AFLL

UNIT - 1

CLASS NOTES

feedback/corrections: vibha@pesu.pes.edu

Vibha Masti

introduction

INTRODUCTION

Course instructor: Prof. Preet Kanwal, preetkanwal@pes.edu
Prof. Kavitha K N

Automata Formal Languages & Logic

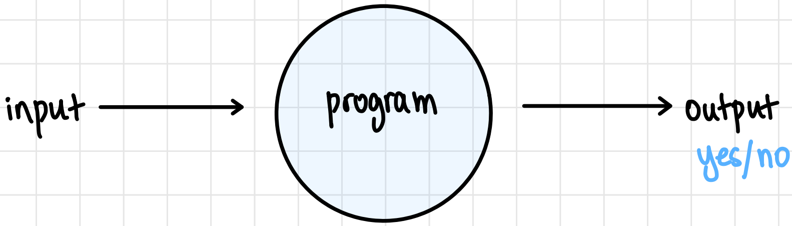
- Automata: abstract/theoretical model of a computer
- Formal Languages: basis of PLs; not natural language

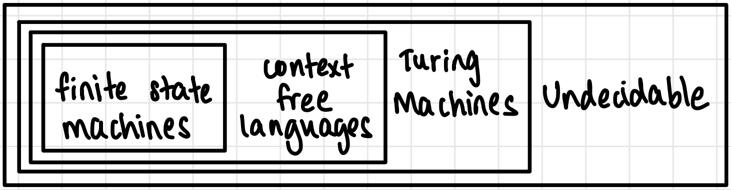
- What can a computer compute & what can't it?
- Theoretical subject

3 Central Areas

1. Complexity theory easy, avg, hard
2. Computability theory solvable, unsolvable (decidable and non decidable)
3. Automata theory theoretical/abstract model

- Turing machine: any task performable of TM can be done on a real computer
- Problems can be decidable or non-decidable; solvable or unsolvable





MATHEMATICAL PRELIMINARIES

Sets

- order of elements does not matter
- { .. , .. , .. }

Set Representation

1. Descriptive form
   English description

   " set of all even numbers "

2. Set builder notation

   rules

   $\{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \}$

   such that/where

3. Roster form

   all elements

   $\{ -1, 0, 1, 2, 3, 4, 5 \}$

Order of a Set / Cardinality

- Number of elements in the set
- What is $|\mathbb{N}| = ?$ (infinite)

$$|S| = |\mathbb{N}| = \aleph_0$$

TYPES OF SETS

Empty Set

- Empty set is represented by $\phi = \{\}$
- No elements
- Note: $\phi \neq \{\phi\}$

Singleton Set

- Single element
- eg: $\{2\}$, $\{4\}$, $\{\{1,2,3,4\}\}$

Finite Set

- finite no. of elements

Infinite Set

- infinite no. of elements
- $S = N$

Equivalent Sets

- same no. of elements
- $A = \{1,2,3,4\}$ and $B = \{13,14,15,14\}$

Equal Sets

- Same elements
- $A = \{1,2,3,4\}$ and $B = \{2,3,1,4\}$

Disjoint Sets

- No common elements

Subsets

- Proper: not equal sets
- $A = \{1,2,3\}$, $B = \{2,3\} \Rightarrow B \subset A$ proper
- $A \subseteq A$ (improper subset)

Superset

- $A = \{1, 2, 3\}$, $B = \{2, 3\}$
- $A \supset B$ and $A \supseteq A$

Universal set

- all sets

Power Set

- set of all subsets (proper & improper)

SET OPERATIONS & IDENTITIES

- refer notes

Functions & Relations

- $f(x) = 3x+2 \longrightarrow$ f is a function
- $y = f(x) \longrightarrow$ y is a function f of x
- f maps x to y

Types of functions

1. One-to-one / injective function

   each element of domain ⟶ one element in codomain

2. Onto / surjective

   all elements in codomain have a pre image

3. Bijective

   both 1-1 and onto

Relation

- A binary relation b/w two sets is a subset of cartesian product of two sets.
- Let A and B be sets ; element $a \in A$ & $b \in B$
- a is related to b as aRb
- eg: $A = \{0,1,2\}$; $B = \{x, y\}$

  $A \times B = \{(0,x), (0,y), (1,x), (1,y), (2,x), (2,y)\}$

  $R = \{(0,x), (1,x), (2,x)\}$

  $R \subset A \times B$

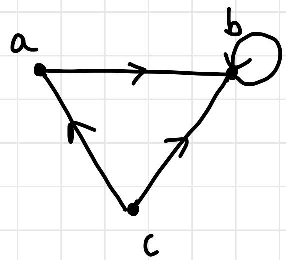
Representing a Relation:

Let $R \subset A \times A$ where $A = \{a, b, c\}$

1) Matrix form

	a	b	c
a	0	1	0
b	0	1	0
c	1	1	0

2) Directed Graph (Digraph)



Properties of Relation

1) Reflexive

iff $(a,a) \in R$ for $a \in A$

$\{(1,1), (2,2), (3,3), (4,4), (1,2)\}$ ✓

2) Symmetric

iff $(b,a) \in R$ and $(a,b) \in R$

3) Transitive

iff $(a,b) \in R$ and $(b,c) \in R$, then $(a,c) \in R$

4) Equivalence Relation

all 3

Equivalence Class

- Set of all elements related to element $a \in A$ is called $[a]_R$ or $[a]$

FINITE STATE MACHINES

- Acceptors — this course
- Recognisers — Moore & Meely machine

Basic Notation

1) Alphabet — Σ
   - finite set of symbols
   - binary : $\{0,1\}$
   - English : $\{a, b, c \ldots z\}$
   - ASCII : all ASCII

2) String — w
   - finite sequence of symbols
   - empty string — $\{\epsilon\}$ or $\{\lambda\}$

3) Length of string — $|w|$
   - $|\{\epsilon\}| = 0$

4) Power of an alphabet — Σ^i
   - set of strings of length i

   - if $\Sigma = \{0,1\}$

     $\Sigma^0 = \{\lambda\}$ set of strings with length 0
     $\Sigma^1 = \{0, 1\}$
     $\Sigma^2 = \{00, 01, 10, 11\}$

5) Kleene Closure / Kleene Star — Σ^*
   - set of strings of length >= 0 (universe)

   - $\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \ldots \infty$

6) Kleene Plus — Σ^+

IPEVO

- set of strings of length > 0

7) Language — L

- set of strings obtained from Σ^*
- $L \subseteq \Sigma^*$
- There are infinite languages from Σ^*
- Languages can be finite or infinite

finite representation OR finite machine

cannot enumerate all strings, but can still write a programe

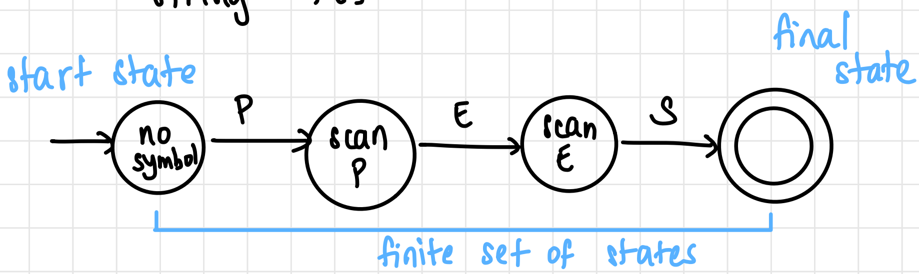
Acceptors / State Machines (recogniser — yes/no)

- Deterministic Finite Acceptor
- Non-deterministic Finite Acceptor
- λ-Non-deterministic Finite Acceptor

Deterministic Finite Acceptor

- Eg: find 's' in string

  string = "PES"



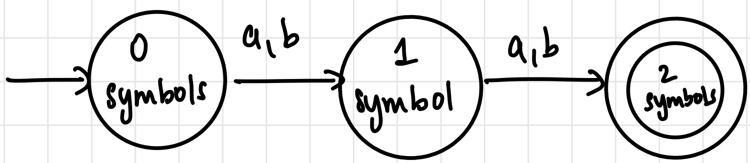
- Define a language & construct DSA

  $\mathcal{L} = \{ w : |w| = 2, \; w \in \{a, b\}^* \}$

  $\mathcal{L} = \{ aa, ab, ba, bb \}$

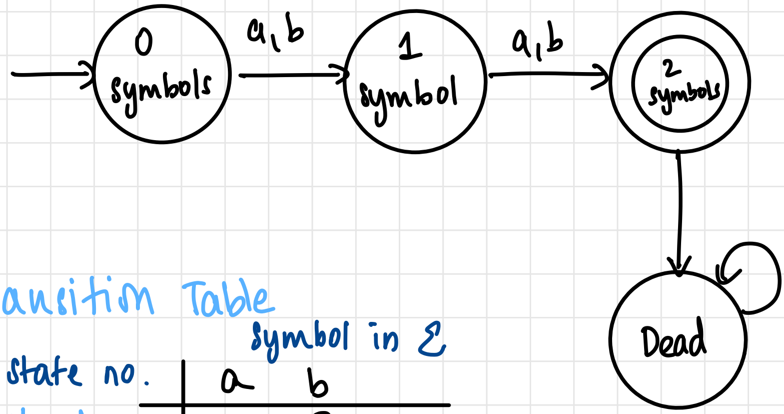
- Construct a machine that accepts words in this language

- Note: SINGLE TRANSITION, finite set of states



- This machine is not deterministic; it does not account for "aba"

1. Transition Diagram



2. Transition Table

state no.	Symbol in Σ: a	b
start → A	B	B
B	C	C
final * C	D	D
D	D	D

jolly ride / self loop

3. Description of Machine using 5 tuples

- DFA, NFA and λ-NFA are 5-tuple machines

$M = \{Q, \Sigma, q_0, F, \delta\}$

Q = set of states in machine M for language L
= $\{A, B, C, D\}$

$\Sigma = \{a, b\}$

q_0 = start state = $A \in Q$

F = set of final states = $\{C\}$

δ = trancition function → differentiating factor b/w 3 acceptors

$\delta: \; Q \times \Sigma \rightarrow Q$ ← a single state

state in Q symbol in Σ goes to which state

- For this machine,

$\delta(A,a) = B \qquad \delta(A,b) = B$

$\delta(B,a) = C \qquad \delta(B,b) = C$

$\delta(C,a) = D \qquad \delta(C,b) = D$

$\delta(D,a) = D \qquad \delta(D,b) = D$

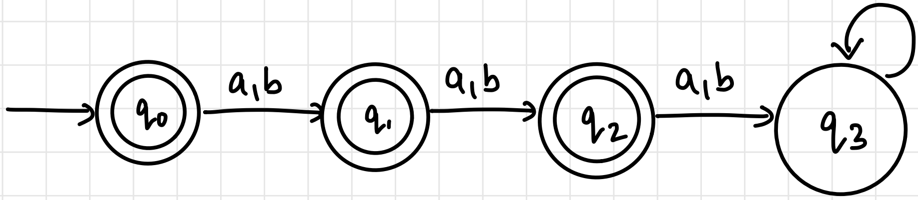
Question 1

$\mathcal{L} = \{ w : |w| \leq 2, \; w \in \Sigma^* \text{ where } \Sigma = \{a,b\}\}$

