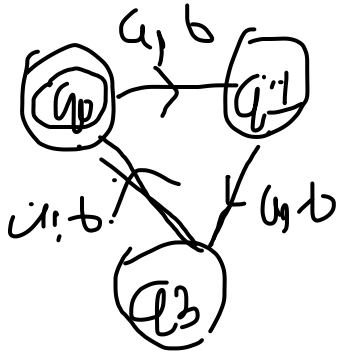
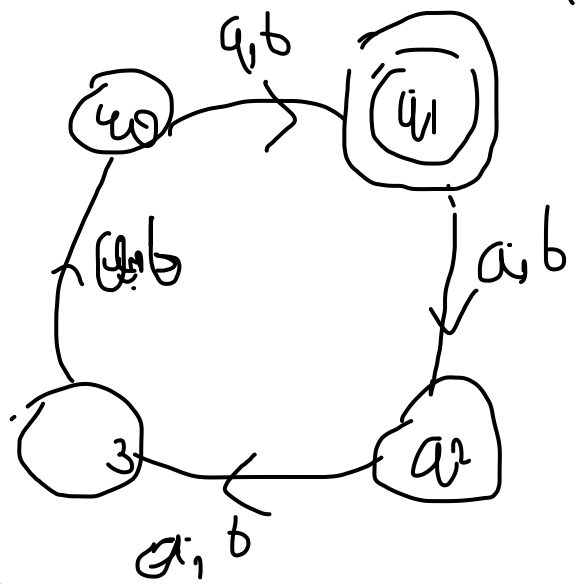
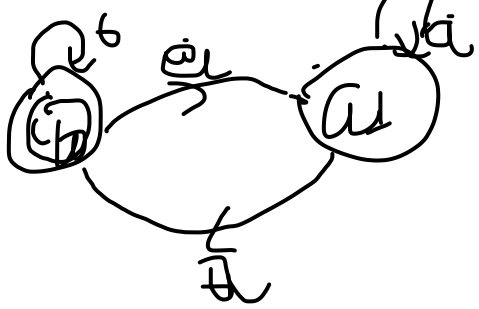
$|w| = 0 \pmod 3$



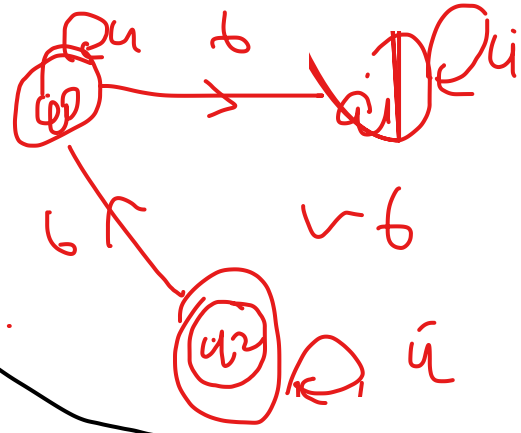
$|w| = 1 \pmod 4$



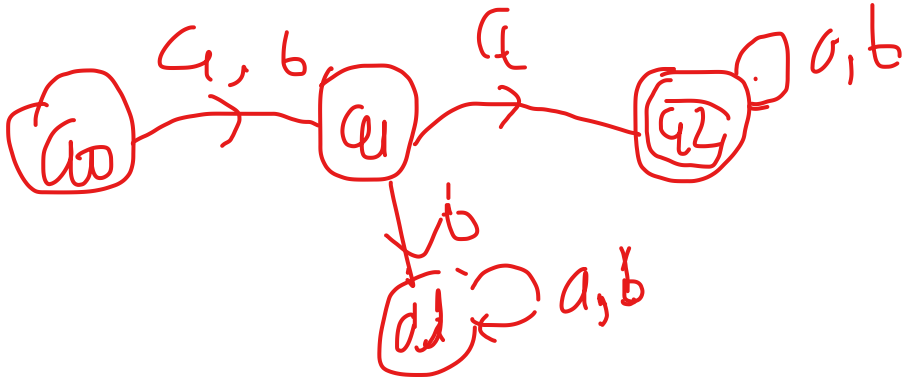
No. of a is $= 0 \pmod 2$



No. of a if $w \in$ ___ 2 mod 3

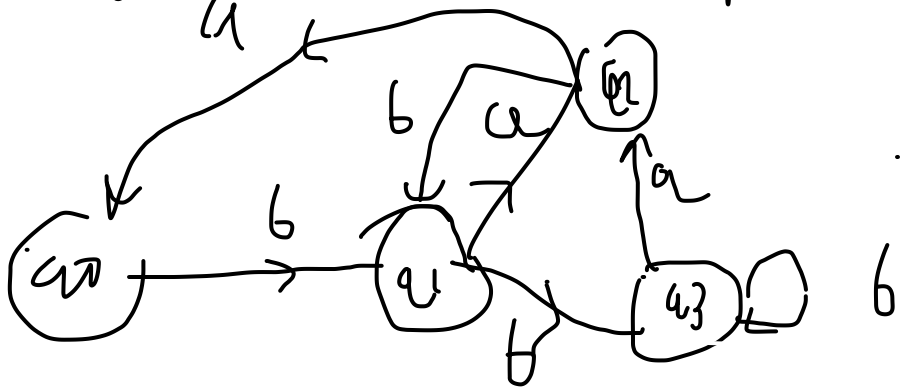


Design a DFA that starting 2nd from left is always a.



Ankur Dutta
2024/02/22

Design a DFA that 2nd from right is always b



---

1) Give the regular expressions:

a) Last two symbols are of same.

$\Rightarrow$ $a(a+b)^*b + b(a+b)^*a$.

b) even no. of a's and one or two b's

$\Rightarrow$ $(aa)^*b + (aa)^*bb$

Answered 2024/02/22

c) |v| mod 3 = 0 {v : v ∈ {a,b}*}

⇒ (v)^3 (✓ + b)* |v|^3

2019-2020

d) {w | w ∈ {a,b}*, last two symbols in w are not same}

⇒ a(a+b)*b + b(a+b)*a

b) {w | w ∈ {a,b}*, w has even no. of a's & one or two b's}

⇒ (aa)*b + (aa)*bb

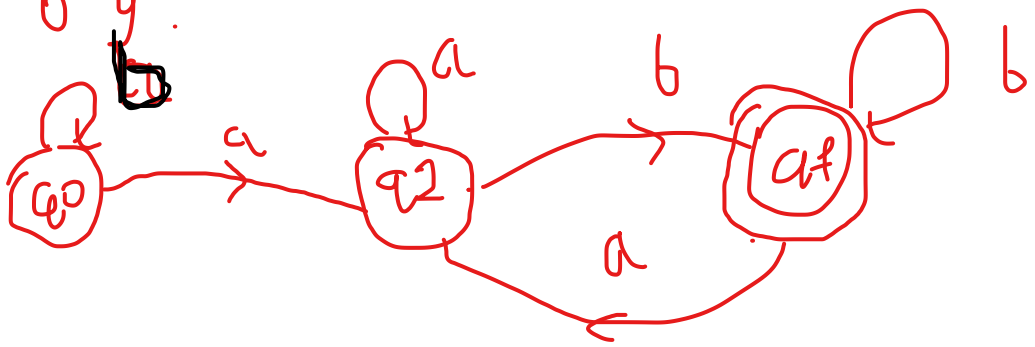
e) {w | w, w ∈ {a,b}*, |v| mod 3 = 0}

⇒ (v)^5 (a+b)* (v)^5

Ankur Dutta
2021/02/22

2) Draw the state transition diagram:-

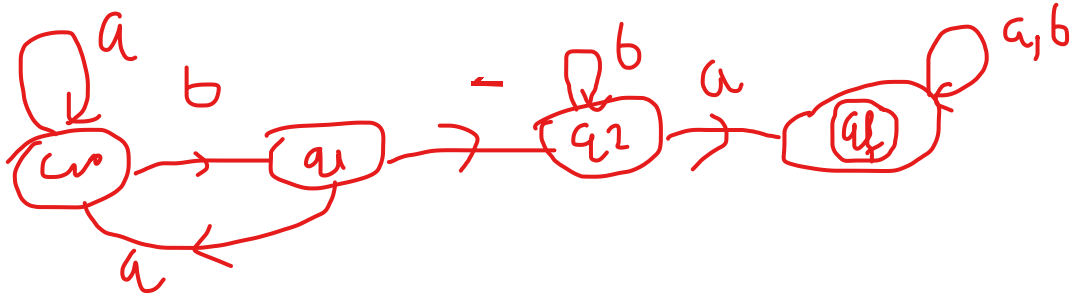
a) $\{w \mid w \in \{a,b\}^*$, w has a odd no. of a's ending with a b $\}$.



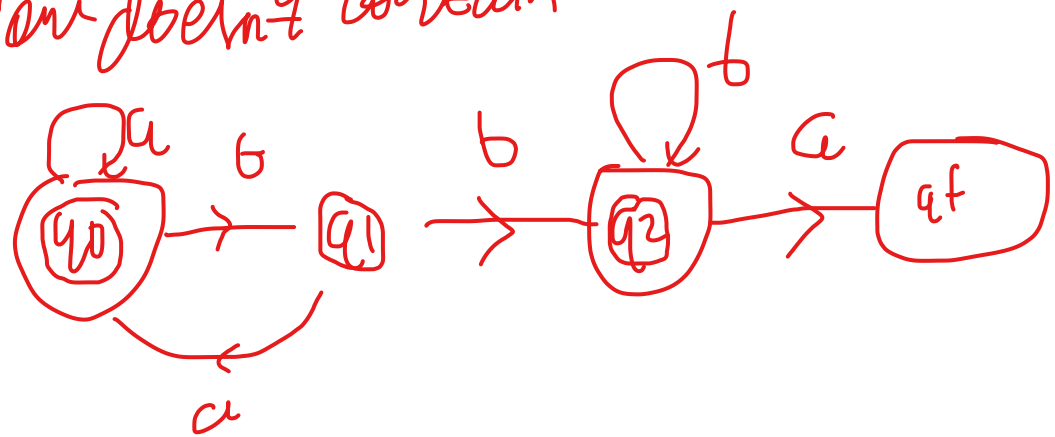
Ankur Dutta
2024/02/22

b) $\{w \mid w \in \{a,b\}^*$, w doesn't contain a substring bba $\}$

→ Step 1: contains substring bba

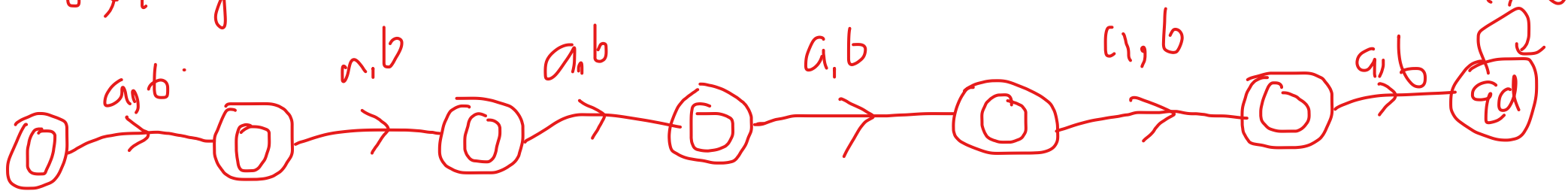


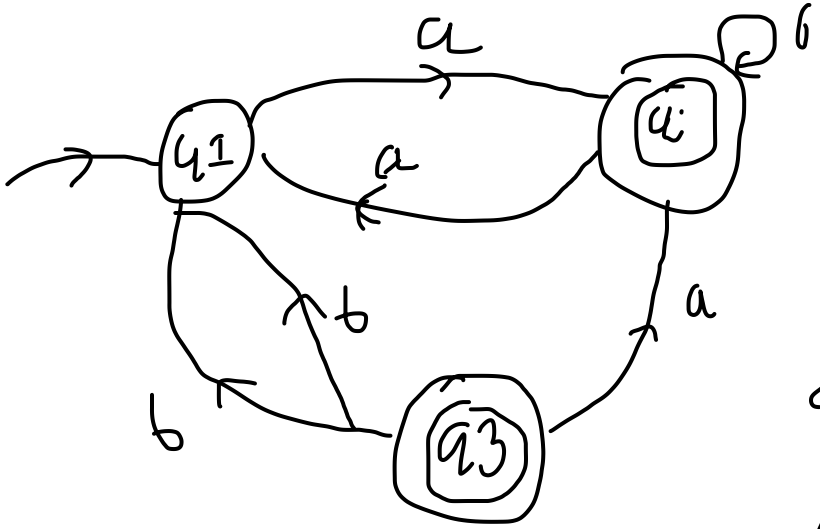
Now doesn't contain



7 $\{w | w \in \{a,b\}^*$, length of w is atmost 5$\}$

→ |w|<-5





Convert the 3 state DFA into regular expressions.

