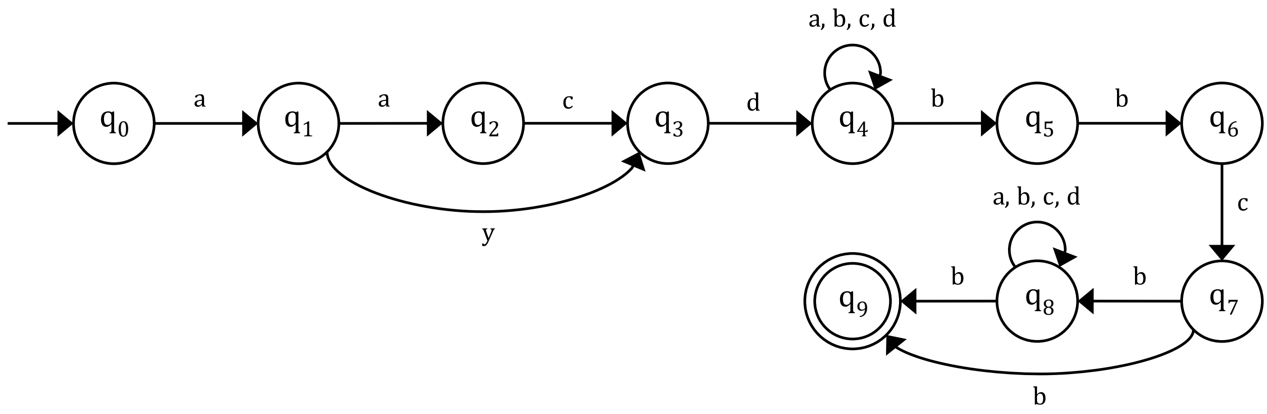
c) $L = \text{starts with '121' and contains '212' or '312' and ends with '2'} \mid \sum = \{1,2,3\}$

Solution:

a) $L = \text{ends with 'b' and contains 'bbcb' and starts with 'aacd'} \mid \sum = \{a, b, c, d\}$

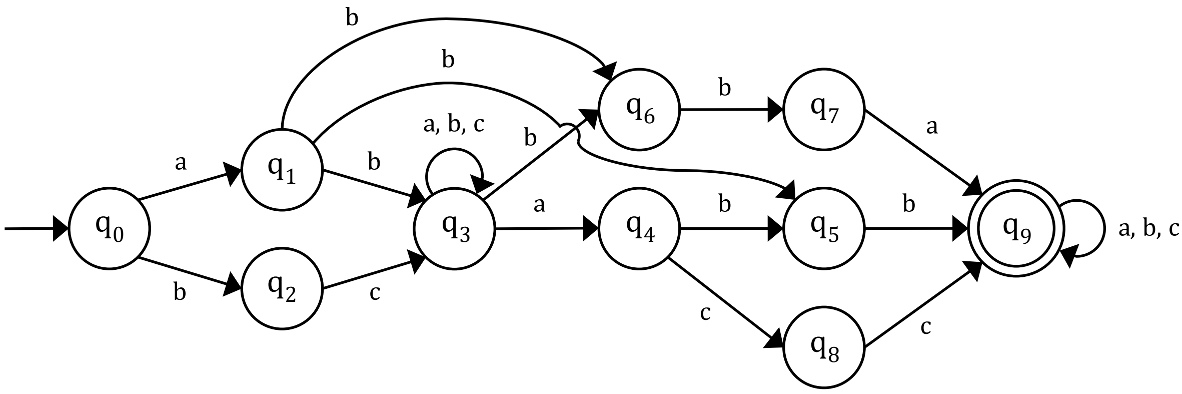
The NFA has been designed below:

[P.T.O]



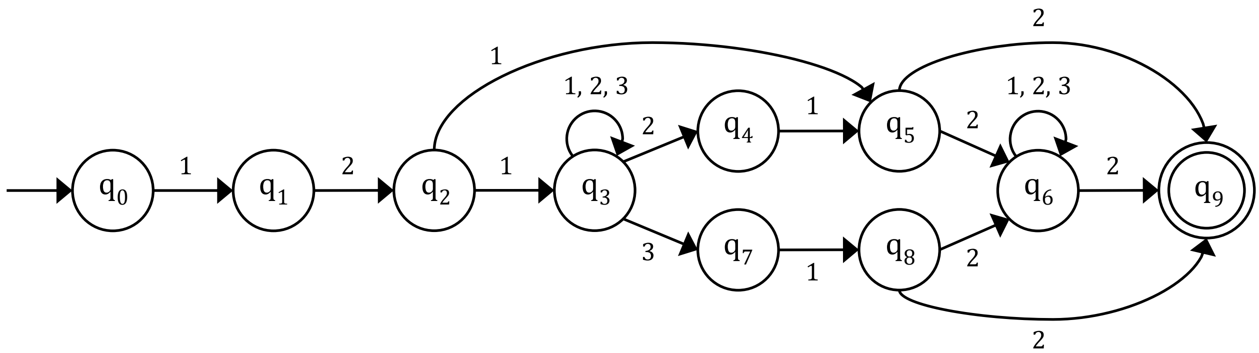
b) L = contains 'bba' or 'abb' or 'acc' and starts with 'ab' or 'bc' | $\Sigma = \{a, b, c\}$