Create a DFA for this language

$\mathcal{L} = \{ \varepsilon, a, b, aa, ab, ba, bb \}$

Transition Diagram



- If string lands on a non-final state (single circle), the string is rejected

Transition Table

	a	b
$*q_0$	q_1	q_1
$*q_1$	q_2	q_2
$*q_2$	q_3	q_3
q_3	q_3	q_3

Description

$M = \{Q, \Sigma, q_0, F, \delta\}$

$Q = \{q_0, q_1, q_2, q_3\}$

$\Sigma = \{a, b\}$

$q_0 = \{q_0\}$

$F = \{q_0, q_1, q_2\}$

$\delta(q_0, a) = q_1$

$\delta(q_0, b) = q_1$

$\delta(q_1, a) = q_2$

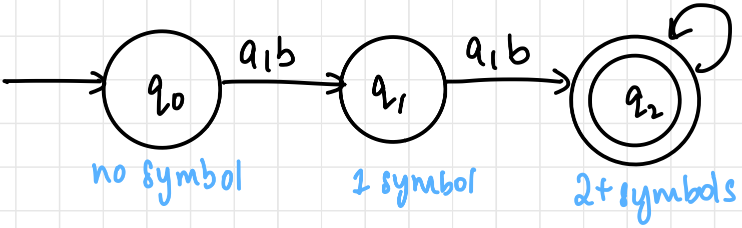
$\delta(q_1, b) = q_2$

$\vdots$

Question 2

$L = \{ w : |w| \geq 2,\ w \in \{a,b\}^* \}$

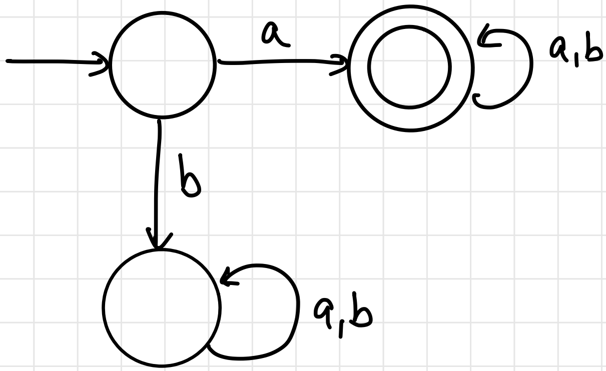
$L = \{ aa, ab, ba, bb, aaa \dots \}$ infinite



(not the only valid DFA, but only 1 unique minimal state)

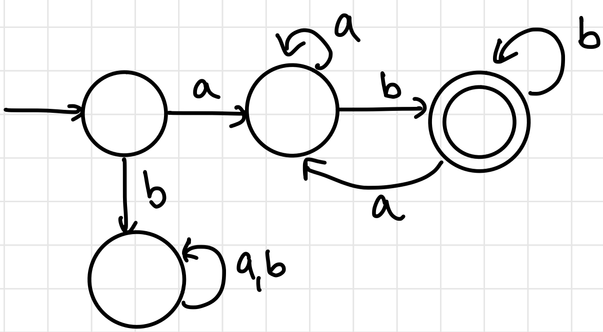
Question 3

$\mathcal{L} = \{ aw \mid w \in \{a,b\}^* \}$ (start with a)



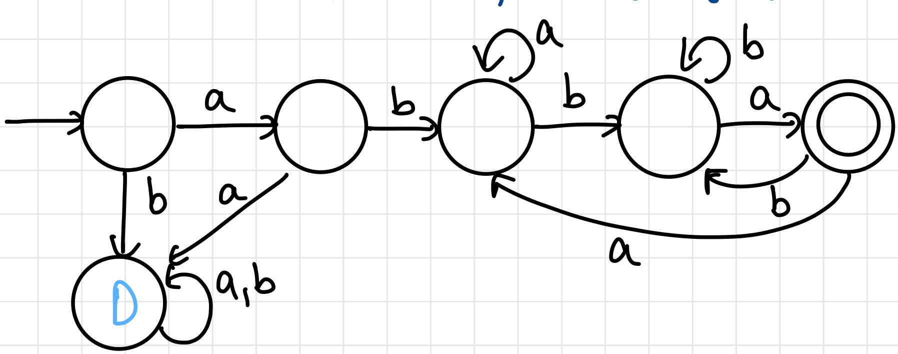
* Question 4

$\mathcal{L} = \{ w \mid aw_0b,\ w_0 \in \{a,b\}^* \}$



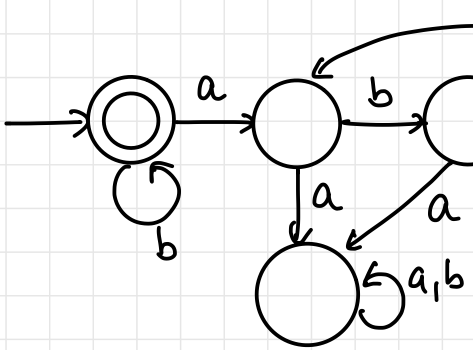
Question 5

$\mathcal{L} = \{ w \mid abwba,\ w \in \{a,b\}^* \}$



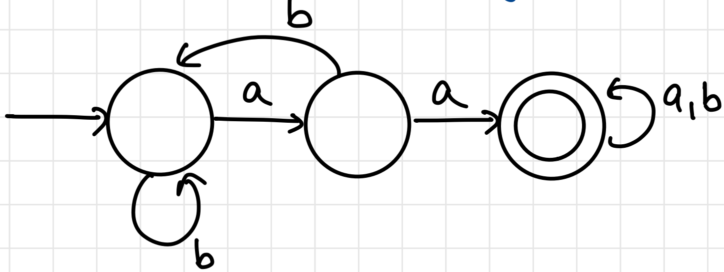
Question 6

$L = \{$ every 'a' followed by a 'bb' $\}$



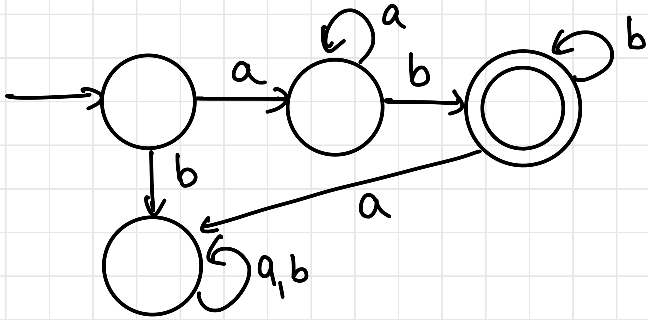
Question 7

$L = \{$ every string contains 'aa' as a substring, $w \in \{a,b\}^* \}$



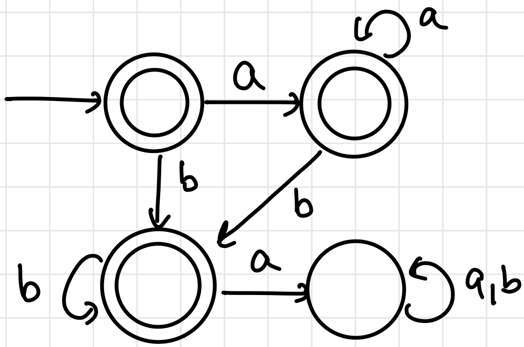
Question 8

$L = \{ a^n b^m \mid n,m \geq 1 \}$



Question 9

$L = \{a^n b^m \mid n,m \geq 0\}$



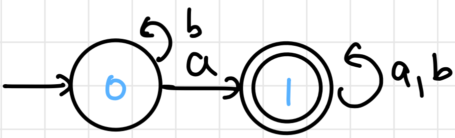
Question 10

$L = \{$ at least one a **and** one b $\}$

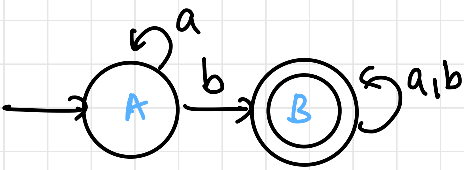
final state of both

[if or, all final would be final states]

Construct 2 DFAs and take cross product



$Q_1 = \{0,1\}$



$Q_2 = \{A,B\}$

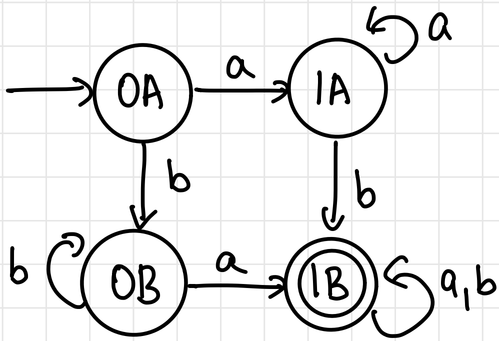
$Q = Q_1 \times Q_2 = \{ (0A), (0B), (1A), (1B) \}$

$(0A) \xrightarrow{a} (1A)$, $(0A) \xrightarrow{b} (0B)$

$(0B) \xrightarrow{a} (1B)$, $(0B) \xrightarrow{b} (0B)$

$(1A) \xrightarrow{a} (1A)$, $(1A) \xrightarrow{b} (1B)$

$(1B) \xrightarrow{a} (1B)$, $(1B) \xrightarrow{b} (1B)$

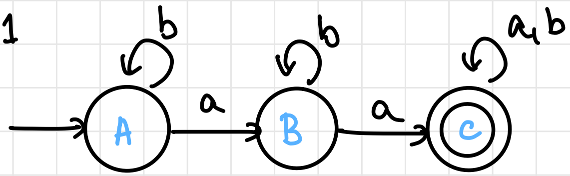


Question 11

L = at least 2 a's & ends with even no. of a's

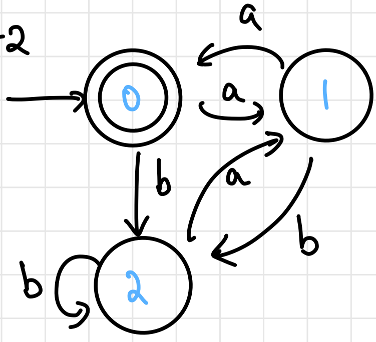
×

DFA-1



$Q_1 = \{A, B, C\}$

DFA-2



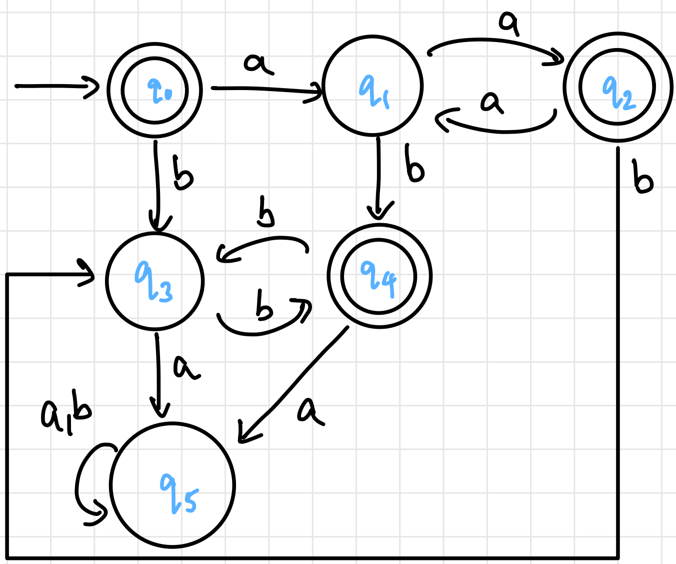
$Q_2 = \{1, 2, 3\}$

do cross product

Question 12

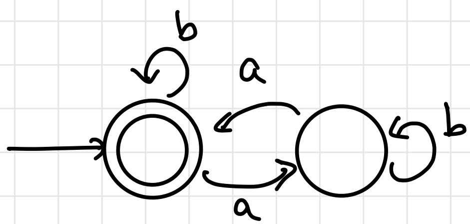
$$L = \{a^n b^m \mid (n+m) \bmod 2 = 0,\ n,m \geq 0\}$$

$L = \{ \lambda, aa, ab, bb, aaab, aabb, abbb, aaaa \ldots \}$



Question 13

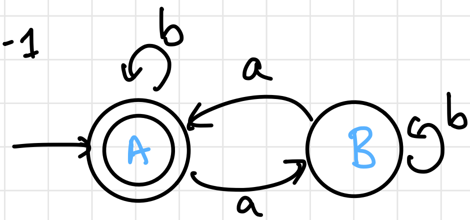
$$L = \{ n_a(w) \bmod 2 = 0,\ w \in \{a,b\}^* \}$$