$q_1 = \in + q_2 \cdot a + q_3 \cdot b$ — (i)

$q_2 = q_1 \cdot a + q_2 \cdot b + q_3 \cdot a$ — (ii)

$q_3 = q_1 \cdot b$ — (iii)

Putting q_3 in q_2 we get

Ankur Dutta
2024/02/22

$q_2 = q_1 \cdot a + q_1 \cdot b + q_3 \cdot a$

$q_2 = q_1 \cdot a + q_2 b + q_1 \cdot b \cdot a$

$q_2 = q_1 \cdot a + q_1 \cdot b \cdot \quad + q_2 \cdot b$

$q_2 = q_1 (a + b \cdot a) + q_2 \cdot b$

Comparing with $R = Q + RP$

$q_2 = q_1 (a + b \cdot a) b^*$

Substituting q_2 & q_3 in (1) we get

$q_1 = \epsilon + q_2 \cdot a + q_3 \cdot b$

$q_1 = \epsilon + q_1 (a + b \cdot a) b^* + q_1 \cdot b \cdot b$

$q_1 = \epsilon + q_1 \left[(a + b \cdot a) b^* + b \cdot b \right]$

Comparing with $R = Q + RP$

$q_1 = \epsilon + q_1 \left[(a + b \cdot a) b^* + b \cdot b \right]^*$

$q_1 = \epsilon \left[(a + b \cdot a) b^* + b \cdot b \right]^*$

q_2

=

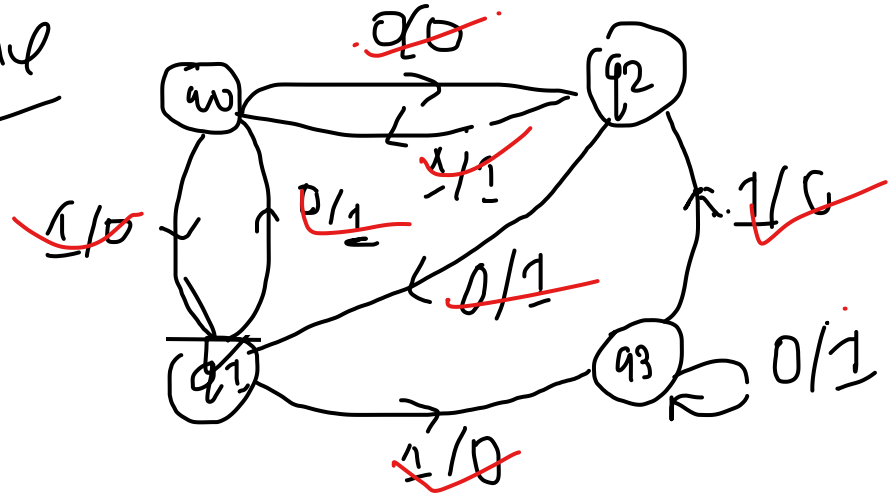
Bikku Durga
2024/02/22

Convert Mealy Machine to Moore Machine :

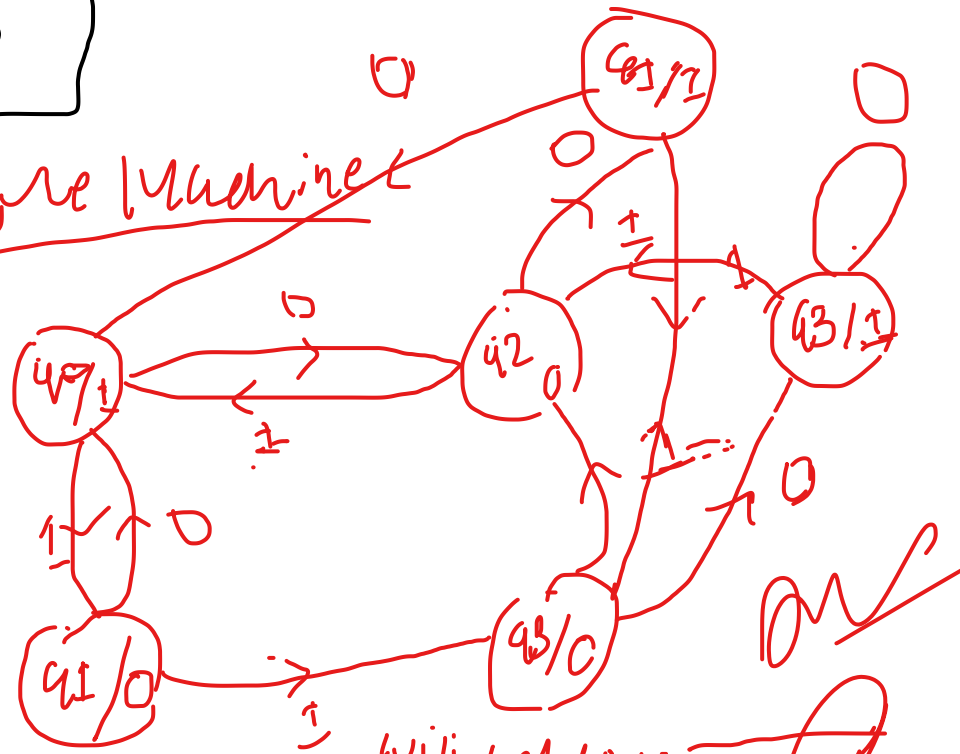
present state	next (0)	O/P (0)	next (1)	O/P (1)
q_0	q_2		q_1	0
q_1	q_0	1	q_3	0
q_2	q_1	1	q_0	1
q_3	q_3	1	q_2	0

Dakua Durga
2024/02/22

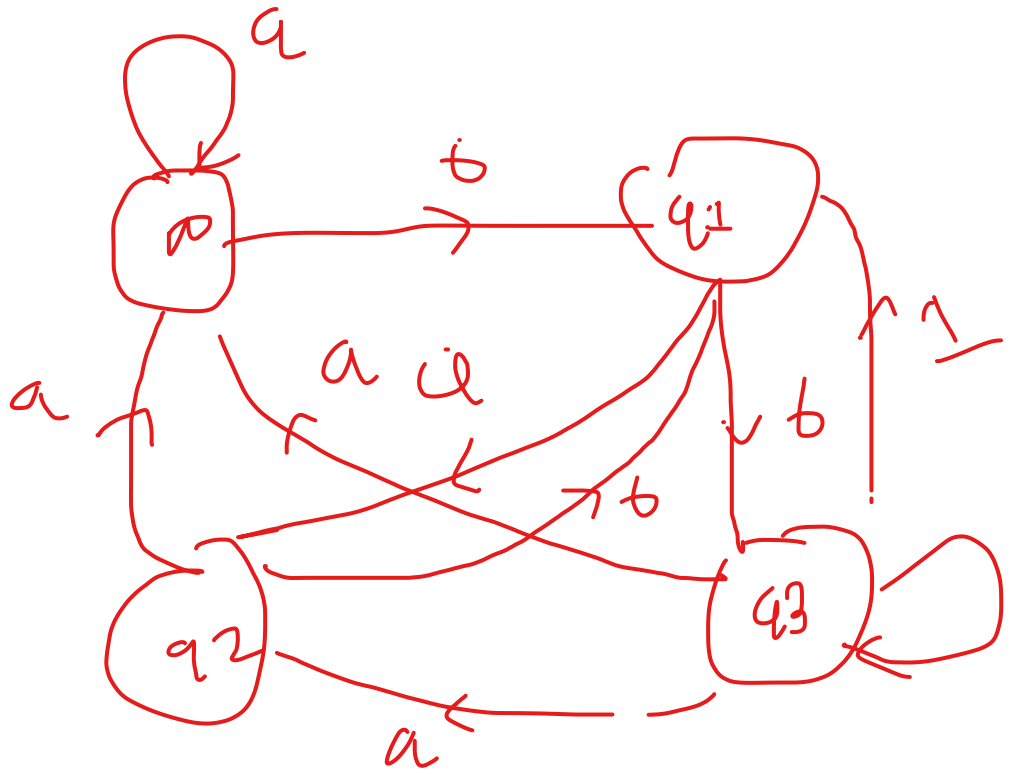
Mealy Machine



Moore Machine

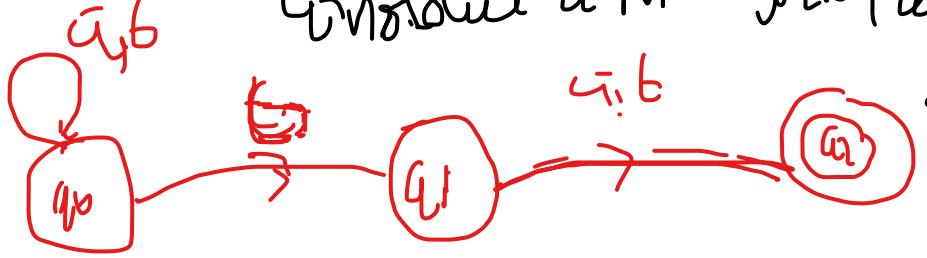


	a	b
$\to q_0$	q_0	q_1
q_1	q_2	q_3
q_2	q_0	q_1
$*q_3$	q_2	q_3



Mixed Quiz.
2024/02/22

Construct a NFA which accepts all strings over $\Sigma = \{a, b\}$ in which at most one 'a' and second symbol from right is b.



Non-deterministic

Non deterministic Finite Automata (NDFA / NFA)

→ Q : non-empty set of states
Σ : non-empty set of alphabets.
q_0 : set of initial states.
F : set of Final states.
δ : state transition function

$$\delta : Q \times \Sigma : 2^Q$$

Solution of Previous Year Questions.
2019-2020

(a)

→ $a(a+b)^*b + b(a+b)^*b$

(b)

→ $(aa)^*b + (ba)^*bb$

b

c)

→ $V = \{aaa, abb, aab, baa, bbb, bbb, abab\}$

$\nu = \gamma^* \beta, \gamma^*(a+b)^*\gamma^*$.

