"श्री"

Languages

Language:- A system suitable for the expression of certain ideas, facts or concepts including a set of symbols and rules for their implem-entation.

Formal Languages:-

1) **Symbol:-** A "Symbol" is an abstract or a user defined entity. Letters and digits are examples of frequently used symbols. Symbols cannot be formally defined like point in geometry.

eg:- 'a', 'b', 'd' are symbols.

2) **Alphabet:-** Let A denote a non empty set of symbols. Such a set A is called an alphabet. An alphabet need not be finite. But, here we will assume alphabet as a finite. So, an alphabet is a finite set of symbols.

eg:- {a, b, ... z}. {0, 1, 2, ...}

3) **String:-** A "string" over an alphabet is a finite sequence of symbols from that alphabet, which is usually written next to one another not separated by commas.

eg:- if $\Sigma = \{a, b\}$ then $u = abbba$

$v = abbab$ strings on Σ.

and if $\Sigma = \{0,1\}$ then 00110011 is the string over.

Length of the String:-

The length of a string is its length as a sequence. So, the length of the string 'w' is denoted by $|w|$, is the no. of symbols in the string.

$u =$ abbba $\quad |u| = 5$

$v =$ abba $\quad |v| = 4$

The string of zero length is called as "empty string". This is denoted by $\wedge$ or ϵ or λ.

Introduction to defining Language

Suppose,
$\Sigma = \{0\}$

$L_1 = \{0, 00, 000, 0000, \ldots\}$

This could also be written as,

$L_1 = \{0^n : n = 1, 2, 3, \ldots\}$

Concatenation:- In concatenation two strings are written down side by side to form a new longer string.

Some basic properties of concatenation are as follows:-

The set of all strings (including $\wedge$) on the alphabet set Σ is denoted by Σ^* and the set of non-empty strings by Σ^+.

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

$\Sigma^* = \{\Lambda, a, b, ab, aab, abb, aabb \ldots\}$

1) Concatenation on a set Σ^* is associative since for each u, v, w in Σ^*, $u(vw) = (uv)w$.

2) <u>Identity Element</u>:- The empty string is an identity element for the operation i.e

$$\Lambda u = u \Lambda = u \quad \forall \; u \in \Sigma^*.$$

3) Σ^* has left and right cancellation.

for, u, v & w in Σ^*.

$wu = wv \Rightarrow u = v$ (Left cancellation)

$uw = vw \Rightarrow u = v$ (Right cancellation)

4) For, u, v in Σ^*, we have

$$|uv| = |u| + |v|.$$

That is, the length of the concatenation of two string is the sum of the individual length.

5) x^i is concatenated with x^j is the string x^{i+j}.

Substring :-

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

Consider any string $u = 0011010$ on an alphabet Σ.

Any sequence $w = 0110$ is called a substring of u.

Prefix and Suffix :-

Given two words u and v over an alphabet Σ $u = v$ if they have same length and the same symbols at the same positions.
We say u is a prefix of v if there is word w over Σ such that $w = uv$ and we say that u is a proper prefix of v if $u \neq \Lambda$, $u \neq v$ and u is a prefix of v.

That is Λ is always a prefix of any word 'u' including itself and a word u is always prefix of itself and if $w = uv$ the v is the suffix of u.

eg :- $w =$ "0011"

prefixes — Λ, 0, 00, 001, 0011

Suffix — 1, 11, 011, 0011

Reverse of a String:-

If $w = w_1 w_2 \ldots w_n$ where $w_i \in \Sigma$ the reverse of w is $w^R = w_n w_{n-1} \ldots w_1$

If 'w' be a word in language L, then reverse(w) is the same string of letters traversed backward called the reverse of w, even if this backward string is not a word in language L.

eg1. reverse $(a_1 a_2 a_3 a_4 a_5) = a_5 a_4 a_3 a_2 a_1$

reverse (2361) = 1632

Transpose is also called reverse operation.

* Palindrome:-

Palindrome of a string is a property of a string in which string can be ~~used~~ read same from left to right as well as from right to left.