Question 14

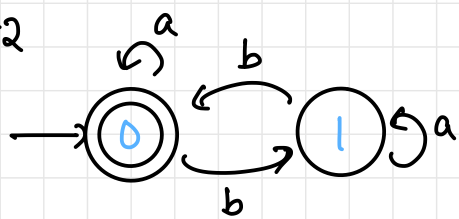
$$L = \{ n_a(w) \bmod 2 = 0 \text{ and } n_b(w) \bmod 2 = 0 \}$$
$$w \in \{a,b\}^*$$

DFA-1



$Q_1 = \{A, B\}$

DFA-2



$Q_2 = \{0, 1\}$

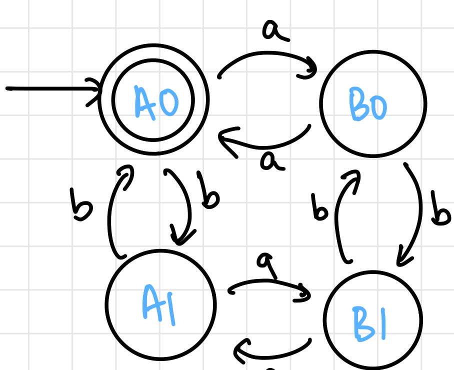
$Q = Q_1 \times Q_2 = \{ A0, A1, B0, B1 \}$

$A0 \xrightarrow{a} (B0)$, $A0 \xrightarrow{b} (A1)$

$A1 \xrightarrow{a} (B1)$, $A1 \xrightarrow{b} (A0)$

$B0 \xrightarrow{a} (A0)$, $B0 \xrightarrow{b} (B1)$

$B1 \xrightarrow{a} (A1)$, $B1 \xrightarrow{b} (B0)$



WITHOUT USING CROSS PRODUCT

no. of a's mod 2

no. of b's mod 2

	a	b
→*	0	0
	0	1
	1	0
	1	1

4 states

0 → even no.

1 → odd no.

$00 \xrightarrow{a} 10$

$00 \xrightarrow{b} 01$

$10 \xrightarrow{a} 00$

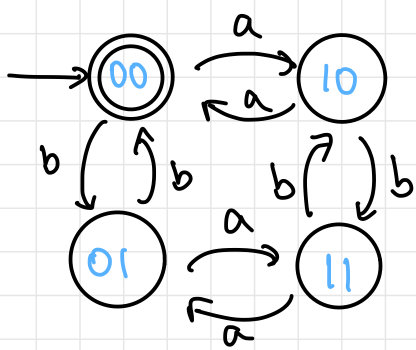
$10 \xrightarrow{b} 11$

$11 \xrightarrow{a} 01$

$11 \xrightarrow{b} 10$

$01 \xrightarrow{a} 11$

$01 \xrightarrow{b} 00$



If Question:

$L = \{ n_a(w) \bmod 2 = 1 \text{ and } n_b(w) \bmod 2 = 0 \}$

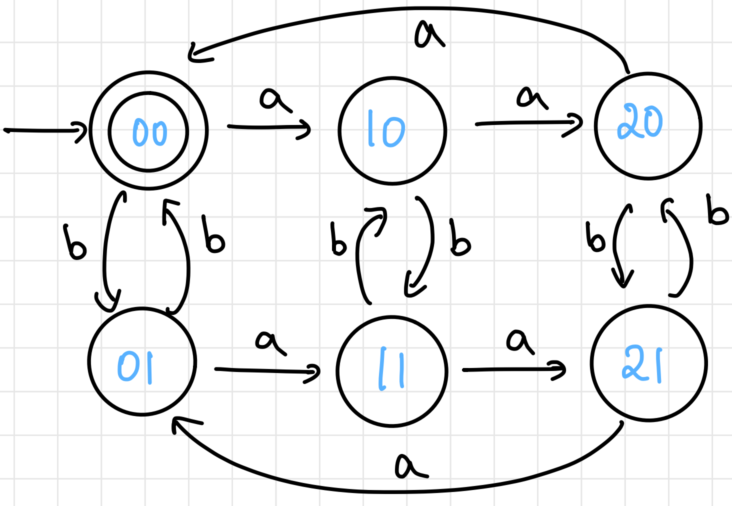
$w \in \{a,b\}^*$

final state = (10)

Question 15

$\mathcal{L} = \{ n_a(w) \bmod 3 = 0 \text{ and } n_b(w) \bmod 2 = 0,\ w \in \{a,b\}^* \}$

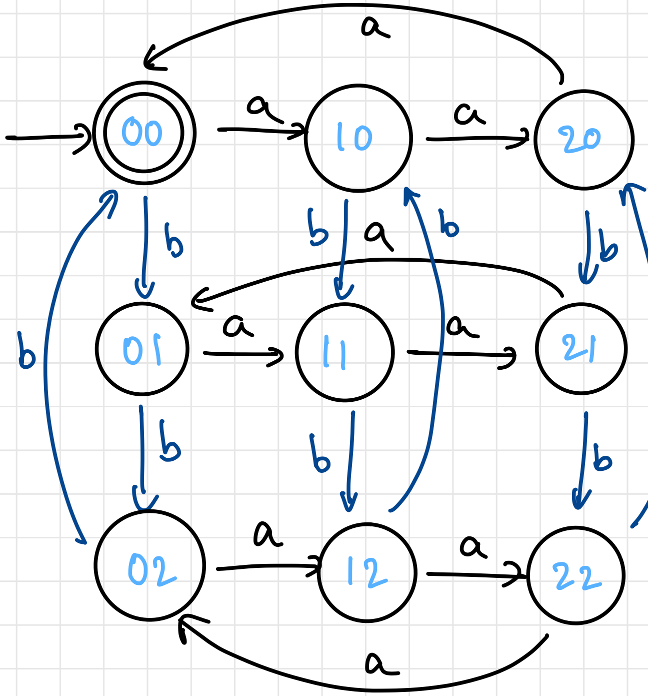
$n_a(w) \bmod 3$ (a)	$n_b(w) \bmod 2$ (b)	
0	0	00
0	1	01
1	0	10
1	1	11
2	0	20
2	1	21



Question 16

$L = \{ n_a(w) \bmod 3 = 0 \text{ and } n_b(w) \bmod 3 = 0 \}$

a	b
0	0
0	1
0	2
1	0
1	1
1	2
2	0
2	1
2	2



Question 17

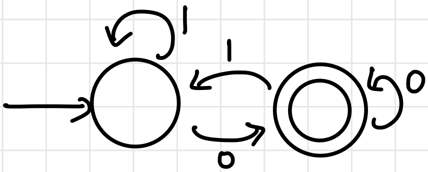
$L = \{ n_a(w) \bmod 3 = 2 \text{ and } n_b(w) \bmod 3 = 1 \}$

same as Question 16, final state (21)

Question 18

$\mathcal{L} = \{$ binary no. divisible by 2 $\}$

$\Sigma = \{0,1\}$, $\Sigma^* = \{0,1\}^* \supset \mathcal{L}$



* Question 19

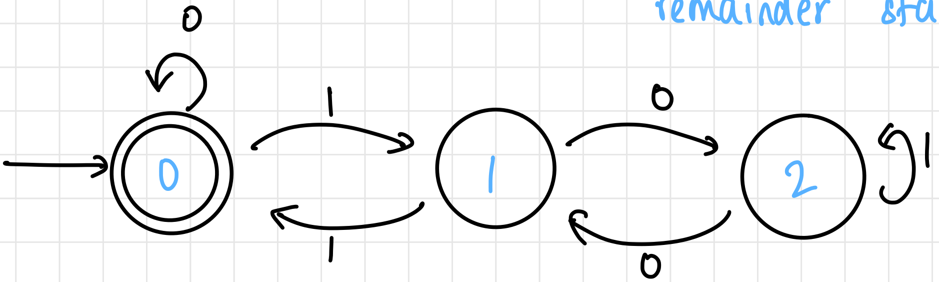
$\mathcal{L} = \{ w \mid w \bmod 3 = 0,\ w \in \{1,0\}^* \}$

Binary no. divisible by 3

8	4	2	1	
1	0	0	0	— 8
1	0	1	1	— 11
1	1	0	1	— 13

rem 0		rem 1		rem 2	
0	0	1	1	2	10
3	11	4	100	5	101
6	110	7	111	8	1000
9	1001	10	1010	11	1011

remainder states



State 1

10 $\longrightarrow$ 2 (2)
11 $\longrightarrow$ 3 (0)

State 2

101 $\longrightarrow$ 5 (2)
100 $\longrightarrow$ 4 (1)

Shortcut:

I/P

states	0	1
0	0	1
1	2	0
2	1	2

÷ by 4

	0	1
0	0	1
1	2	3
2	0	1
3	2	3

* Question 20

$L = \{ w \mid w \bmod 2 = 0 \text{ and } w \bmod 3 \neq 0 \}$
$w \in \{1,0\}^*$

6 states

÷2		÷3		
0	1	0	1	2

2	3	Binary	Decimal
0	0	0000	0
1	1	0001	1
0	2	0010	2
1	0	0011	3
0	1	0100	4
1	2	0101	5
0	0	0110	6
1	1	0111	7

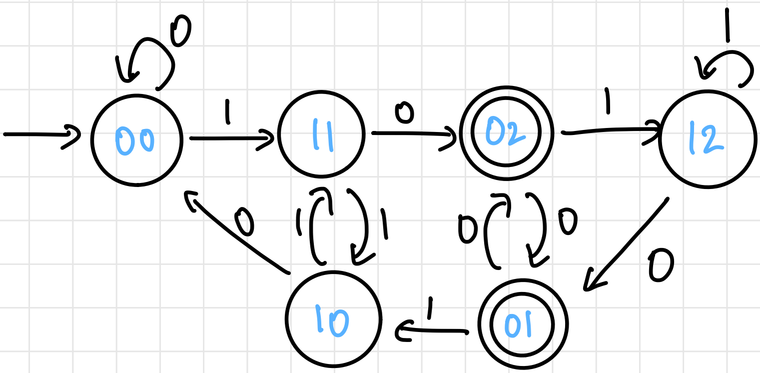
[11] 10 → 2 (0,2)
11 → 3 (1,0)

[02] 101 → 5 (1,2)
100 → 4 (0,1)

[10] 110 → 6 (0,0)
111 → 7 (1,1)

[01] 1001 → 9 (1,0)
1000 → 8 (0,2)

[12] 1010 → 10 (0,1)
1011 → 11 (1,2)



Question 21

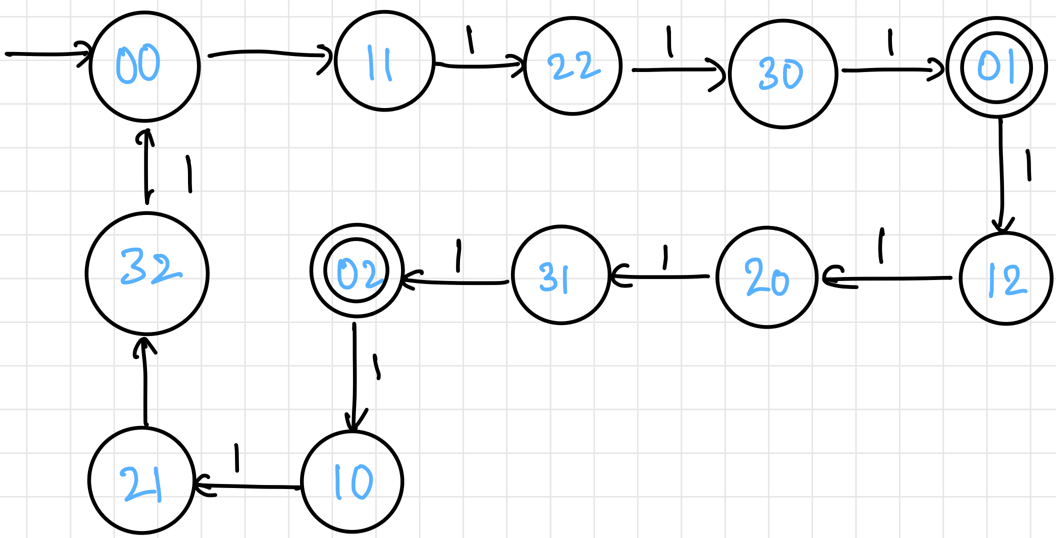
Unary no. divisible by 4 but not 3

	rem ÷ 4	rem ÷ 3
	0	0
	1	1
	2	2
	3	0

No ÷ by 4 = 0 1 2 3

No ÷ by 3 = 0 1 2

] 4 x 3 = 12 states



NON-DETERMINISTIC FINITE ACCEPTOR (NFA)

- On a given input, can go to any no. of states
- Transitions not determined; choices
- Computation looks like a tree
- Construction of NFA easier, but must be converted to DFA
- DFAs will be constructed as synchronous sequential circuits.