Dakuy Dutta.
2021/02/22

1 a) $a(a+b)^*b + b(a+b)^*a$

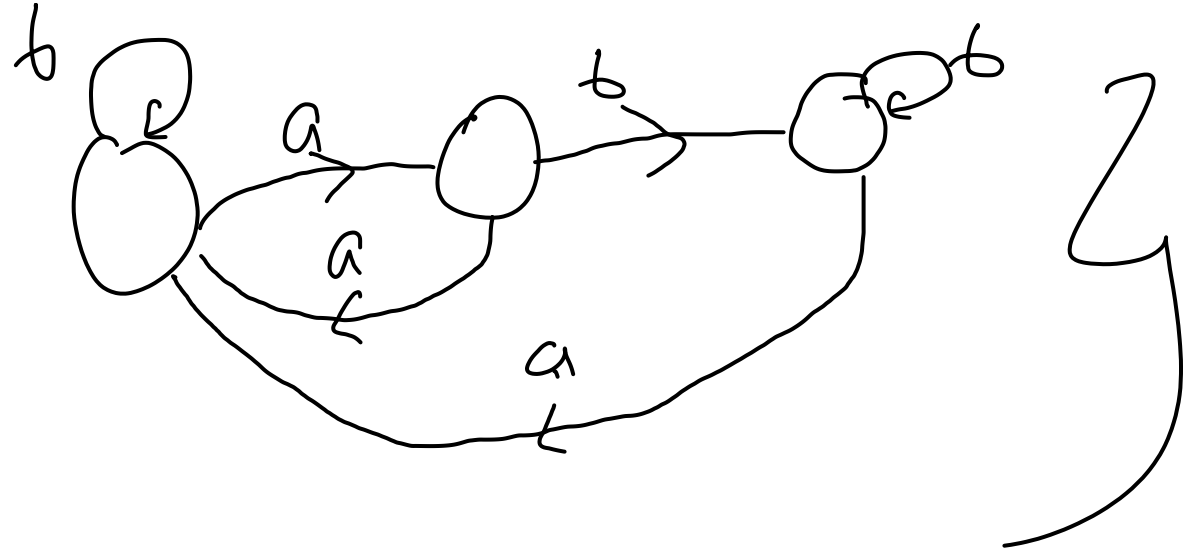
b)

c) $L = \{aaa, bbb, aaaa, abbb, baaa, bbbb, bbba, aabb\}$

$V = \gamma^*$

$\gamma^* (a+b)^* \gamma^*$

2a)

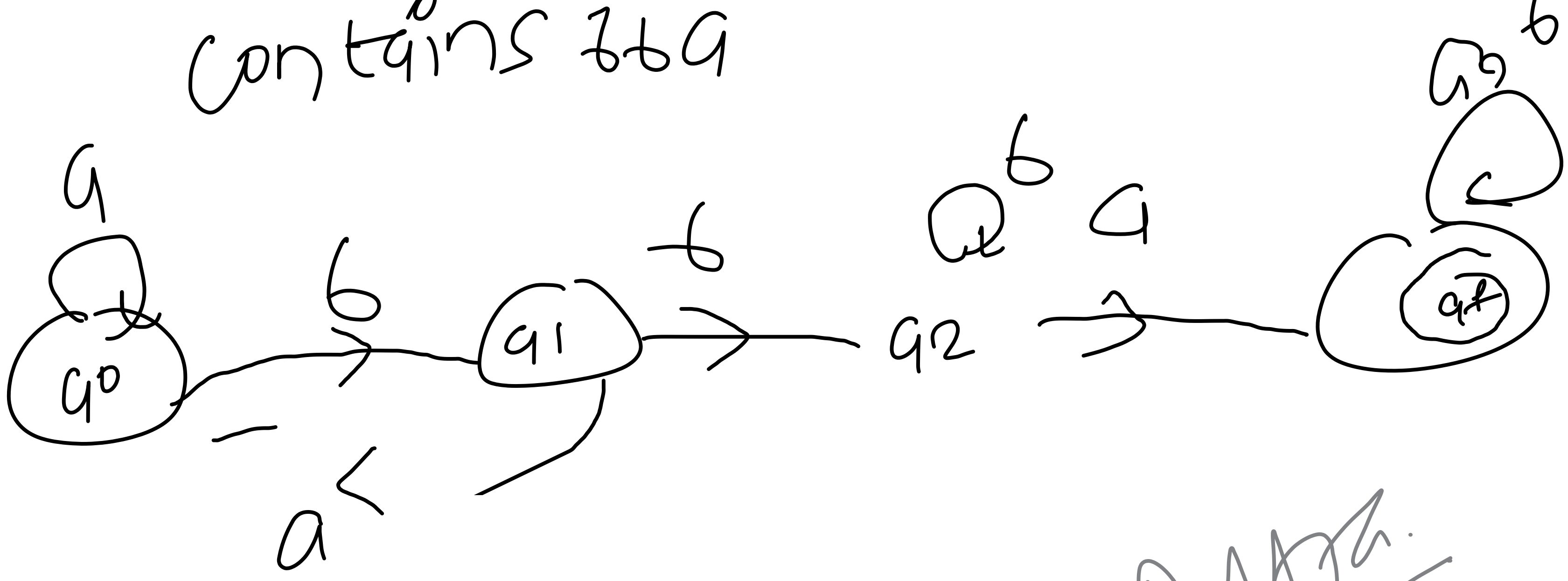


Ankur Dutta.

2024/02/22

b) w does not contain a substring bba.