* Lexicographic Ordering:-

The lexicographic ordering of a string is the same as the dictionary ordering, except that shorter strings precede longer strings.

lexicographic ordering {0,1} is

{Λ, 0, 1, 00, 11, 01, 10, 111, 000, 101, 010, 011...}

(V_N, Σ, S, P)

$(Q, \Sigma, \delta, q_0, F)$

Q1: $L_1 = \{x, xy, x^2\}$

$L_2 = \{y^2, xyx\}$

Find (a) $L_1 L_2$ (b) L_2^2 (c) $L_2 L_1$

(a) $L_1 L_2 = \{x, xy, x^2\}\{y^2, xyx\}$

$= \cancel{xy^2, xy^3, x^2y^2, x^2yx, x^3}$

$= xy^2, xy^3, x^2y^2, x^2yx, xyxyx, x^3yx\}$

$L_2 L_1 = \{y^2, xyx\}\{x, xy, x^2\} = \{y^2x, xyx^2, y^2xy, y^2x^2, xyx^2y, xyx^3\}$

(b) $L_2^2 = \{y^2, xyx\}\{y^2, xyx\}$

$= \{y^4, xyxy^2, y^2xyx, xyx^2yx\}$

Q2: let $L = \{ab, aa, baa\}$ which of the following are in L^*

(a) abaabaaabaa (b) aaaabaaaa

(c) baaaaabaaaab (d) baaaaabaa

(a) abaabaaabaa $\in L^*$

(b) aaaabaaaa $\in L^*$

(c) baaaaabaaaab $\notin L^*$

(d) baaaaabaa $\in L^*$

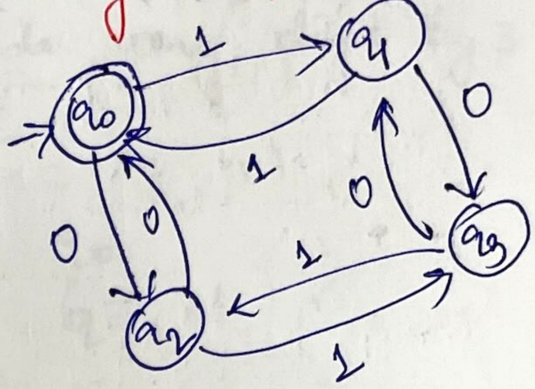
$L = \{a, b\}^*$

$L^* = \{\epsilon, a, b, ab, ba, abb, baa, abba$

Acceptability of a string by FA

Q:- For the FSM, M described by table,

State	Input 0	Input 1
$\rightarrow$ q_0	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2



find the strings among the following strings which are accepted by M:

(a) 101101 ✓ (b) 11111 × (c) 000000 ✓ * (d) 1101011

* (e) 0011011

✓ (f) 11010010

Soln:

$\delta(q_0, 101101) \Rightarrow \delta(q_1, 01101)$

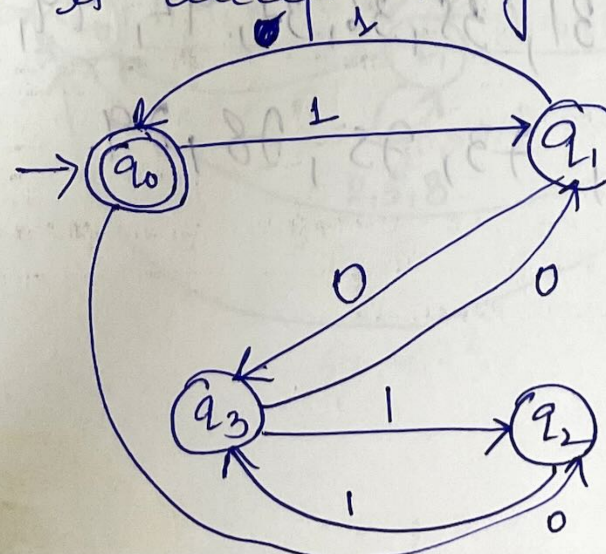
$\Rightarrow \delta(q_3, 1101)$

$\Rightarrow \delta(q_2, 101)$

$\Rightarrow \delta(q_3, 01)$

$\Rightarrow \delta(q_1, 1) \Rightarrow q_0$

q_0 is the final state which we got after processing the provided input. So, the string 101101 is accepted by M.