Formal Definition

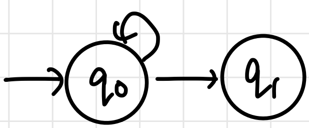
- Five-tuple (quintuple)

$$M = \{Q, \Sigma, \delta, q_0, F)$$

Q = set of finite states
Σ = set of finite I/P symbols
δ = transition
q_0 = start state
F = set of final states

$\phi \rightarrow$ no trans.

$$\delta = Q \times \Sigma \rightarrow 2^Q$$

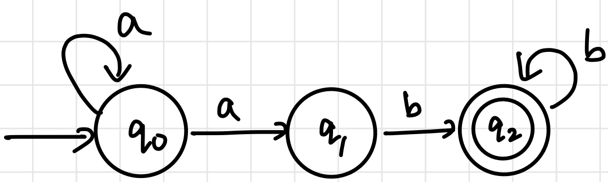


$\underbrace{\{\phi, q_0, q_1, \{q_0, q_1\}\}}_{\text{power set}}$

Question 22

Construct an NFA that accepts a string of a's followed by b's

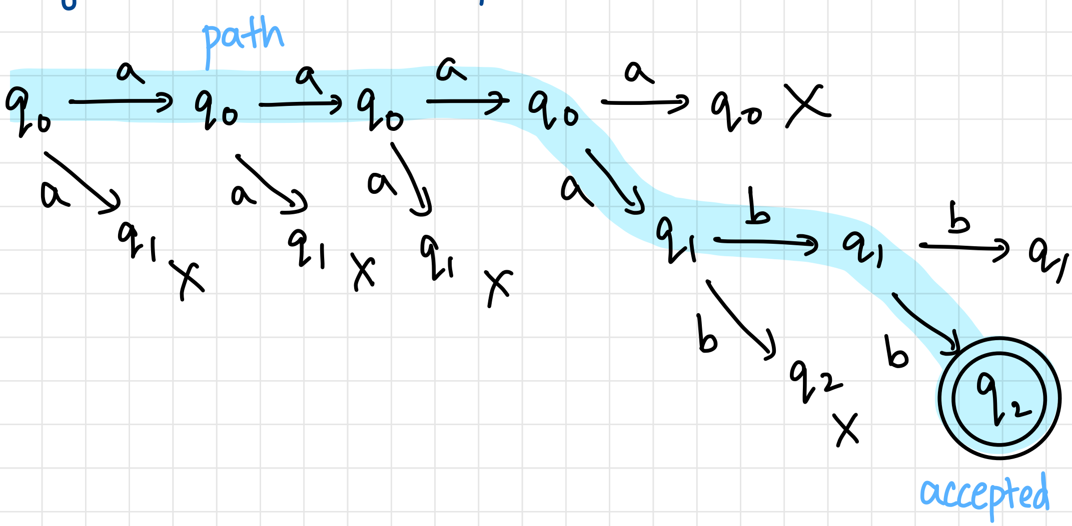
minimal string: $(a)_n\ ab\ (b)_m$



Tree → computation

$L = \{a^n b^m \mid n, m \geq 1\}$

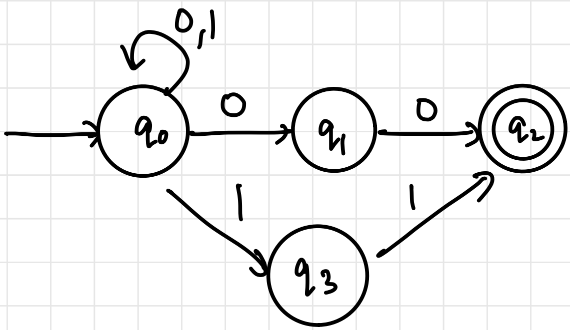
try: aaaabb → computation



Question 23

String that ends with 2 0's or end with 2 1's

$L = \{ w00 \text{ or } w11 \mid w \in \{0,1\}^* \}$

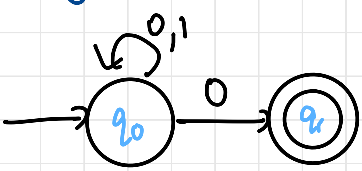


Transition Table

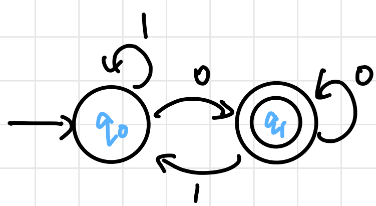
	0	1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_0, q_3\}$
q_1	q_2	ϕ
$*q_2$	ϕ	ϕ
q_3	ϕ	q_2

Question 24

Binary even no.



NFA

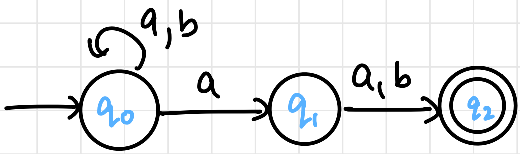


DFA

Question 25

NFA: String where second symbol from RHS is a

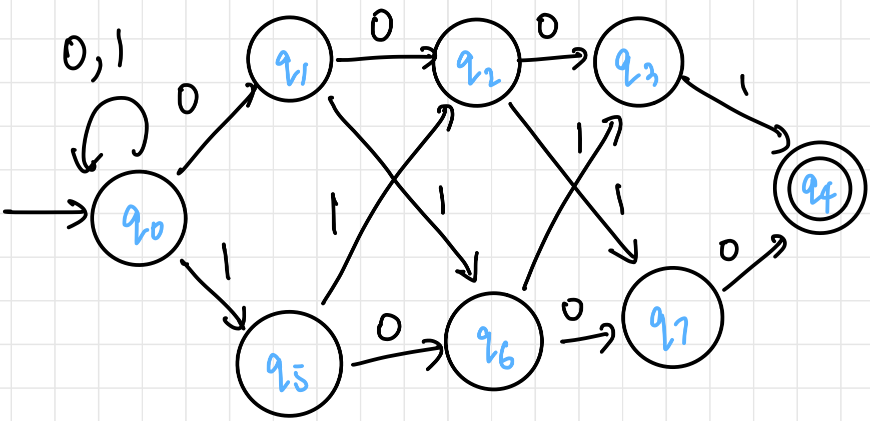
$(a \text{ or } b)_n \; a \; (a \text{ or } b)$



Question 26

NFA: $\mathcal{L}$ = { binary string, sum of last 4 digits is odd }

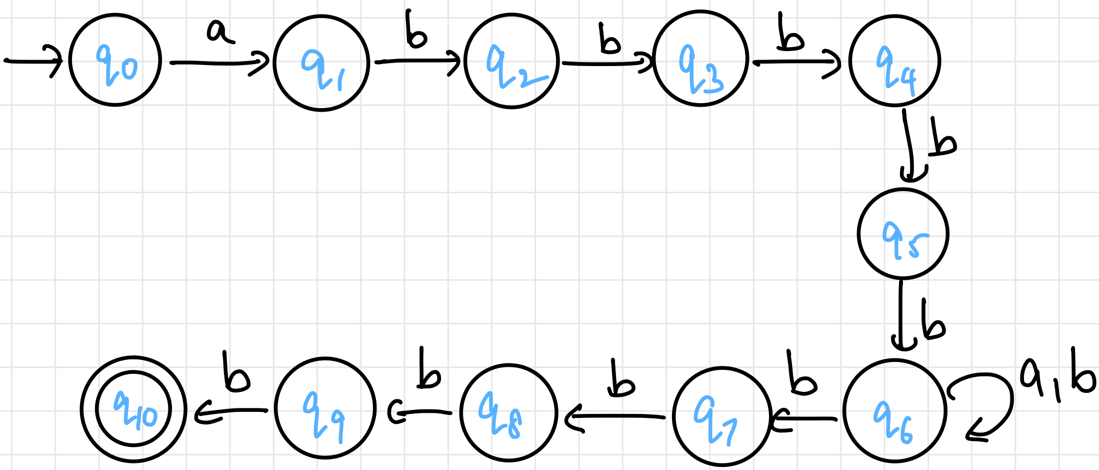
0001 1
0010 2
0100 4
0111 7
1000 8
1011 11
1101 13
1110 14



Question 27

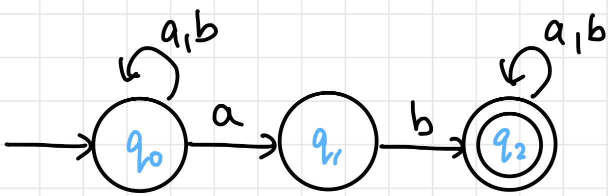
NFA: $\mathcal{L} = \{ a b^5 w b^4 \mid w \in \{a,b\}^* \}$

$a\ bbbbb\ (a,b)_n\ bbbb$



Question 28

NFA: $\mathcal{L} = \{ w\,ab\,w \mid w \in \{a,b\}^* \}$



Note: Regular languages: accepted by NFA/DFA

NFA TO DFA
subset construction method

ALGORITHM

NFA state diagram

↓

DFA state table

↓

DFA state diagram

1. Insert start state of NFA as start state of DFA
2. Repeat: $\forall\ a \in \Sigma$

$$\delta(q_i, a) \rightarrow q_j$$

New row of DFA state

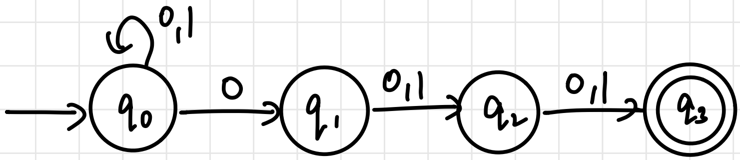
Question 29

$L = \{$ Third last symbol is 0 $\}$

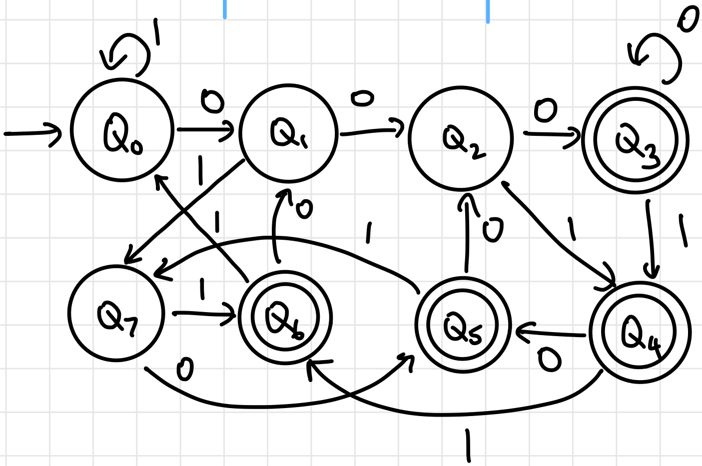
$$(0,1)_n \quad 0 \quad (0,1)_2$$

Convert NFA to DFA

NFA

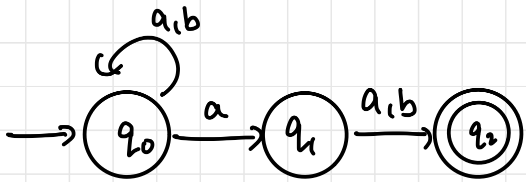


new state		0		1	
Q_0	$\rightarrow q_0$	$\{q_0, q_1\}$	Q_1	q_0	Q_0
Q_1	$\{q_0, q_1\}$	$\{q_0, q_1, q_2\}$	Q_2	$\{q_0, q_2\}$	Q_7
Q_2	$\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2, q_3\}$	Q_3	$\{q_0, q_2, q_3\}$	Q_4
Q_3	$*\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_1, q_2, q_3\}$	Q_3	$\{q_0, q_2, q_3\}$	Q_4
Q_4	$*\{q_0, q_2, q_3\}$	$\{q_0, q_1, q_3\}$	Q_5	$\{q_0, q_3\}$	Q_6
Q_5	$*\{q_0, q_1, q_3\}$	$\{q_0, q_1, q_2\}$	Q_2	$\{q_0, q_2\}$	Q_7
Q_6	$*\{q_0, q_3\}$	$\{q_0, q_1\}$	Q_1	q_0	Q_0
Q_7	$\{q_0, q_2\}$	$\{q_0, q_1, q_3\}$	Q_5	$\{q_0, q_3\}$	Q_6