Step 1: Draw transition diagram for contains bba

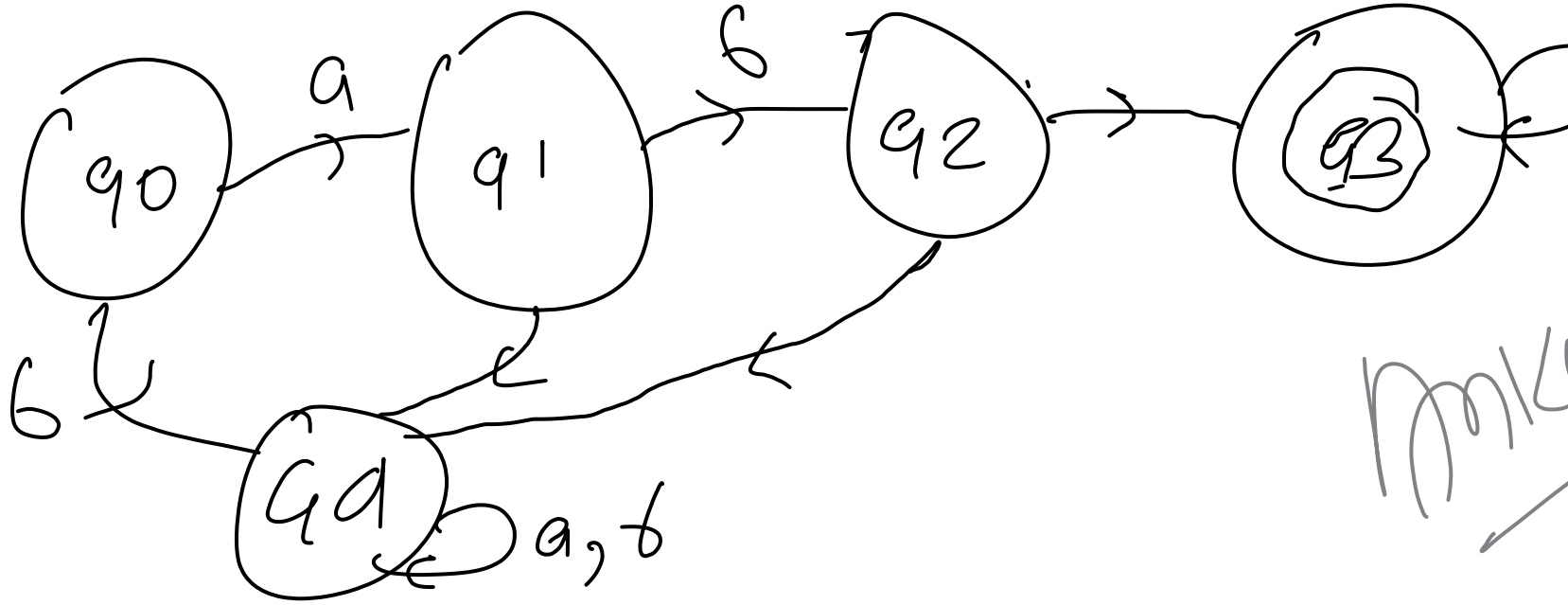


2024/02/22

c) length of w is at most 5

|w| <= 5

Ankush
20241022

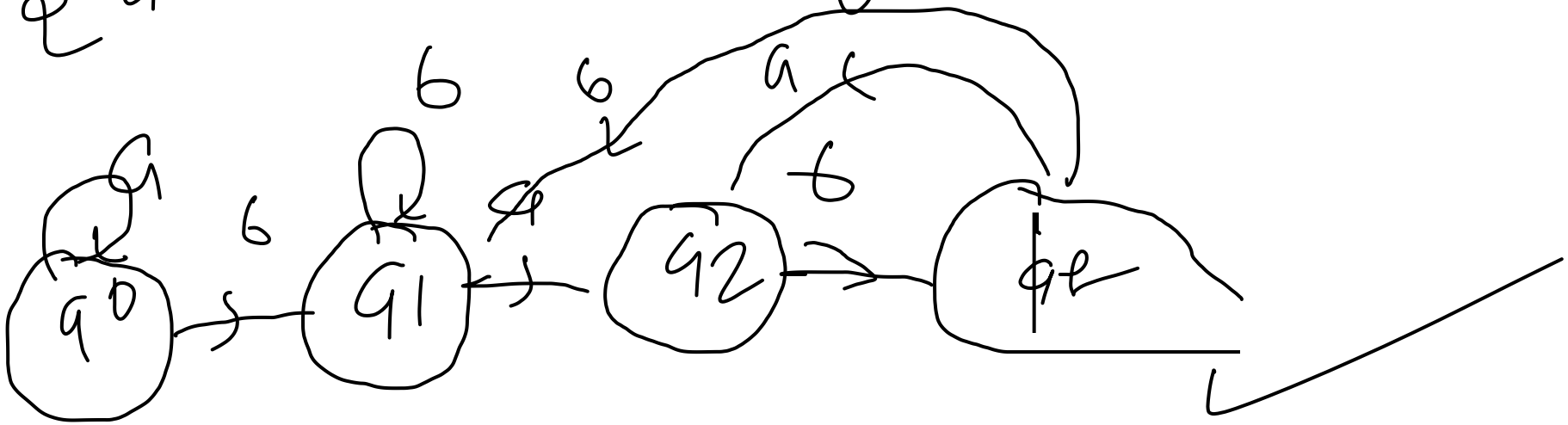


Mitu Dutta
2024/02/22

Eg: Strings that starts with ab and

→ This can end with any number of a's or any number of b's

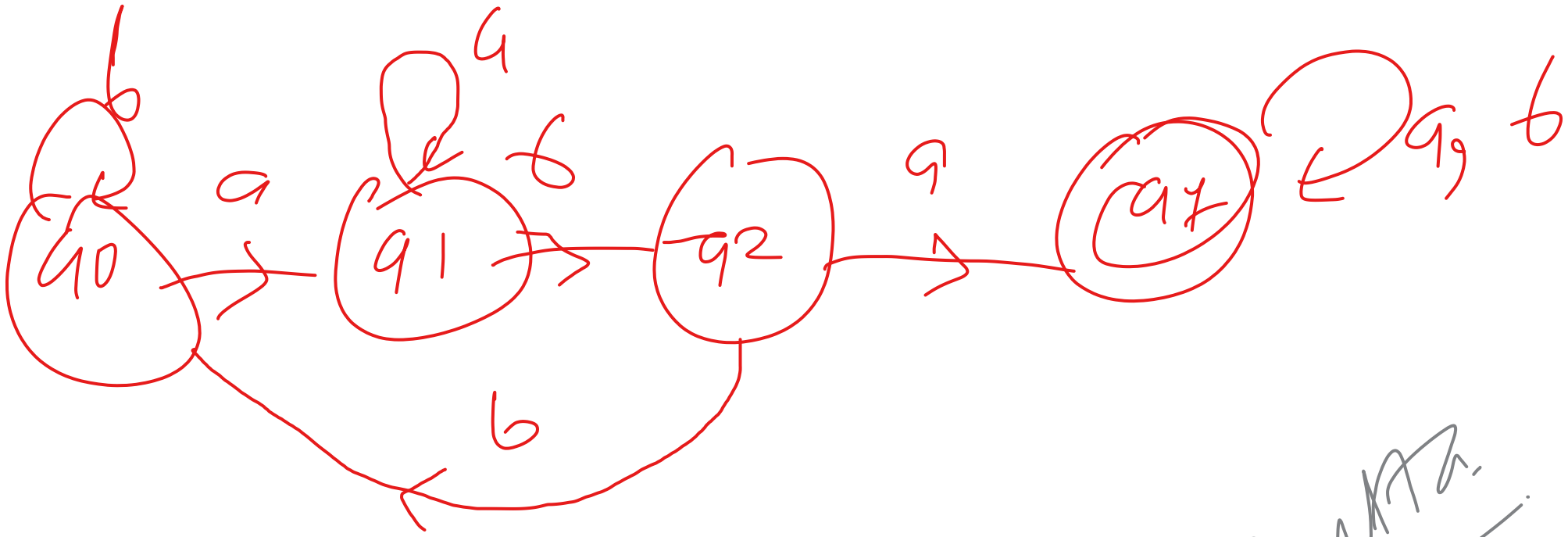
6) Accepted string w ends with 6 ab 3



c.) all the strings that contains substring

Abb

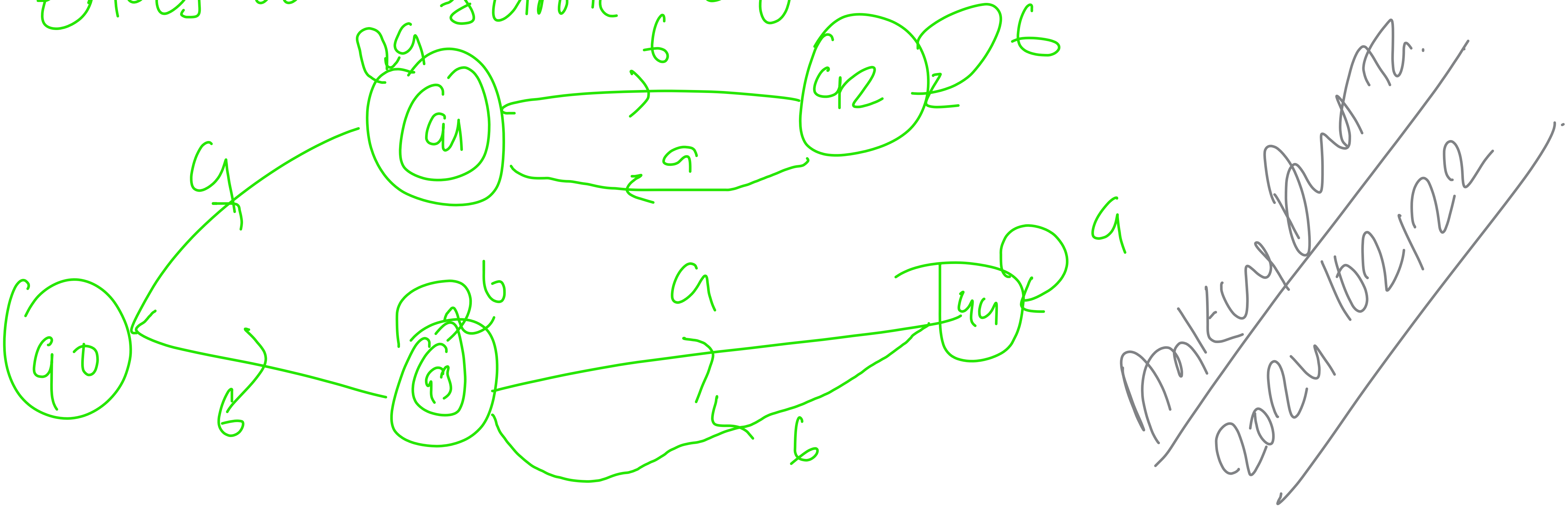
Dikshya
2024/02/22


b
a
a
b
a
q0
q1
q2
q7
q7 b
b

{ length of all strings starts a and ends with a

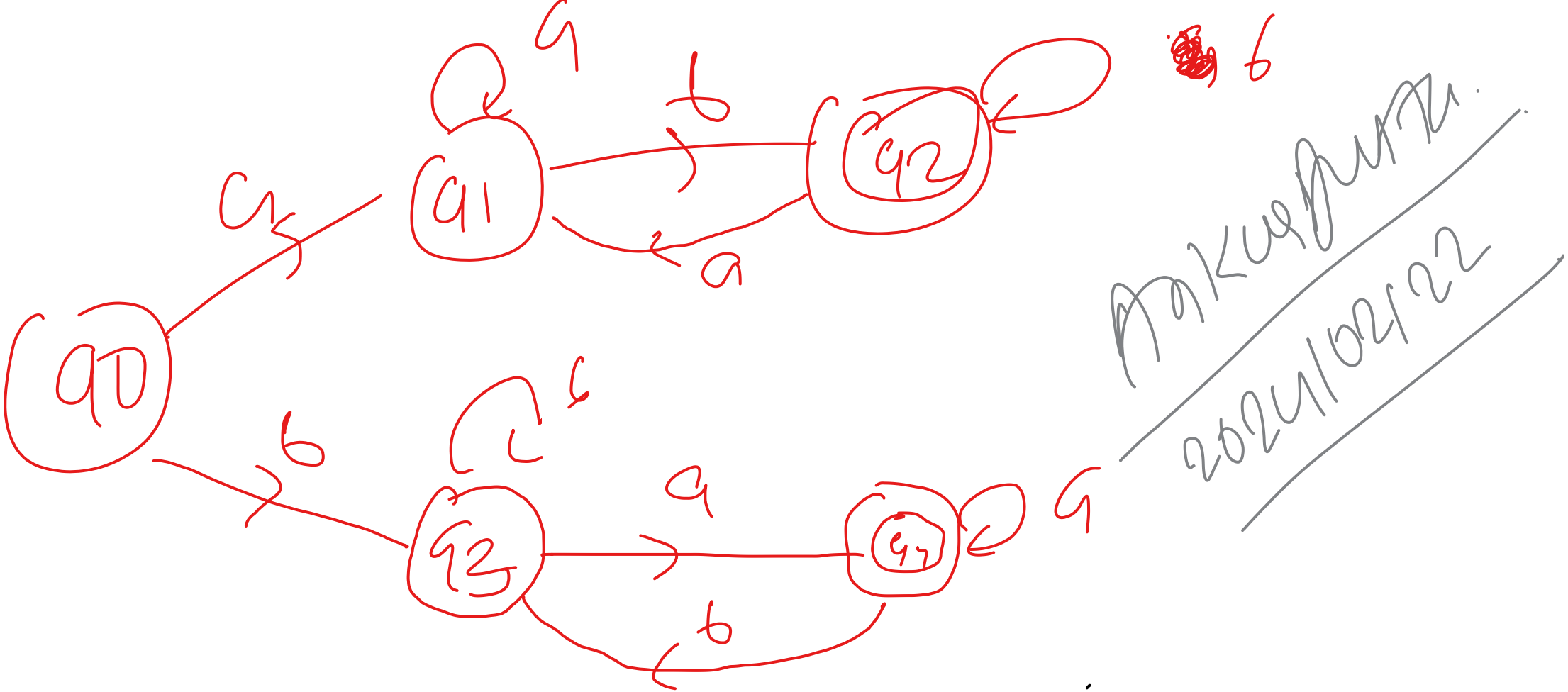
2024/02/22

Length of all strings that starts and ends with same symbol.



Σ = {a, b}

L = Length of all strings that starts and ends with different symbol.