ii) 11111

$\delta(q_0, 11111) \vdash \delta(q_1, 1111)$

$\vdash \delta(q_0, 111)$

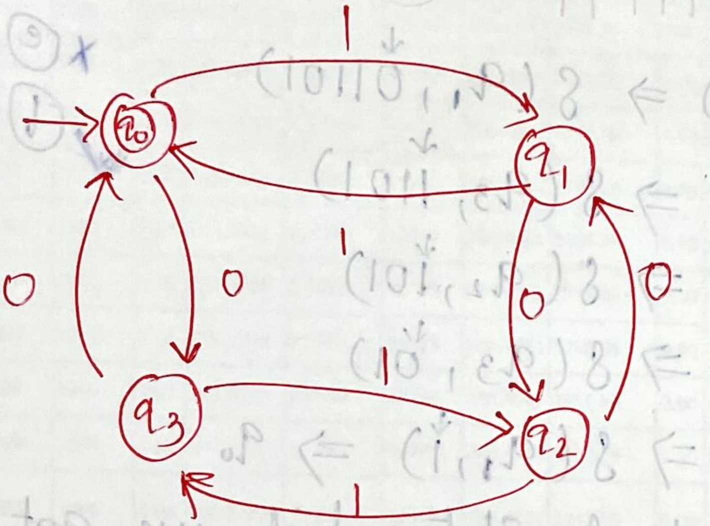
$\vdash \delta(q_1, 11)$

$\vdash \delta(q_0, 1)$

$\vdash q_1$ Non final State

Not acceptable

Q:- Check



~~110110~~

~~1101011~~

(a) 1101011 no

(b) 0011011 no

(c) 11010010 yes

1-80

518

1, 11, 24, 25, 31, 35, 36, 37, 42, 44, 50, 53, 54, 58, 60, 69, 73, 75, 78, 79.

Q: Construct a DFA for accepting the decimal no. which is divisible by 3.

Soln: Always whenever we divide any digit by 3 we get remainders 0, 1 or 2.

For 0, 3, 6, 9 — rem 0 — q_0

1, 4, 7 — rem 1 — q_1

2, 5, 8 — rem 2 ← q_2

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

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

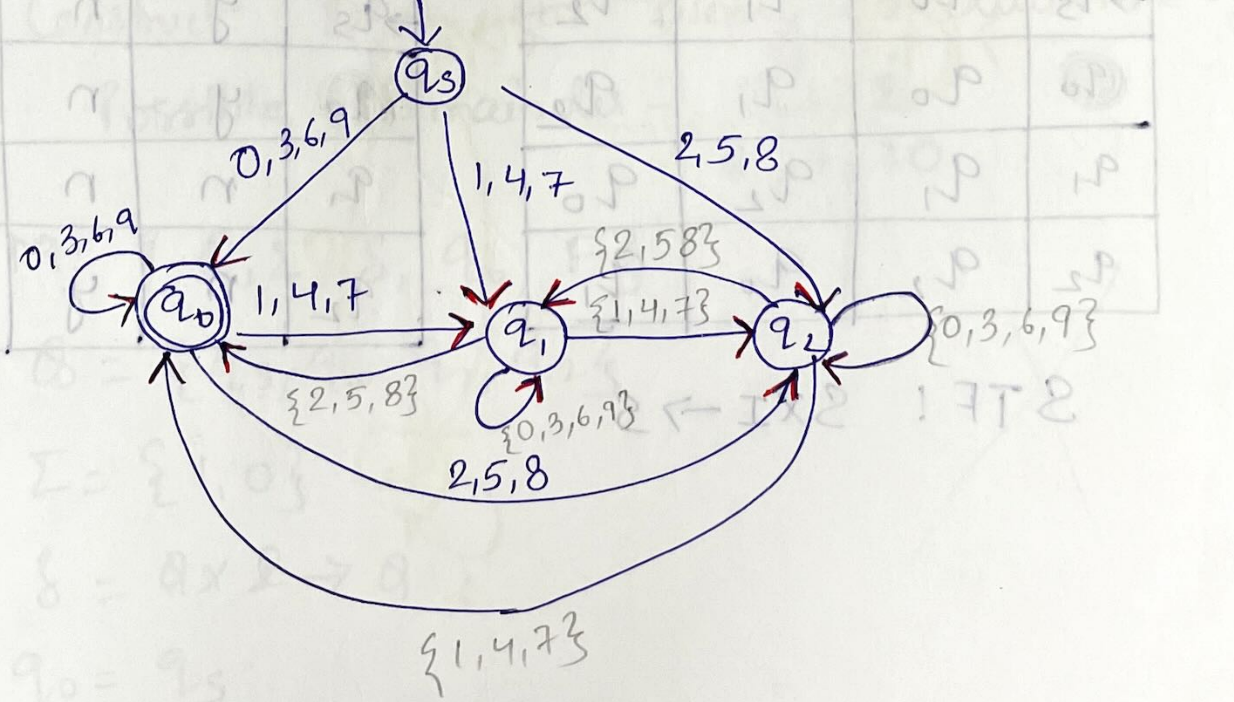
$\Sigma = \{0, 1, \ldots 9\}$

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

$q_0 = q_s$

$F = q_0$

	0	1
$\rightarrow$ * q_0 (00)	q_0	q_1
q_1 (01)	q_2	q_0
q_2 (10)	q_1	q_2



Suppose the no. is 256542

$\delta(q_s, 2) \rightarrow q_2$

$\delta(q_2, 5) \rightarrow q_1$

$\delta(q_1, 6) \rightarrow q_1$

$\delta(q_1, 5) \rightarrow q_0$

$\delta(q_0, 4) \rightarrow q_1$

$\delta(q_1, 2) \rightarrow q_0$

It reaches to final state q_0 so it is divisible by 3.