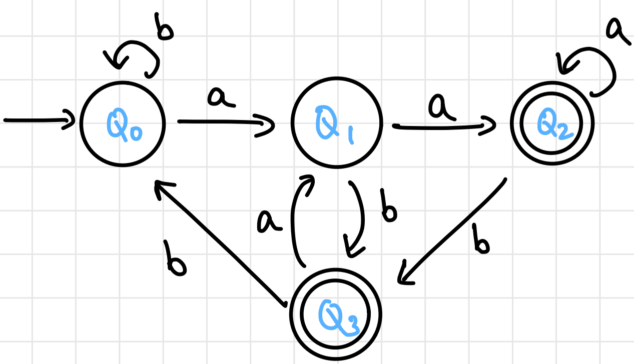
Question 30

$L = \{\text{second last symbol is } a\}$



			a	b
Q_0		$\rightarrow q_0$	$\{q_0, q_1\}$ Q_1	q_0 Q_0
Q_1	new state	$\{q_0, q_1\}$	$\{q_0, q_1, q_2\}$ Q_2	$\{q_0, q_2\}$ Q_3
Q_2		$*\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2\}$ Q_2	$\{q_0, q_2\}$ Q_3
Q_3		$*\{q_0, q_2\}$	$\{q_0, q_1\}$ Q_1	q_0 Q_0

perform union

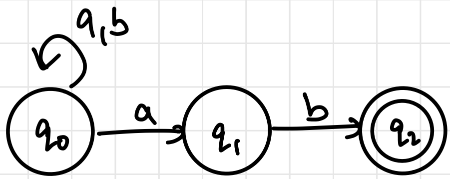


try: aabab

Question 31

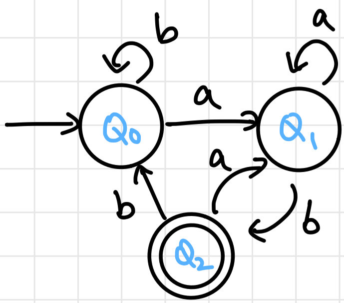
$L = \{\text{strings ending in } ab\}$

NFA



Transition Table

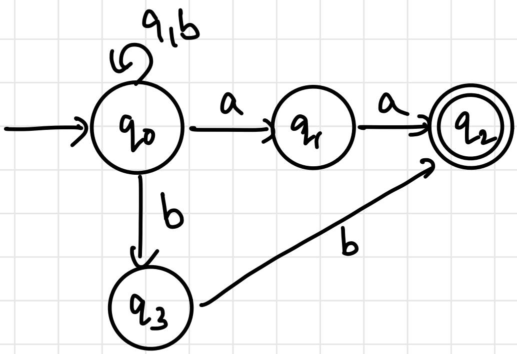
		a	b
Q_0	$\rightarrow q_0$	$\{q_0, q_1\}$ Q_1	q_0 Q_0
Q_1	$\{q_0, q_1\}$	$\{q_0, q_1\}$ Q_1	$\{q_0, q_2\}$ Q_2
Q_2	$*\{q_0, q_2\}$	$\{q_0, q_1\}$ Q_1	q_0 Q_0



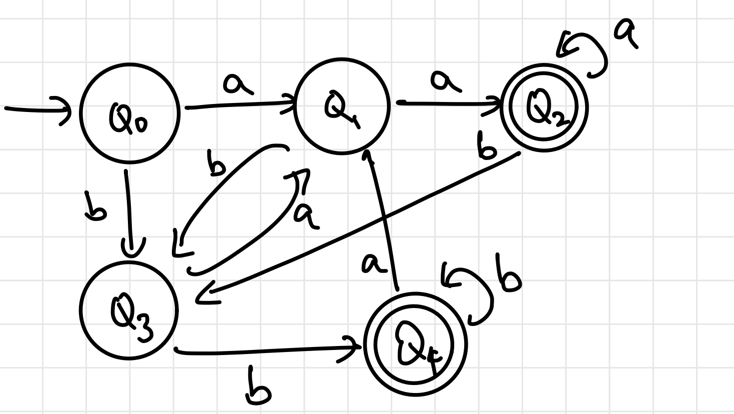
try: aab

Question 32

$L = \{$ string ends with aa or ends with bb $\}$



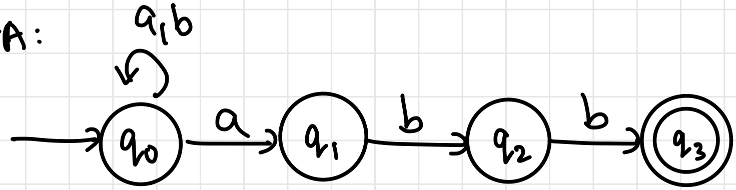
		a	b
Q_0	$\rightarrow q_0$	$\{q_0, q_1\}$ Q_1	$\{q_0, q_3\}$ Q_3
Q_1	$\{q_0, q_1\}$	$\{q_0, q_1, q_2\}$ Q_2	$\{q_0, q_3\}$ Q_3
Q_2	$*\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2\}$ Q_2	$\{q_0, q_3\}$ Q_3
Q_3	$\{q_0, q_3\}$	$\{q_0, q_1\}$ Q_1	$\{q_0, q_3, q_2\}$ Q_4
Q_4	$*\{q_0, q_3, q_2\}$	$\{q_0, q_1\}$ Q_1	$\{q_0, q_3, q_2\}$ Q_4



Question 33

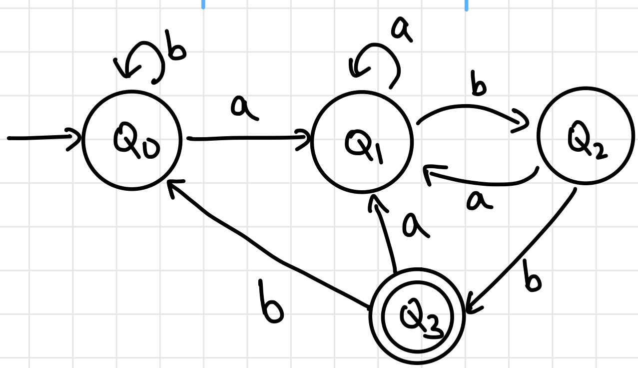
$\mathcal{L} = \{wabb \mid w \in \{a,b\}^*\}$ NFA → DFA

NFA:



Transition Table

		a	b
Q_0	$\rightarrow q_0$	$\{q_0, q_1\}$ Q_1	q_0 Q_0
Q_1	$\{q_0, q_1\}$	$\{q_0, q_1\}$ Q_1	$\{q_0, q_2\}$ Q_2
Q_2	$\{q_0, q_2\}$	$\{q_0, q_1\}$ Q_1	$\{q_0, q_3\}$ Q_3
Q_3	$*\{q_0, q_3\}$	$\{q_0, q_1\}$ Q_1	q_0 Q_0



try: ababb

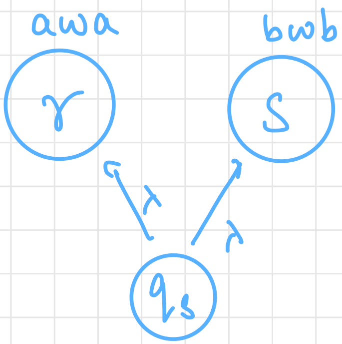
λ-NFA

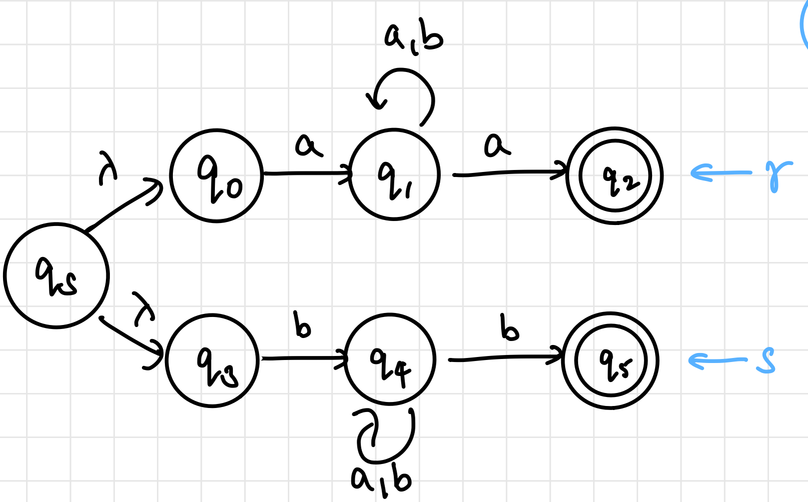
- still non-deterministic
- automata with λ transition
- without any input can move to different states
- regex

Question 34

$L = \{$ start & end with same symbol $\in \{a,b\} \}$
λ-NFA

awa or bwb

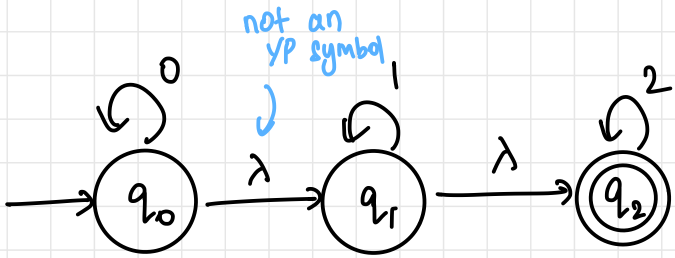




Question 35

$L = \{ 012 \text{ accepted} \}$ (λ-NFA)

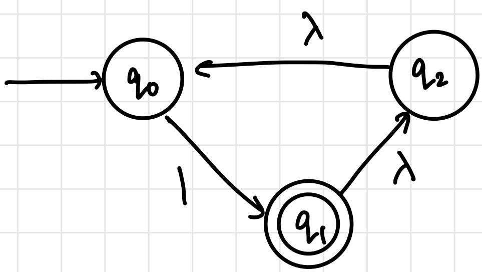
0 12	✓		
00	✓	0	✓
01	✓	1	✓
012	✓	2	✓
12	✓	21	✗
2	✓	10	✗



try: 2 ✓

Question 36

What language is accepted?