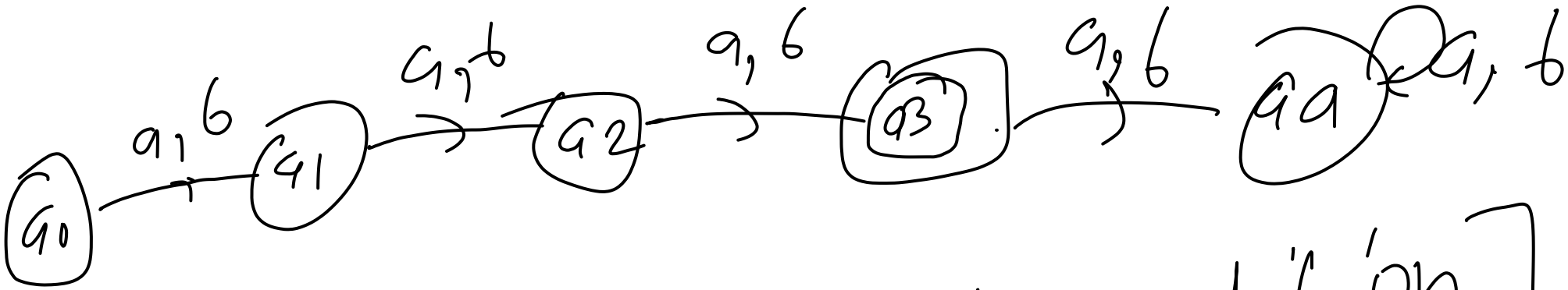


Ankur Gupta
2021/02/22

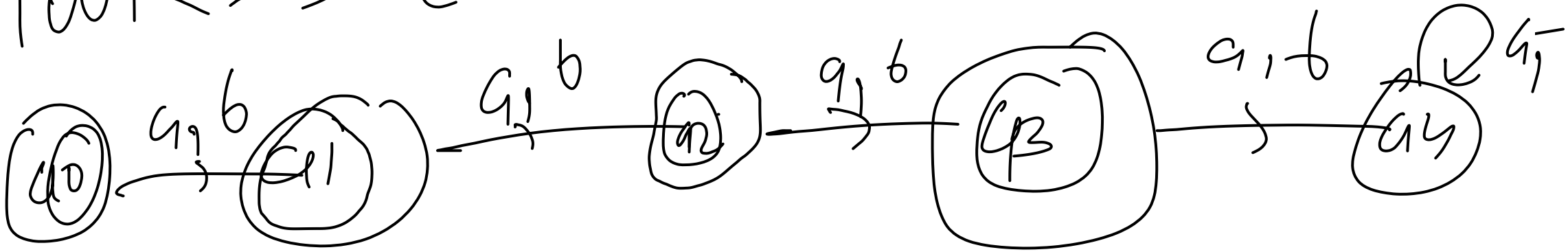
L = Every accepted string must be like

2021/02/12

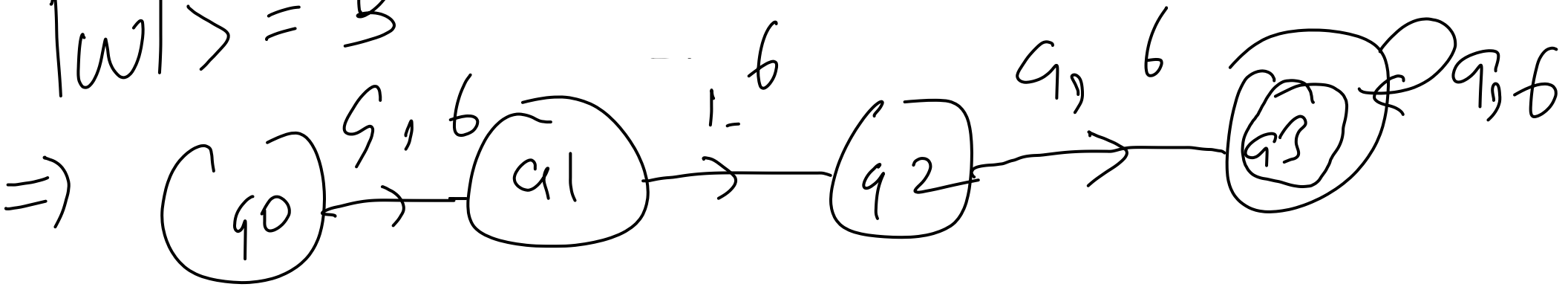
i) |w| = 3



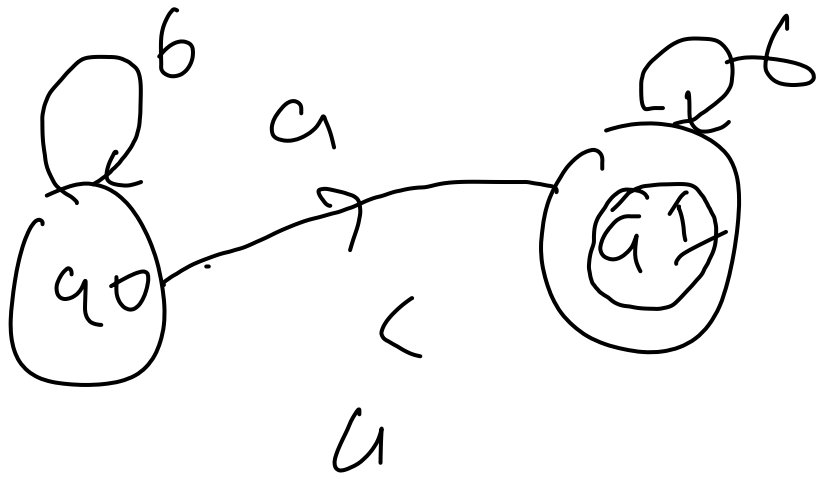
ii) |w| <= 3 [atmost condition]