Transition Table

S \ I	{0,3,6,9}	{1, ~~2~~4,7}	{2,5,8}
$\rightarrow q_s$	q_0	q_1	q_2
q_0	q_0	q_1	q_2
q_1	q_1	q_2	q_0
q_2	q_2	q_0	q_1

S \ I	0,3,6,9	1,4,7	2,5,8
$\rightarrow q_s$	y	n	n
q_0	y	n	n
q_1	n	n	y
q_2	n	y	n

STF : $S \times I \rightarrow S$

Q: Give DFA for accepting a language of string over the alphabet (0,1) such that they contain a substring "1001".

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

$Q = q_s, q_0, q_1, q_2, q_3, q_4$

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

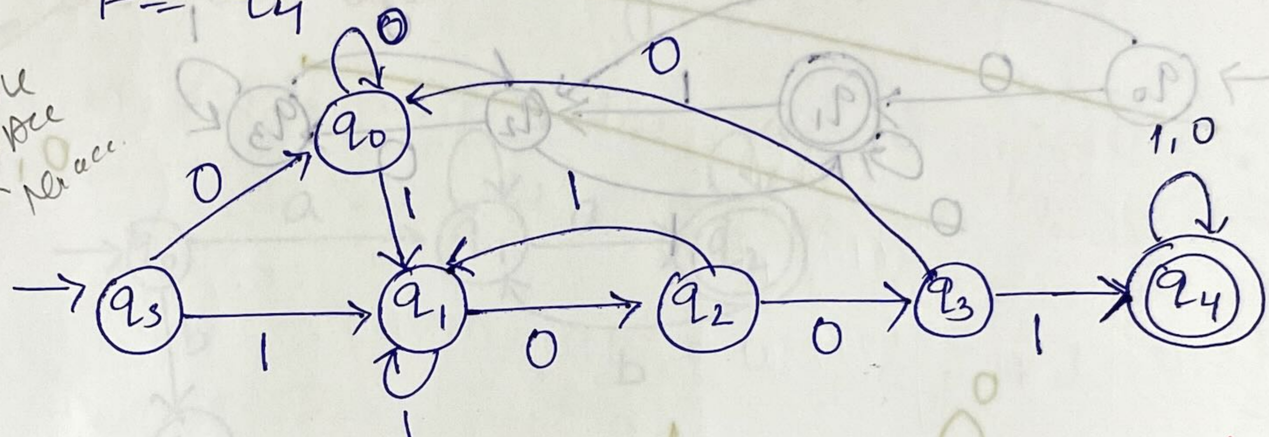
$\delta = Q \times \Sigma \rightarrow Q$

$q_0 = q_s$

$F = q_4$

q_s - Start State
q_0 - end with 0
q_1 - " " 1
q_2 - " " 10
q_3 - " " 100
q_4 - " " 1001

State Dig, Tr table, ℓd < acc, reject



Q: Construct DFA for Binary no. divisible by 3.