$L = \{ 1^n \mid n \geq 1 \}$

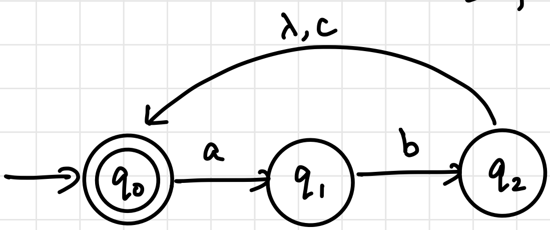
λ-closure : set of all states reachable from current state just by λ moves including itself

$\lambda\text{ closure}(q_1) = \{q_1, q_2, q_0\}$

Question 37

Construct λ-NFA for the language that accepts the string $(ab|abc)^*$ using only 3 states

$\{\lambda, ab, abc, abab, abcabc \ldots\}$



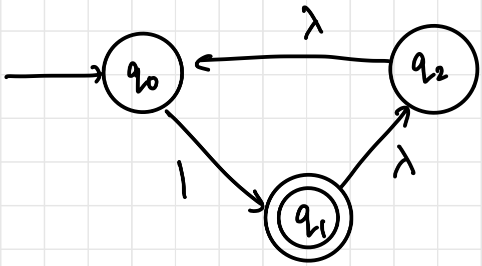
$M = \{Q, \Sigma, \delta, q_0, F\}$

$$\delta = Q \times (\Sigma \cup \lambda) \rightarrow 2^Q$$

To convert λ NFA to DFA is similar to NFA to DFA, taking into account λ-closure of states & start state

Note: λ-closure of start state of λ-NFA is start state of DFA

Question 38

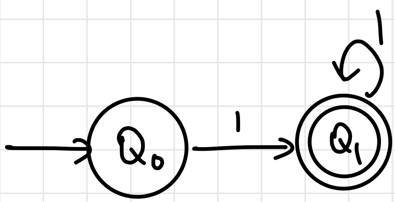


Construct DFA for this λ-NFA

do not forget!!

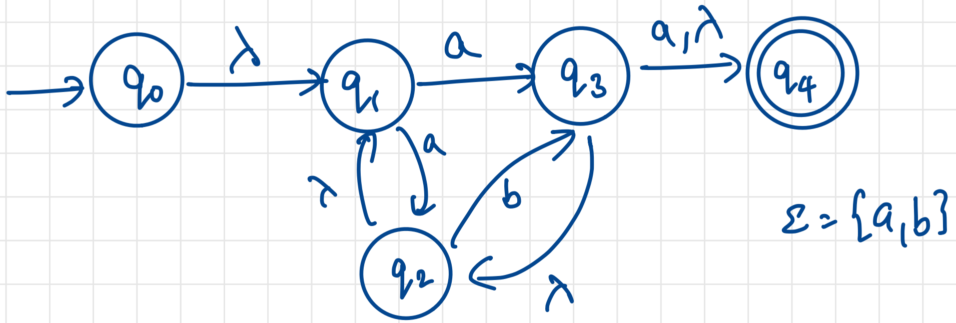
states	1
Q_0 $\rightarrow q_0$	$\{q_1, q_2, q_0\}$ Q_1
Q_1 $\{q_1, q_2, q_0\}$	$\{q_1, q_2, q_2\}$ Q_1

λ-closure $[\delta(q_0, 1)]$



Question 39

Compute λ closure for all states

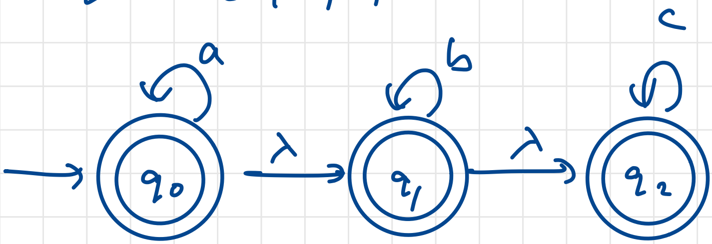


state	a	b	λ	λ-closure
$\rightarrow q_0$	ϕ	ϕ	q_1	$q_0 q_1$
q_1	$\{q_2 q_3\}$	ϕ	ϕ	q_1
q_2	ϕ	q_3	q_1	$q_1 q_2$
q_3	q_4	ϕ	$\{q_2 q_4\}$	$q_1 q_2 q_3 q_4$
$* q_4$	ϕ	ϕ	ϕ	q_4

Question 40

Convert to DFA

$L = \{a^n b^m c^k \mid n,m,k \geq 0\}$



λ-NFA

states

	state	a	b	c	λ	λ-closure
*→	q_0	q_0	$\emptyset$	$\emptyset$	q_1	$\{q_0, q_1, q_2\}$
*	q_1	$\emptyset$	q_1	$\emptyset$	$\emptyset$	$\{q_1, q_2\}$
*	q_2	$\emptyset$	$\emptyset$	q_2	$\emptyset$	q_2

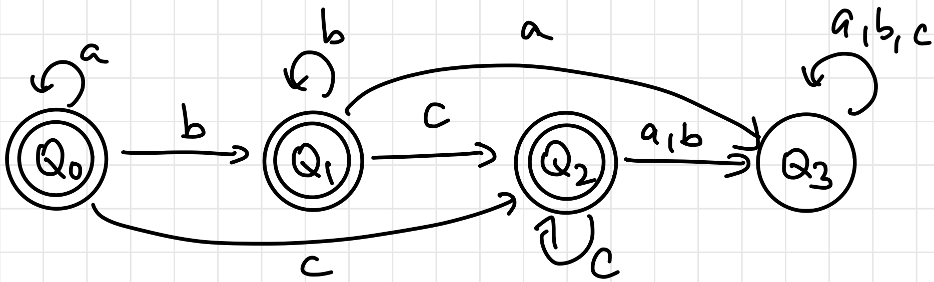
Transition table for DFA

find start state as λ-closure

		state	a	b	c
Q_0	*→	$\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2\}$ Q_0	$\{q_1, q_2\}$ Q_1	q_2 Q_2
Q_1	*	$\{q_1, q_2\}$	$\emptyset$ Q_3	$\{q_1, q_2\}$ Q_1	q_2 Q_2
Q_2	*	q_2	$\emptyset$ Q_3	$\emptyset$ Q_3	q_2 Q_2
Q_3		$\emptyset$	$\emptyset$ Q_3	$\emptyset$ Q_3	$\emptyset$ Q_3

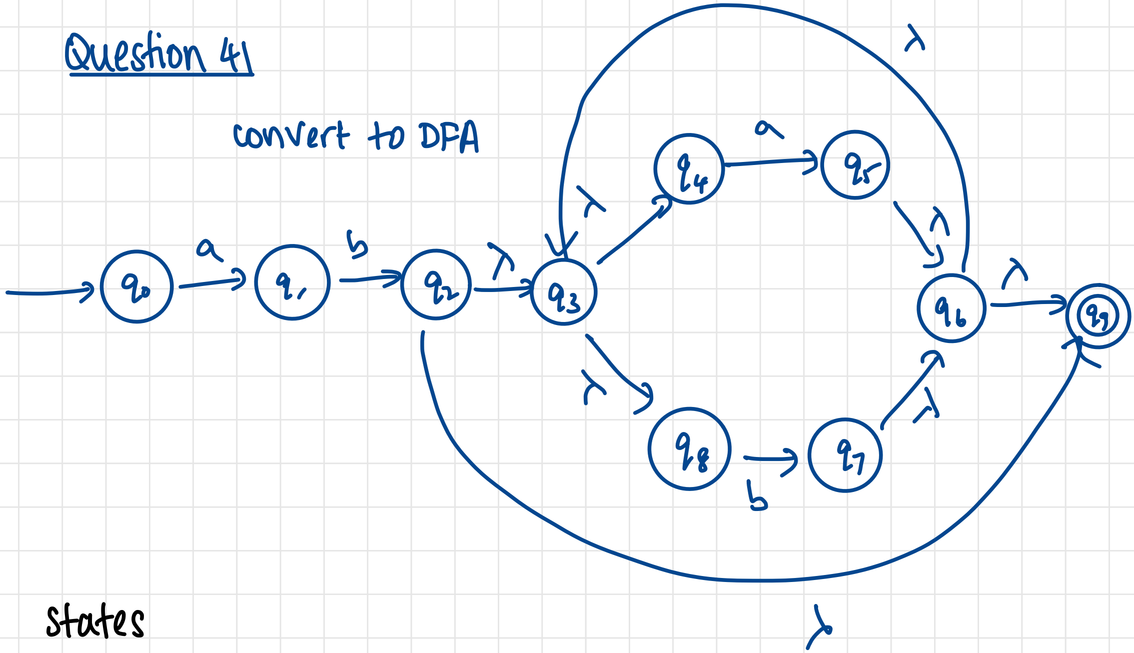
λ closure(q_0)

end states



Question 41

convert to DFA

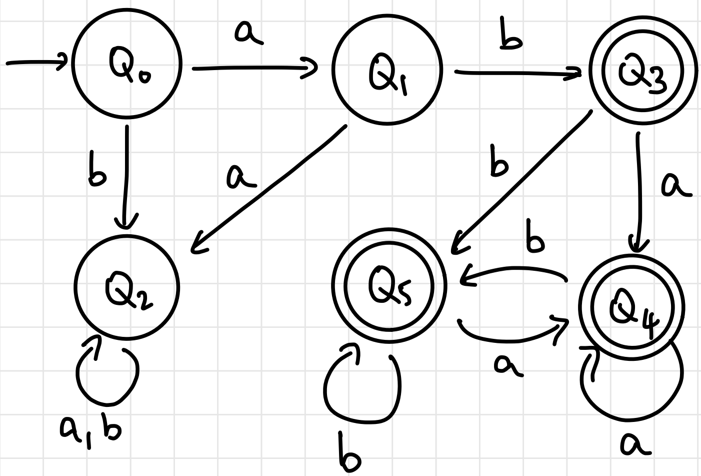


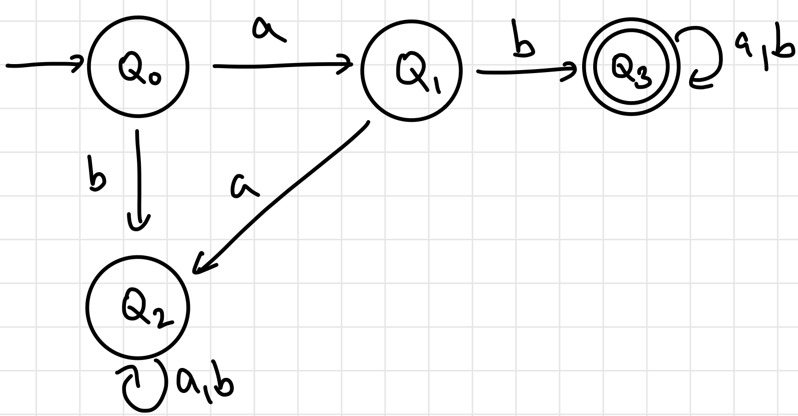
States

state	a	b	λ	λ-closure
→ q_0	q_1	∅	∅	q_0
q_1	∅	q_2	∅	q_1
q_2	∅	∅	$q_3 q_9$	$q_2 q_3 q_4 q_8 q_9$
q_3	∅	∅	$q_4 q_8$	$q_3 q_4 q_8$
q_4	q_5	∅	∅	q_4
q_5	∅	∅	q_6	$q_5 q_6 q_3 q_4 q_8 q_9$
q_6	∅	∅	$q_3 q_9$	$q_6 q_3 q_9 q_4 q_8$
q_7	∅	∅	q_6	$q_7 q_6 q_3 q_4 q_8 q_9$
q_8	∅	q_7	∅	q_8
* q_9	∅	∅	∅	q_9