The NFA has been designed below:

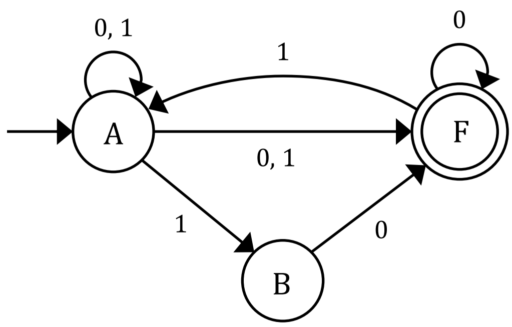


c) L = starts with '121' and contains '212' or '312' and ends with '2' | $\Sigma = \{1,2,3\}$

The NFA has been designed below:

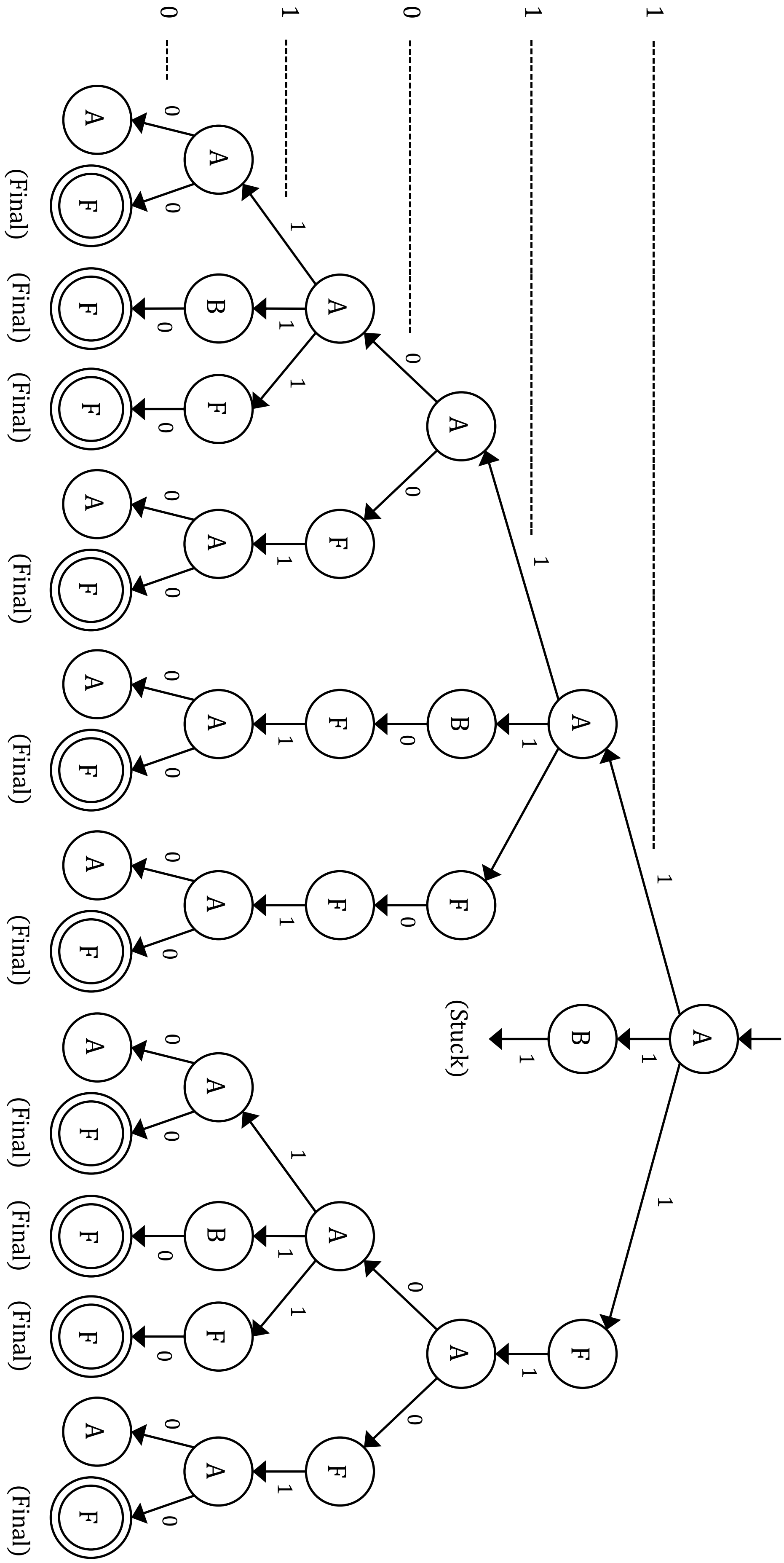


3. Consider the following NFA, and show with help of NFA-tree whether the string "11010" is accepted or not.



Solution:

NFA Tree:



With help of NFA-tree, we can see the string "11010" reach the final state.
∴ The string "11010" is accepted.