$|w| >= 3$

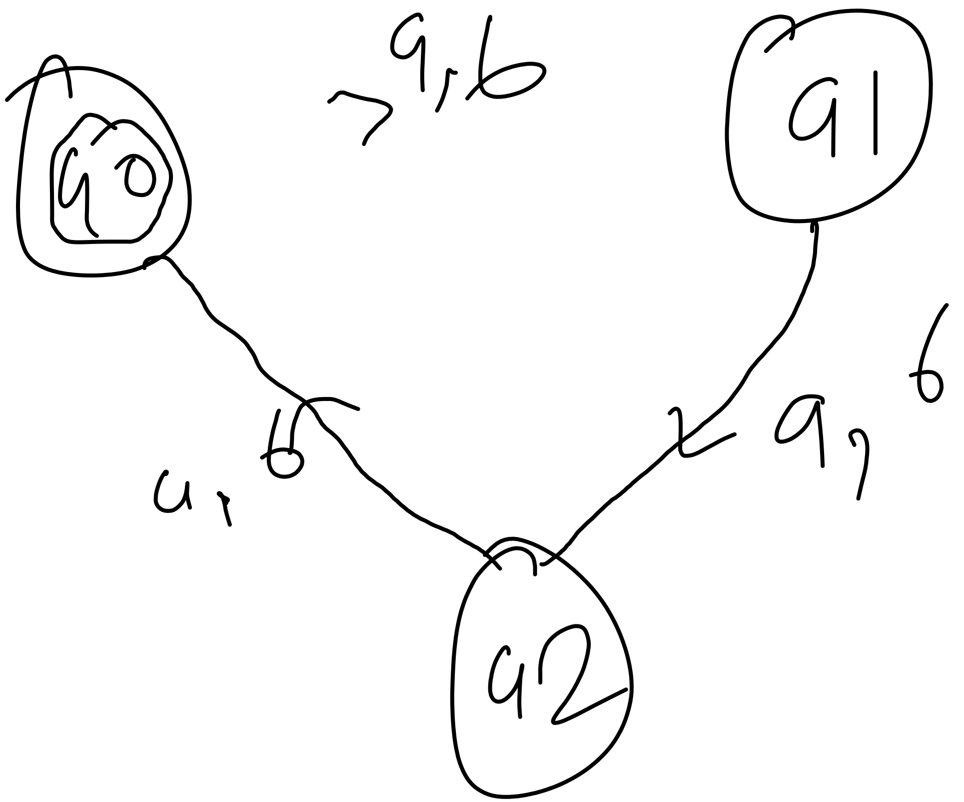


$|w|_a = 2$



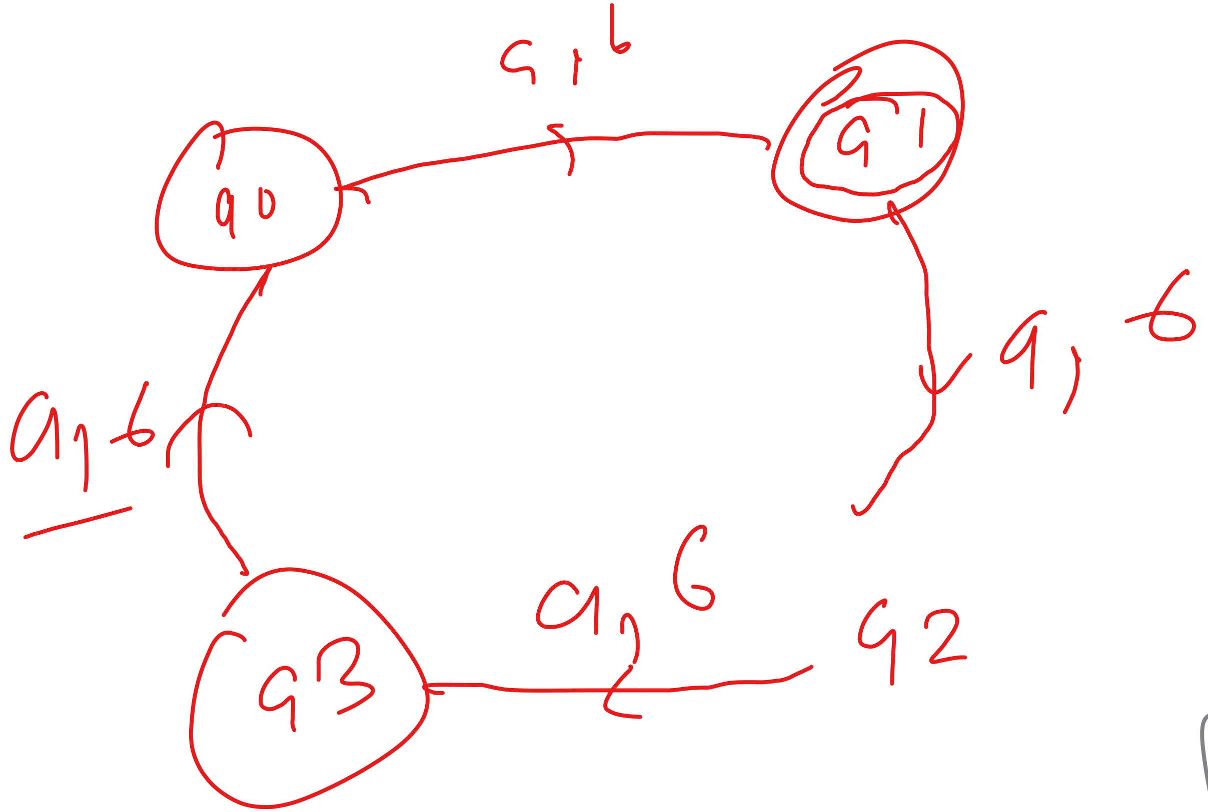
Dokuhara
2024/02/22

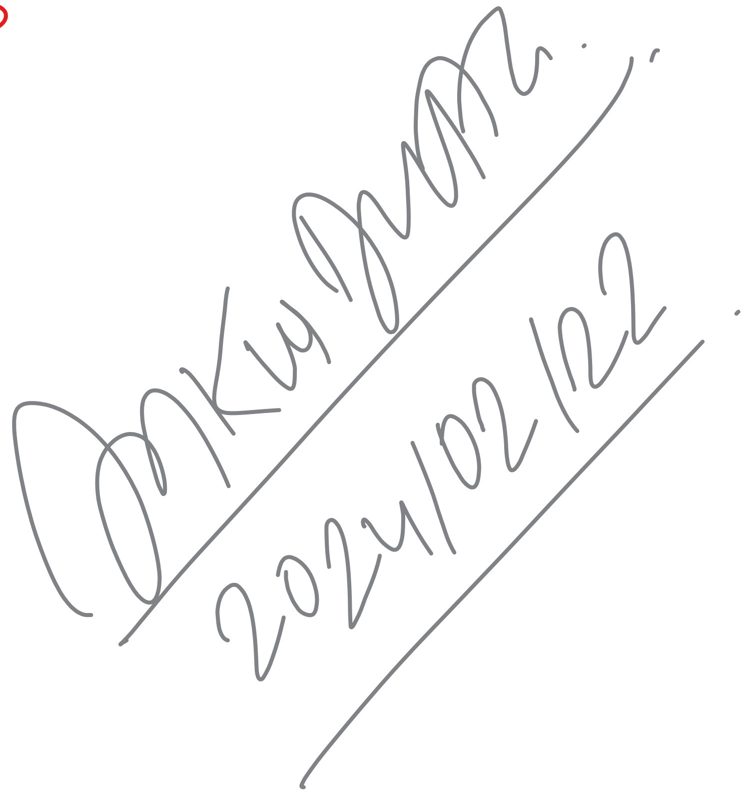
i) $|w| = 0 (\bmod 3)$



ii) $|w| \equiv 1 \pmod 4$

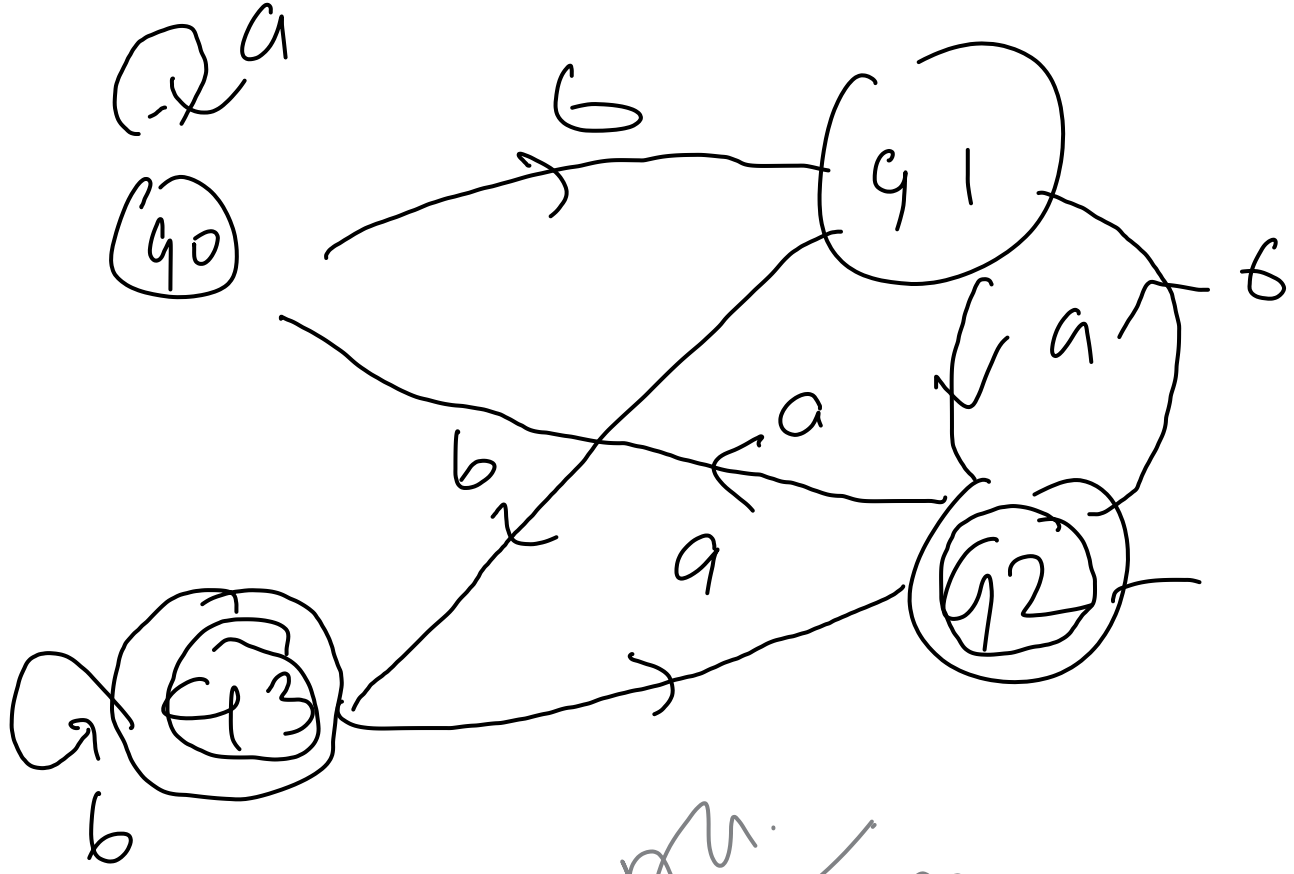
⇒





6) second from right is always 6

	a	6
q0	q0	q1
q1	q2	q3
(q2)	q0	q1
(q3)	q2	q3



Okuyama
2021/02/2

Topics

1. Divide and conquer.
   a) Merge sort (✓) 6 marks
   b) Maxmin (✓) 3 marks
   c) Binary search (✓) 6 marks.

Preparation for 3rd Test

DAA

Ankapur.

2024/02/23

Design Analysis & Algorithm

(1). Sortings: Binary Search
Quick sort
Merge sort
Max-min
etc.

Pankaj Dutta
2024/02/23

let's start with Merge sort because two questions from merge sort were repeated in PYQ's. learn such that you will secure full

(3+3=6) marks in this.

Question (PYQ's 2019-2020)

9) Suppose a[1:m] and b[1:n] both contains sorted elements in non-decreasing order. Write an algorithm 'merge' that merges these items into c[1:m+n] single sorted sequence in O(n) time. is it possible to design an efficient algorithm in comparison to merge?

→ 1. Algorithm Merge8(a, b, c, m, n)
2. // these are elements a[1:m] and b[1:n] both contains sorted elements in non-decreasing order such that a element c[1:m+n] single sorted element
3. // time complexity: O(n)
4. {
5. i:=1, j:=1; k:=1 // define the pointers
6. while(i<=m & j<=n){
7. if(a[i]<= b[j])

Ankur Gupta.
2024/02/23

```
8. {
9.    c[k] = a[i] ;
10.   i = i + 1
11.   else
12.   j = j + 1 ;
13. }
14. k := k + 1
15. Solve the subproblem
16. While (i <= n)
17.   c[k] := a[i];
18.      i := i + 1
19.      j = j + 1
20. }
```

{Bmatrix}

作业

2024/02/23

21. While (j<=n)
22. j = j+1;
23. k<=k+1;
24.}
25.}

Ankur Gupta
2024/02/23

The time complexity remains O(n). Since the time to merge based on the comparison. Hence, it is O(n)

Again repeating the same question:-

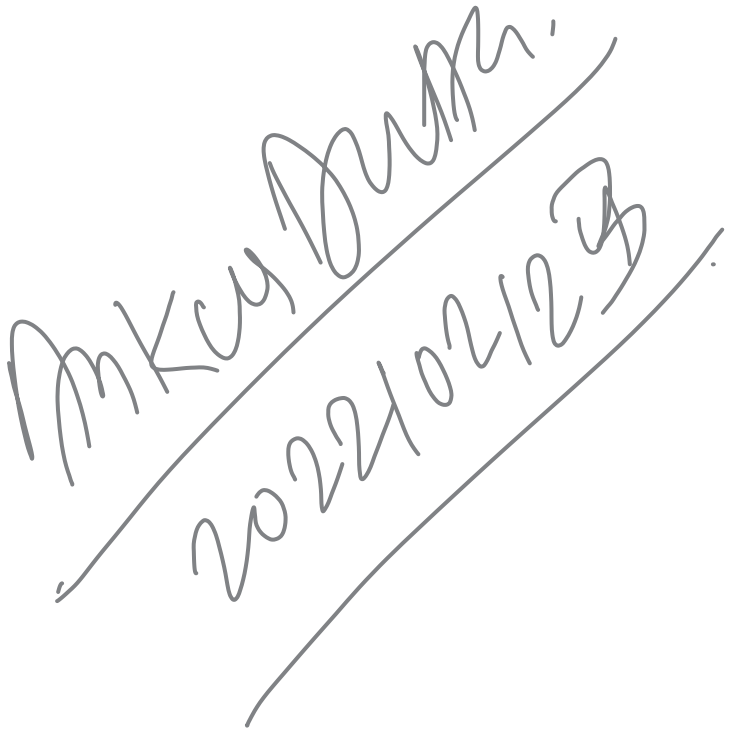
1. Algorithm MergeSort(a, b, c, m, n)
2. // the elements a[1:m] and b[1:n] both contains the sorted elements in non-decreasing order. Now, an element c such that c[1:m+n]

is sorted in single sequences:
3. //time complexity: O(n)
4. {
5. i:=1;j:=1;k:=1 // these are the pointers for a, b & c
6. while(i<=m && i<=n)
7. {
8. if(a[i]=b[j]) then{
9. c[k] = a[i]
i=i+1
10. else
11. j=j+1
12. }
13. k:=k+1
14. Solving the subproblem, we get
15.

Ankur Gupta
2024/02/23

```
16. while (i<=n){
17.     C[K]=A[i]
18.     i=i+1
19.     j=j+1
20. while (j<=n) {
21.     C[K]=B[j]j
22.     j=j+1;
23.     K<=K+1;
24. }
25. }
```

Q.2) from Mergesort



Q/ Consider the following mergesort algorithm. The algorithm first divides the input into four parts A1, A2, A3, A4 instead of two.

⇒ The algorithm:-

```
1. Algorithm Fourwaymerge(low, high, mid)
2. {
3.   IF ( high-low<4)
4.   return sorted array ( low:high+1)
5.   {
6.     mid1 : low+(high-low) /4
7.     mid2 : low+(high-low) /2
8.     mid3 : low+(high-low) /4
                                  {Binary}
```

Ankush Das
10/10/2023

```
9. }
10. //divide the problem into four subproblems. submat A1, A2, A3, A4
11. A1 = mid1( A, low  mid1 );
12. A2 := mid2 (A, mid1+1, mid2) ;
13. A3 := mid3(A, mid2+1, mid3) ;
14. A4 := mid4(A, mid3+1, high ) ;
15. }
16. return joinedImage(A1, A2, A3, A4)
17. }
18. }
```

$T(n) = 4T(n/4) + \theta(n)$

Answered
2021/02/23

Q.7) Write an divide and conquer algorithm "maxmin" to find from a set of n elements? Modify algorithm maxmin so that it works when n is not a power of 2. Is the number of comparision performed by the new algorithm $\lceil 3n/2 - 2 \rceil$ even if n is not a power of 2?

⇒ 1. Algorithm maxmin (i, j, max, min)
2. // write the n elements which are in the list
3. & if (i = j) then
4. max := min := a[i]
5. else if i = j - 1
6. }
7. if [a[i] > a[j])
8. then
9. max := a[i] ; min := a[j]

Ankur Kumar
May 02/23

```
10. else
11.   min := a[j] ; min := a[i]
12. }
13. }
14. //Solve the subproblem:
15. mid := Floor (i+j/2)
16. maxmin (i, mid, max, min)
17. maxmin (mid+1, j, max, min1)
18. }
19. }
```

[5 marks]

2024/02/23

Most important for exam point of view

Binary Search

Ankur Gupta
02/02/23

1) Recursive Algorithm question

Keys problem x1 and x2

→ 1. Algorithm BinSrchTwoKeys(a, i, l, x)
2. // there are n no. of elements x1 and x2 and there are two no. of keys k[x1] and k[x2]
3). // determine the position of k1 & k2 such that x1 = a[k1] and x2 = a[k2]
4. {
5. if (l=i) then
6. {

7. if (k1=a[x1]) then return k1:=i else k1:=0;
8. if (k2=a[x2]) then return k2:=i else k2:=0;
9. return k1, k2.
10. }
11. }
12. Divide the solve problems.
13. mid:=(i+1/2)
14. if(k1=a[x1]) return k1; else if (k1<a[x1]) return Binsrch two keys (a, i, mid-1, 1)
16. if (k2=a[x2]) return k2; else if (k2<a[x2]) return Binsrch two keys (a, 1, i, mid-1

Ankur Gupta
2024/02/23

```
        else
IF(ICI:=0) then return BinSrchTwoKeys(a,l,mid+1,i)
        else
            return (BinSrchTwoKeys(a,l,mid+1,i)
    }
}
```

Mkarpagam
20/10/23

---

ternary search (when it is not divided into two halves instead of 3 halves.

1) Algorithm Ternarysearch(a, i, l, x)
2. //elements are arranged in non-decreasing order.
3. determine whether x is present if not return j such that
   x = a[j]

```
4. if (l >= i)
15. {
16. onethird := i + (l - i / 3) // 3 ;
17. twothird = i + ( l - i/3)//3 ;
18. if (x = a[onethird]) return onethird;
19. if (x = a[twothird] ) return twothird;
20. if (x < a[onethird] then
      return BinSrch(a, i, onethird - 1, l)
    else if x > a[twothird] then
      return BinSrch( a, i, twothird + 1 ., x)
      else
```

Ankur Bhatt
2024/02/23

return BinSrch(a, i, middle + 1, x)

}

}

---

smallest largest index

1. Algorithm BinSearch (low, high, mid)
2.
3.
4. {
5. low := 1, high := n
6. result := 0
7. while (low < high) then

(Mikyung)
2024/02/22

```
8.  if (x<a[mid])
9.      high=low-1;
10. else if (x>a[mid])
11.     low:=high+1;
12. }
13. else result:= mid;
        high:= mid-1
14.
15. }
16. }
17. }  return result;
```

xi) equal sizes into 3 halves.