Transition Table for DFA

	state	a	b
Q_0	$\rightarrow q_0$	q_1 Q_1	ϕ Q_2
Q_1	q_1	ϕ Q_2	$q_2q_3q_4q_8q_9$ Q_3
Q_3	$*\,q_2q_3q_4q_8q_9$	$q_5q_6q_3q_4q_8q_9$ Q_4	$q_7q_6q_3q_4q_8q_9$ Q_5
Q_4	$*\,q_5q_6q_3q_4q_8q_9$	$q_5q_6q_3q_4q_8q_9$ Q_4	$q_7q_6q_3q_4q_8q_9$ Q_5
Q_5	$*\,q_7q_6q_3q_4q_8q_9$	$q_5q_6q_3q_4q_8q_9$ Q_4	$q_7q_6q_3q_4q_8q_9$ Q_5
Q_2	ϕ	ϕ Q_2	ϕ Q_2

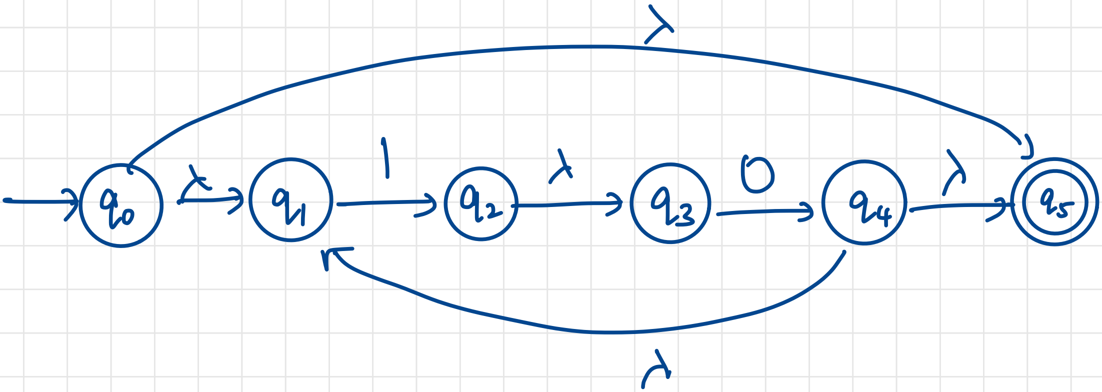




Question 42

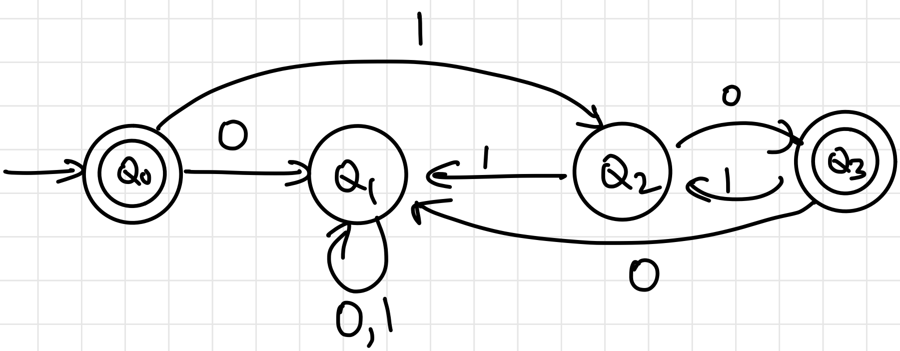
$L = \{ (10)^n \mid n \geq 0 \}$

Convert λNFA to DFA



DFA

	State	0	1
Q_0	$* \rightarrow \{q_0, q_1, q_5\}$	$\emptyset$ Q_1	$\{q_2, q_3\}$ Q_2
Q_1	$\emptyset$	$\emptyset$ Q_1	$\emptyset$ Q_1
Q_2	$\{q_2, q_3\}$	$\{q_4, q_1, q_5\}$ Q_3	$\emptyset$ Q_1
Q_3	$* \{q_4, q_1, q_5\}$	$\emptyset$ Q_1	$\{q_2, q_3\}$ Q_2



Minimisation of DFAs

Mark & Reduce / Table Filling Algorithm

(ONLY ON DFA)

- Redundant states
- Merge states
- Compare 2 states and compare to merge (pair)
- Two kinds of pairs

Distinguishable pair

$(q_0\, q_1) \xrightarrow{a}$ (FS, NFS) or (NFS, FS)

$(q_0\, q_1) \xrightarrow{b}$ (FS, NFS) or (NFS, FS)

any 1 can satisfy; need not be both

Indistinguishable pair

can be diff

$(q_0\, q_1) \xrightarrow{a}$ (FS, FS) or (NFS, NFS)

$(q_0\, q_1) \xrightarrow{b}$ (FS, FS) or (NFS, NFS)

- cannot distinguish
- can merge
- must identify these pairs

- First find distinguishable pairs
- Remaining: indistinguishable

Steps

1. Eliminate unreachable states
2. Mark distinguishable pairs
3. Any unmarked, (M,N)

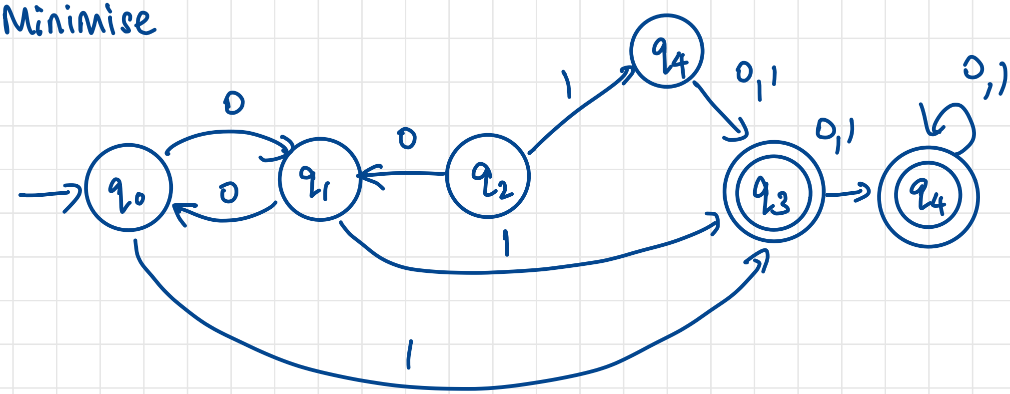
   if $\delta(M,a) = X$ and $\delta(N,a) = Y$ and (X,Y) is marked, mark (M,N)

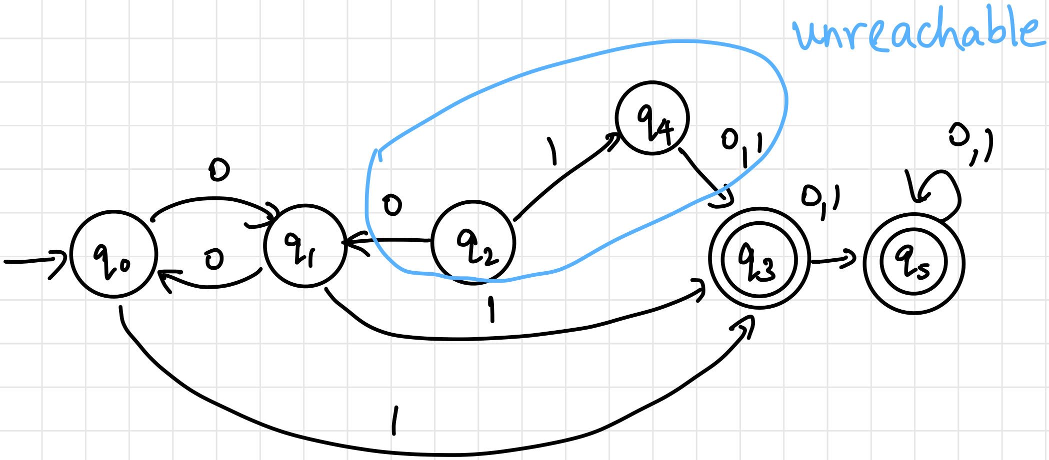
   Repeat until no new states can be marked

4. Combine unmarked states and make into single state

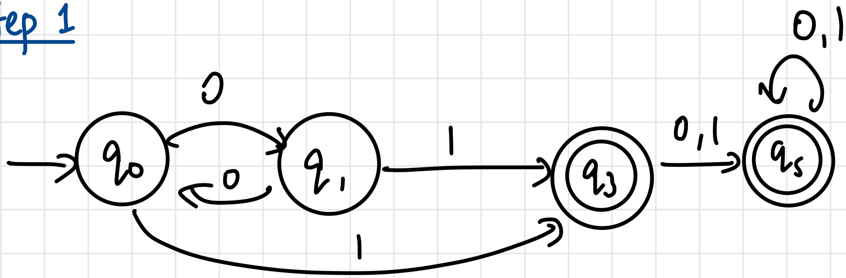
Question 43

Minimise





Step 1



Step 2

- Draw table (no need for reflexive & symmetric pairs) eg: $(q_0 q_1)$ & $(q_1 q_0)$ OR $(q_0 q_0)$, $(q_1 q_1)$
- Table is triangle (horizontally start & leave out 1)

ind. pairs first iteration

leave out first

q_1	○		
$*q_3$	X (NF,F)	X (NF,F)	
$*q_5$	X (NF,F)	X (NF,F)	○
	q_0	q_1	$*q_3$

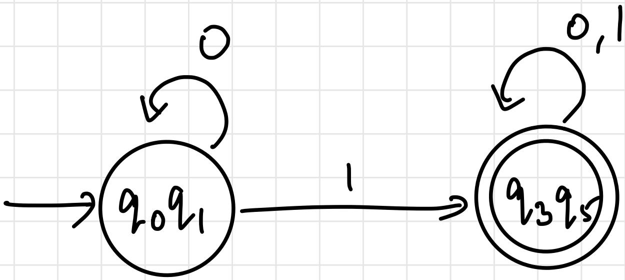
leave out last

Transition table

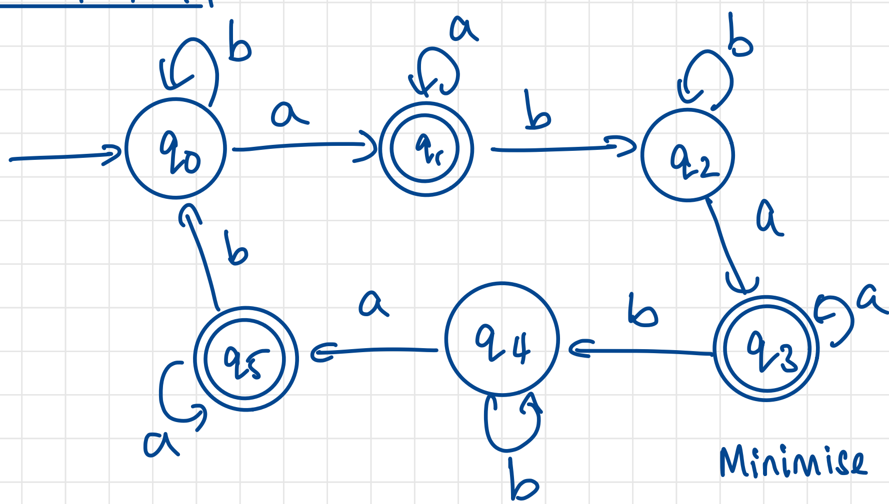
states	0	1
q_0	q_1	q_3
q_1	q_0	q_3
* q_3	q_5	q_5
* q_5	q_5	q_5

1) $(q_0 q_1)$ $\xrightarrow{0}$ $(q_1 q_0)$ UM
$\xrightarrow{1}$ $(q_3 q_3)$ X

2) $(q_3 q_5)$ $\xrightarrow{0}$ $(q_5 q_5)$ X
$\xrightarrow{1}$ $(q_5 q_5)$ X



Question 44



Minimise the DFA

	a	b
$\rightarrow q_0$	q_1	q_0
$* q_1$	q_1	q_2
q_2	q_3	q_2
$* q_3$	q_3	q_4
q_4	q_5	q_4
$* q_5$	q_5	q_0