Q.5 Upto now; 15 marks.

Ankur Gupta
2021/02/23

8) Differences and define deterministic algorithm and non-deterministic algorithm?

→ The differences bet^n deterministic and non-deterministic algorithm are:

Deterministic	Non-deterministic
→ The algorithm which covers a single path from input to output.	→ The algorithm which covers a single path stemming into multiple instruction.
→ It allows only one output instruction.	→ It allows multiple instructions

Ankur Dutta
2024/02/23

primality testing:

```
1. Algorithm Primality (I)
2. //_ ---- — ——-
3. for i = 2 to t do
4. {
5.     n = rand(p, 0.5
6.     j = mod( ) n-1
7.     if(j mod n = 0) then return True.
8.     else
9.     return False
10. }
```

Mukundan
22/02/23

Q.no.14) Revision

→ The algorithm which takes random input to decide what to do next is called Randomized algorithm.

Randomized Rsearch algorithm

1. Algorithm Rsearch(a, n, x)
2. // search the elements from a[1:n] from the set of non-decreasing order.
3. For i := 2 to t do
4. While (true) do

Ankur Gupta
2021/02/23

```
5. i := random() mod n+1
6. if (a[i] = x)

7. return x
8. }
9. }
```

---

control abstraction for divide & conquer Algorithm

```
1. Algorithm DAC(control, input)
2. {
3.     if basecase(input)
           return correctbasecasevalue;
```

Ankur Gupta.
2024/02/23

```
4. // divide
5. subproblems = divide (input)
6. // conquer
7. solution = divide & conquer (control, subproblem)
8. solution.append(solution)
9. result = combine(solutions)
10. return result;
11. }
```

Ankur Gupta.
2024/02/23

How to write in proper way?

→ (1) Binary search which searches for two keys.

1. Algorithm Binsrchtwo keys (a, i, j, x)
2. // Given an array a[I:n] of elements in non-decreasing

Michael Mc. 2024/02/23 --- Read, it will require more time, we need to also practice strassen's multiplication

Revision:

1 → Binary search
- keys (i=)
- not divided two by 3 (i<=)
- → largest index
- → equal & divided by 3 halves.

incase of binary search only condition is

x > a[mid] (x < a[mid]

OR

x = a[one mid]

x = b[two mid]

Anupama
2024/01/23

merge sort: → a[1:m], b[1:n], c[1:m+n]

← A1, A2, A3, A4. $2T(N-1) + cN$.

$O(K \cdot 4^K)$

---

max min - 1 Question. $T(n) = 2T(n/2) + n$

i =>

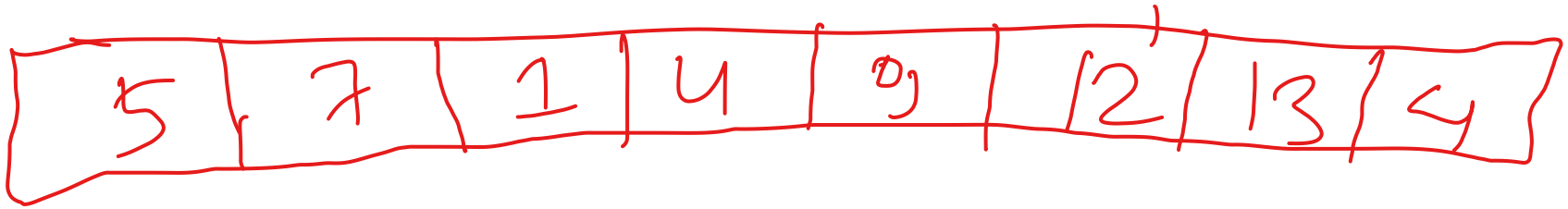
a[i] = a[ ]

Defective chessboard problem

$2^k \times 2^k$ where $k \geq 0$

$4t(n-1)+c$.

Quick Sort



Pivot = 12

5 7 1 4 9 12 13 14

Choose and sort using Quicksort procedure

1. Algorithm QuickSort (arr, low, high)

2. int Pindex = partition (arr, low, high)

int pivot = arr(low)

low := 0

high := n

while (low ≤ pivot) && (~~high~~ low ≤ left)

left + 1

Ankur Dutta
2024/02/23

{
high

while(high ≥ pivot) && (high ≥ right)

right --;

}

{

swap (pivot, left)

return (left)

}

} From smallest to largest using Quick-Select

Ankur Dutta.

2024/02/23

Strassen's multiplication

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}, C = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

b_{11} a_{11} a_{22} a_{22}

P : $(a_{11} + a_{22}) \times (b_{11} + b_{22})$

Q : $b_{11} (a_{21} + a_{22})$

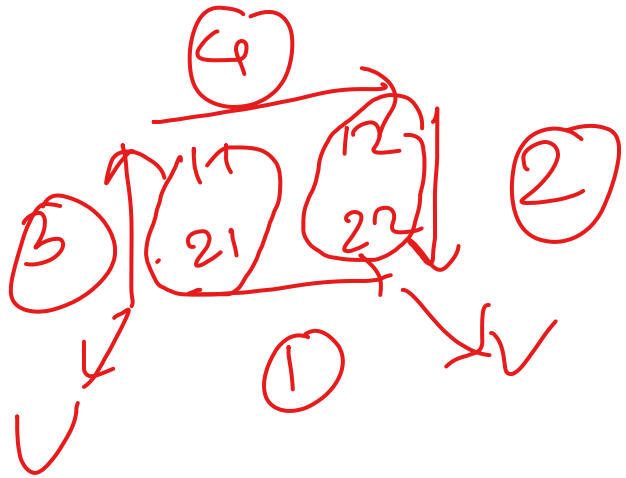
R : $a_{11} (b_{12} - b_{22})$

S : $a_{22} (a_{21} - a_{11})$

T : $a_{22} (b_{11} - b_{12})$

U : $b_{11} + b_{12} (a_{21} - a_{11})$

V : $b_{21} + b_{22} (a_{12} - a_{22})$



Ankur Mukherjee
2024/02/23

$$C_{11} = R + S - T + V$$

$$C_{12} = R + T$$

$$C_{13} = Q + S$$

$$C_{14} = P + R - Q + V$$

2024/02/23

nxn

Ankush Gupta
2024/02/23

a_0 a_1

b_0 b_1

a_0b_1 a_1b_1

a_0b_1 a_1b_0

$+\ a_0b_0$ a_1b_0

a_0b_0 $a_0b_1+a_1b_0$ a_1b_1

$\times$

$x_1 = a_0b_0$

$x_2 = a_1b_1$

$x_3 = a_0b_1 + a_1b_0$

$(a_0+a_1) \times (b_0+b_1) = x_3$

Now,

$x_3 - x_2 - 1$

Time $T(n) = 3T(n/2) + n^2$

$O(n^2)$

Binary search: $T(n/2) + c$ ✓

Merge sort: $2T(n-1) + cN$ ✓

Quicksort $= 2T(N/2) + N$ ✓

worst case $= 2T(n-1) + n$

Maxmin $= 2T(N/2) + 2$ ✓

Ankur Dutta.
2024/02/23

Towers of Hanoi:

$2T(n-1) - 1$ ✗ ✓ $O(2^K)$ by master's theorem

Chessboard

$4T(n-1) + c$

$O(4^k)$

✓

Master's theorem

Ankur Dutta
2024/02/23

Master's Theorem

$T(N) = aT(N/b) + bN$

Now,

$\log_b a > k \quad n^k$

$\log_b a > k \quad n^{\log_b a}$

$\log_b a < k \quad n^{k \log b}$

$\int_a^b f(x)dx = \int_{x_1}^{x_1+h} f(x)dx + \int_{x_1+h}^{x_2} f(x)dx$

Lecture 2024/02/23

$$|f(z)-f(z_0)|$$

$$< \varepsilon.$$

Ankur Gupta.
2024/02/23

Database Management System

for

Database Engineering

Created
2024/02/23
Friday

PYQ's 2019 - 2020

Make sure you prepare the gate exam questions and most importantly the question taught in the class

2019-2020

Q.1) Answer all the questions

Consider a relation schema, A(Id, Name, Age), B(id, Name, Age) & C(Id, Phone, Area) with instances as .....

→ Solution

Ankur Pant
2024/02/23

Data: Collection of raw facts and information:

information: stored data!

Database: collection of interrelated data and information which can be easily managed, accessed and utilized is known as database.

Types of database

Traditional d.b

→ Data stored in the form of text or numbers.

multimedia

→ Data stored in the form of images and video is called multimedia database.

Data warehouse :

→ A huge and special type of data that needs special technique for retrieval is known as datawarehouse.

→ It stores temporal data across a timeline

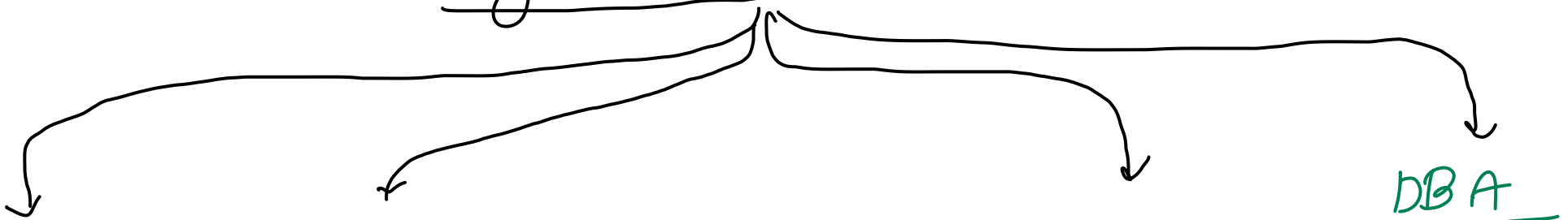
Data Mining :

→ Data mining is a process of analyzing the dense volume of data to find pattern, such trends, and gain insights into how that data can be used is known as data mining.

Database:

→ Connolly and Begg defines the database as, "Database is a software system that enables the users to define, control, manage and access the database. For e.g: MySQL etc.

Types of DBMS users



(i) Naive Users:

→ Those users having no knowledge of DBMS system or any system supporting their usage is called DBMS.

(ii) Online Users

→ Those users who uses the database via any online terminal is known as online users.

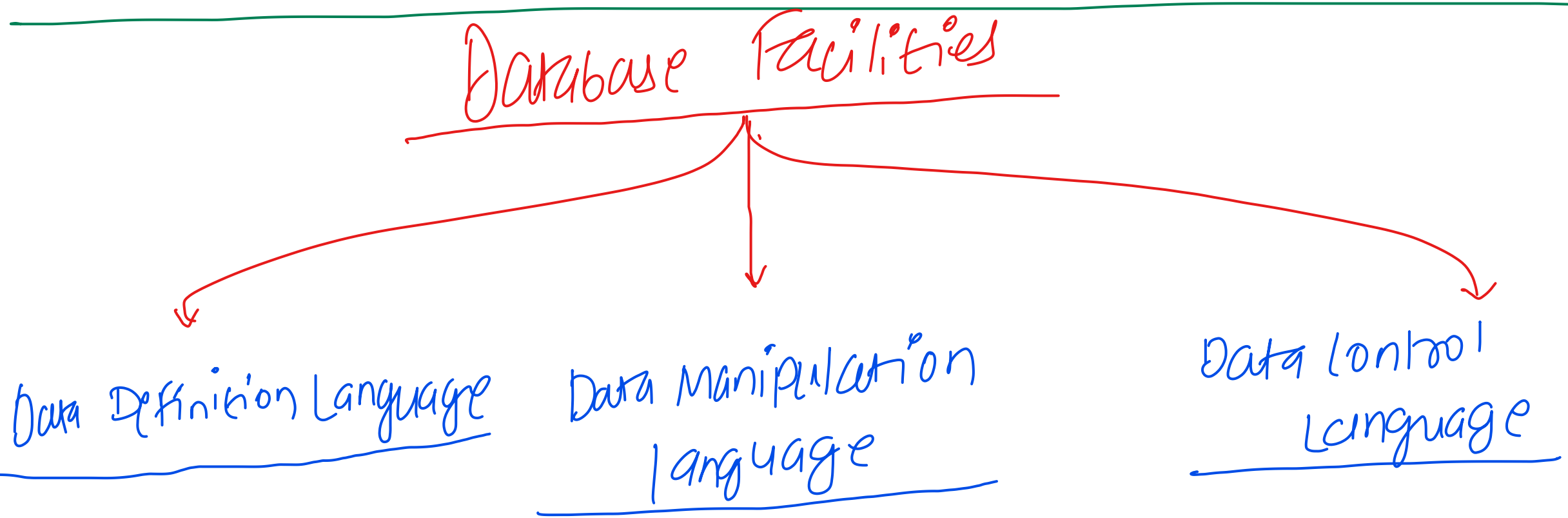
(iii) Application Programmers:

→ Professional programmers who are responsible for developing database system or manage the system program in database is known as application Programmers.

(iv) Database Administrator:

→ A person or a group of person who is responsible for controlling, maintaining and make the smooth run of a database is known as database administrator.

Database Facilities



(I) Data defination Language:

→ It is used to define and alter the structure of a database such as Create, Drop, truncate, rename etc.

(II) Data Manipulation Language:

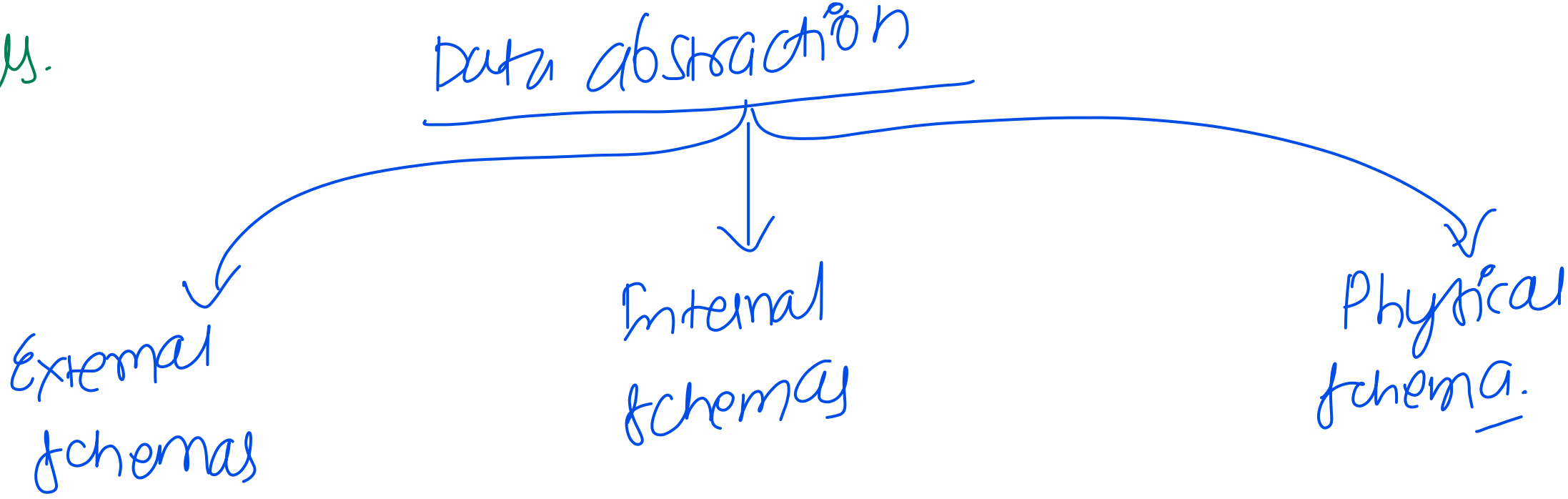
→ It allows to manipulate the database using select, insert, delete, merge etc. functionalities.

(III) Data Control Language:

→ It allows to grant the permission and return the permission or the control user can have / given. For e.g: Grant and Revoke are the two database commands used in data control language.

Data Abstraction and its level.

→ The process of hiding the complexity and only showing essential features to the user is known as data abstraction. It has three levels.



Use the rule of interference

to show that the hypothesis.

"If it doesn't rain or if it is not

foggy, then the sailing race will be held and

the lifesaving demonstration will go on."

"If the sailing race is held then the trophy will be awarded" and "the trophy was not awarded".

Imply the conclusion "it rained".

Soln

let, r be the proposition "it rains",

f be the proposition "it is foggy",

S be the proposition, "the sailing race will be held".

l be the proposition, "the lifesaving demonstration will go on".

therefore,

let it doesnot rain, $\neg r$

let it is not foggy, $\neg f$.

trophy was not represented as $\neg t$.

the premises for the above are:

$(\neg r \vee \neg f) \rightarrow (s \wedge l)$ where

$s \rightarrow t$ for $\neg t$

① By hypothesis

$\neg t$

(ii) By hypothesis

$$\frac{s \rightarrow t}{}$$

(iii) By modus tollens

$$\frac{\neg t \qquad s \rightarrow t}{\neg s}$$

From above,

$(\neg r \vee \neg t) \rightarrow (s \wedge t)$

By contrapositive of

$\neg r \vee \neg t$

$s \wedge t$

$(\neg(s \wedge t) \rightarrow \neg(\neg r \vee \neg t)$

By Demorgan's law

$$\frac{\neg s \vee \neg t}{(r \wedge t)}$$

$$\frac{\neg s}{\neg s \wedge \neg t} \text{ (Adding } \neg s)$$

By Modus Ponens

$$\frac{\neg s \vee \neg t \qquad r \wedge t}{\neg s \vee \neg t}$$

By simplification

78V74

8n7

1Qs

2024/02/24

(1) Previous year Questions.

Q.no.1)

Solu

A

Id	Name	Age
12	Arun	60
15	Shreya	24
99	Rohit	11

B

Id	Name	Age
15	Shreya	24
25	Hari	40
98	Rohit	20
99	Rohit	11

C

Id	Phone	Area
10	2200	02
99	2100	01

$(A \cup B) \bowtie_{A.id > 40 \vee C.id < 15} C$.

$\Rightarrow [\sigma_{A.id > 40}(A \cup B) \times C] \cup [\overline{(A \cup B)}_{C.id < 15} \times C]$

So $(A \cup B) \times C$ where $A.id > 40$

I.d	Name	Age	Cid	Phone	Area
98.	Rohit	20	10	2200	02
98.	Rohit	20	10	2200	02
99	Rohit	11	99	2100	01
99.	Rohit	11	99	2100	01

Now,

$(A \cup B)_{C.id < 15} \times C$

Id	Name	Age	C.Id	Phone	Area
12	Arun	60	10	2200	02
15	Shreya	24	16	2200	02
25	Hari	40	16	2200	02
98	Rohit	20	10	2206	02
99	Rohit	11	10	2200	02

Now, from U :

Id	Name	Age	L.id	Phone	Area
12.	Arun	60	10	2200	02
15.	Shreya	24	10	2200	02
25.	Hari	40	10	2200	02
98.	Rohit	20	10	2200	02
98.	Rohit	11	99	2100	01
99.	Rohit	20	10	2200	02
99.	Rohit	11	99	2100	01

In total, there are 7 tuples in the table.

(b) Given A and C where A is the primary key and C is the Foreign key.

A	C
2	4
3	4
4	3
5	2
7	2
4	5
6	4

(3,4), (4,3) & (6,4) ✗

On-delete (cascading)

(9)

→ Candidate keys of R:

$A \rightarrow B$

$BC \rightarrow D$

$E \rightarrow C$

$D \rightarrow A$

How to find?

2024/02/23

DELHI PUBLIC SCHOOL.

full scholarship in
B. Optometry in Pokhara University.

1/b) Ans

A	C
~~2~~	~~4~~
3	4
4	3
~~5~~	~~2~~
~~7~~	~~2~~
~~9~~	~~5~~
6	4

When tuple (2,4) is deleted (5,2), (7,2) & (9,5) are additionally deleted.

---

d) A → B, BC → D, E → C, D → A

A^+ = ABCDEA

How many month ... worked in a company

→ Select DateDiff (Month, Date of Join, Get Date())

From tablename where emp_name = " ... "

Soundness & Completeness.

how to find candidate keys? shortest tricks done with Jenny's lectures

$X=1, Y=1$

X	Y
1	
2	
3	
4	
5	
6	
7	

$X=2,\ 2\times1+1 = 3$

$X=3,\ 2\times3+1 = 7$

$X=4,\ 2\times7+1 = 15$

$X=5,\ 2\times15+1 = 31$

$X=6,\ 2\times31+1 = 63$

$X=7,\ 2\times63+1 = 127$ ✓

(3)

X	Y	Z
1	4	2
1	5	3
1	6	3
3	2	2

$XZ \rightarrow Y$, $Y \rightarrow X$, $X \rightarrow Z$

Soundness: the axiom in which all the function dependency f belong to the set of Functional dependency F

Completeness: the axiom which consists repeatedly functional dependency of F.D.F which is extracted from FD Ft.

Question

Check if $AC \rightarrow E$

$F = \{A \rightarrow B, C \rightarrow D, BD \rightarrow E, AC \rightarrow E\}$

$f = \{A \rightarrow B, C \rightarrow D, BD \rightarrow E\}$

$P1 = f - f$

$Y = \{AC \rightarrow E\}$

$T_i = \{AC\}$

$A \subseteq T_i(AC)$

$T_i = ABC$

$C \subseteq T_i(ABC)$

$T_i = ABCD$

$BD \subseteq T_i(ABCD)$ ✓

$T_i = ABCDE$

✓

Steps	G	Is G = ∅?	T_i	Is $\in$ G = T_i?
1.	$A \to B, C \to D, BD \to E$	No	AC	No
2.	$C \to D, BD \to E$	No	ABC	No
3	$BD \to E$	No	ABCD	Yes
			ABCDE	

Since, $AC \to E$ is redundant. proved.

Functional Dependency

If $X \rightarrow Y$, $Y \subseteq X$ such that if $T_1(x) = T_2(x)$ then $T_1(y) = T_2(y)$

Armstrong axioms

- Reflexive
- Transitive
- Augmentation
- Union
- Pseudotransitive
- Decomposition

Prove Union

Given: $X \to Y$ and $X \to Z$

$X \to XY$ and $XX \to YZ$ (Augmentation of X on $X \to Y$ and Y on $X \to Z$.

Now,

$X \to XY$

$XY \to YZ$ By transitivity on (i) & (ii)

$X \to YZ$

proved.

(1) Prove pseudotransitive:

Given :- $X \to Y$ and $YW \to Z$

~~$YW \to Z$~~

$XW \to YW$ and $YW \to Z$ $\leftarrow$ Augmentation of W on $X \to Y$

proved $XW \to Z$

(3) Prove decomposition:

Given : $X \to YZ$

We know, $YZ \to Y$ and $YZ \to Z$.

How to check decomposition?