$(q_0 q_4) \xrightarrow{a} (q_1 q_5)$ UM

$(q_0 q_4) \xrightarrow{b} (q_0 q_4)$ UM

$(q_0 q_2) \xrightarrow{a} (q_1 q_3)$ UM

$(q_0 q_2) \xrightarrow{b} (q_0 q_2)$ UM

	q_0	q_1 *	q_2	q_3 *	q_4
* q_1	X				
q_2		X			
* q_3	X		X		
q_4		X		X	
* q_5	X		X		X

$(q_1 q_5) \xrightarrow{a} (q_1 q_5)$ UM

$(q_1 q_5) \xrightarrow{b} (q_2 q_0)$ UM

$(q_1 q_3) \xrightarrow{a} (q_1 q_3)$ UM

$(q_1 q_3) \xrightarrow{b} (q_2 q_4)$ UM

$(q_2 q_4) \xrightarrow{a} (q_3 q_5)$ UM

$(q_2 q_4) \xrightarrow{b} (q_2 q_4)$ UM

$(q_3 q_5) \xrightarrow{a} (q_3 q_5)$ UM

$(q_3 q_5) \xrightarrow{b} (q_4 q_0)$ UM

All unmarked

find eq. class

$(q_0 q_4)$
$(q_0 q_2)$
$(q_1 q_5)$
$(q_1 q_3)$
$(q_2 q_4)$
$(q_3 q_5)$

equivalence relation

elements: $(q_0 q_1 q_2 q_3 q_4 q_5)$

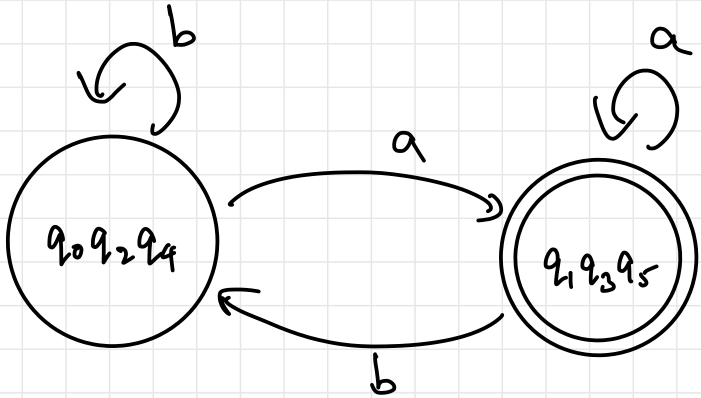
Equivalence classes

$[q_0] = (q_0 q_4 q_2)$
$[q_1] = (q_1 q_5 q_3)$
$[q_2] = (q_2 q_0 q_4)$
$[q_3] = (q_3 q_1 q_5)$
$[q_4] = (q_4 q_0 q_2)$
$[q_5] = (q_5 q_1 q_3)$

same elements $= (q_0 q_2 q_4)$

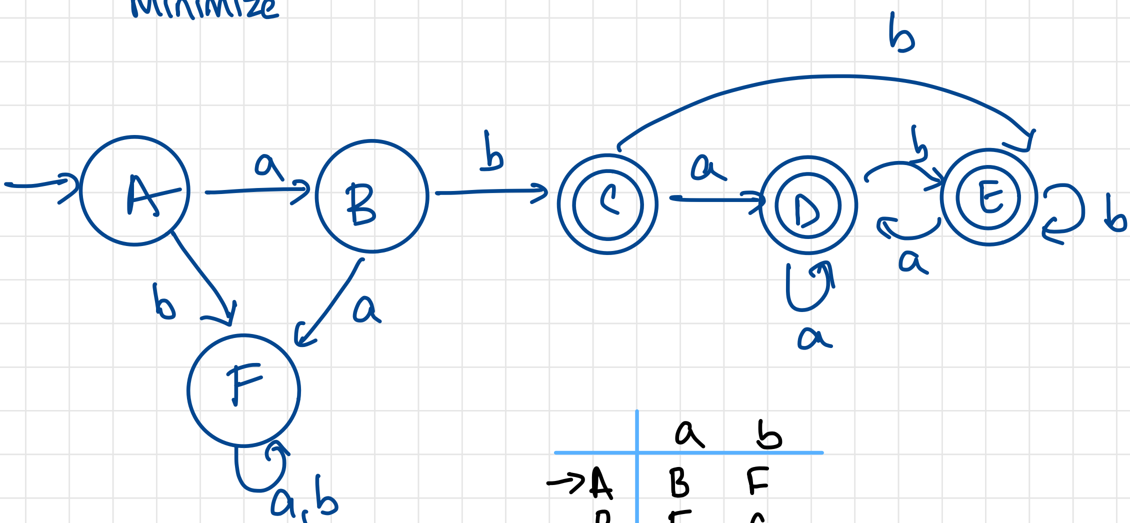
same elements $= (q_1 q_3 q_5)$

two partitions



Question 45

Minimize



	a	b
→A	B	F
B	F	C
*C	D	E
*D	D	E
*E	D	E
F	F	F

1st iteration
2nd iteration
3rd iteration

	A	B	C	D	E
B	X				
* C	X	X			
* D	X	X			
* E	X	X			
F	X	X	X	X	X
			*	*	*

		2nd	3rd
AB	a	BF	UM
	b	FC	M
AF	a	BF	UM/M
	b	FF	X
BF	a	FF	X
	b	CF	M
CE	a	DD	X
	b	EE	X
CD	a	DD	X
	b	EE	X
DE	a	DD	X
	b	EE	X

All unmarked

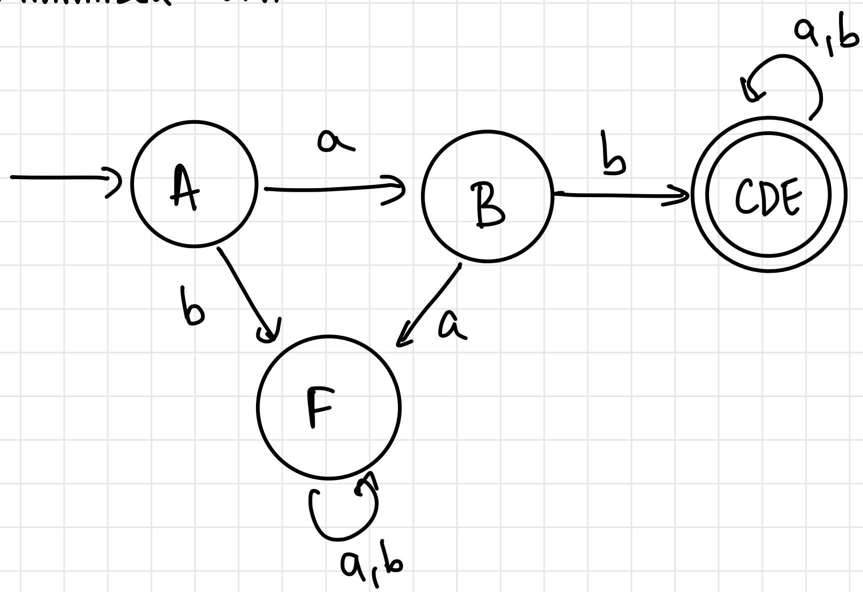
(CD, DE, CE)

equivalence class

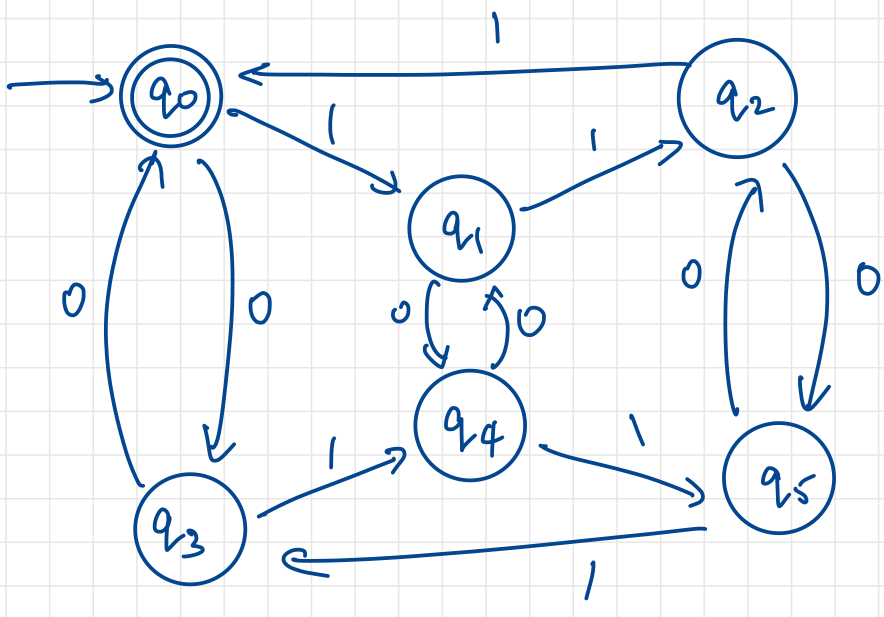
[C] = (CDE)
[D] = (DEC)
[E] = (ECD)

one state

Minimized DFA



Question 46



Transition table

	0	1
$\rightarrow * q_0$	q_3	q_1
q_1	q_4	q_2
q_2	q_5	q_0 ×
q_3	q_0	q_4
q_4	q_1	q_5
q_5	q_2	q_3

q_1	X				
q_2	X	X			
q_3	X	X	X		
q_4	X	X	X	X	
q_5	X	X	X	X	X
	$*q_0$	q_1	q_2	q_3	q_4

$\cancel{(q_1 q_5)}$ $\xrightarrow{0}$ $q_4 q_2$ UM M
$\xrightarrow{1}$ $q_2 q_3$ UM

$\cancel{(q_2 q_5)}$ $\xrightarrow{0}$ $q_5 q_2$ UM
$\xrightarrow{1}$ $q_0 q_3$ M

$\cancel{(q_1 q_4)}$ $\xrightarrow{0}$ $q_4 q_1$ UM
$\xrightarrow{1}$ $q_2 q_5$ UM M

$\cancel{(q_2 q_4)}$ $\xrightarrow{0}$ $q_5 q_1$ UM
$\xrightarrow{1}$ $q_0 q_5$ M

$\cancel{(q_1 q_3)}$ $\xrightarrow{0}$ $q_4 q_0$ M
$\xrightarrow{1}$ $q_2 q_4$ UM

$\cancel{(q_2 q_3)}$ $\xrightarrow{0}$ $q_5 q_0$ M
$\xrightarrow{1}$ $q_0 q_4$ M

$\cancel{(q_1 q_2)}$ $\xrightarrow{0}$ $q_4 q_5$ UM
$\xrightarrow{1}$ $q_2 q_0$ M

$\cancel{(q_3 q_5)}$ $\xrightarrow{0}$ $q_0 q_2$ M
$\xrightarrow{1}$ $q_4 q_3$ UM

$\cancel{(q_3 q_4)}$ $\xrightarrow{0}$ $q_0 q_1$ M

$\cancel{(q_3 q_4)}$ $\xrightarrow{1}$ $q_4 q_5$ UM

$\cancel{(q_4 q_5)}$ $\xrightarrow{0}$ $q_1 q_2$ UM

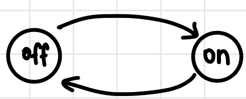
$\cancel{(q_4 q_5)}$ $\xrightarrow{1}$ $q_5 q_3$ UM/M

The DFA has already been minimized

Real-Time Applications of Automata

- text processing
- compilers
- network protocols
- hardware design
- gaming

1) Switch implementation



2) Code lock implementation
3) Communication link
   - acknowledgement
4) Spellcheck
   - error detection + suggestions prediction

use edit distance (Levenshtein distance)

Advantage
- testing

JFLAP

DFA → strings with 'aaa'