Soln: Possible Remainder - 0, 1, 2
0, 1, 10

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

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

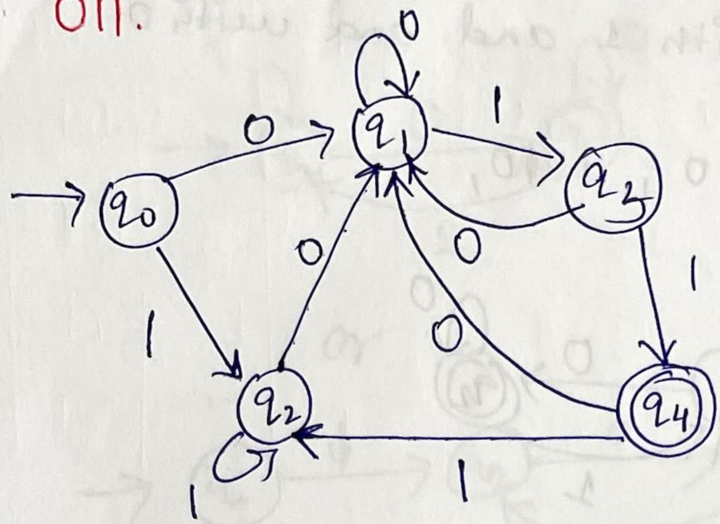
$\Sigma = \{1, 0\}$

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

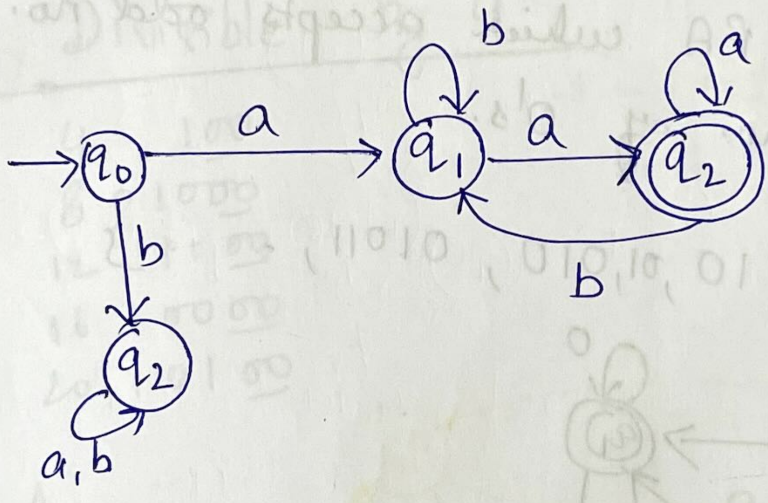
$q_0 = q_s$

$F = q_0$

Q1. DFA to accept string of 0's & 1's ending with 011.

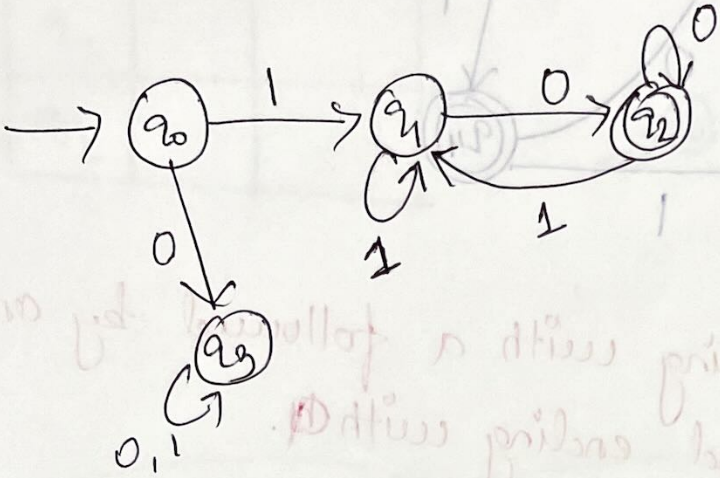


Q: String starting with a followed by any no. a's & b's and ending with a.



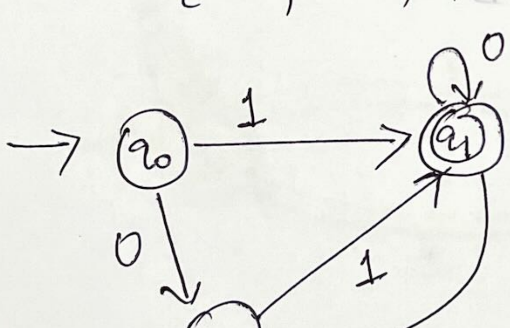
Q1. Design a FA which accept string which starts with 1 and end with 0.

Solⁿ: 10, 010, 110, 1100, ...



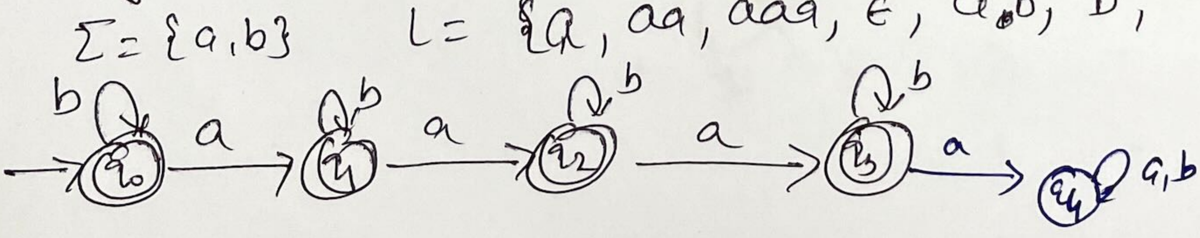
Q: Design an FA which accepts odd no. of 1's and any no. of 0's.

Solⁿ: L = { 1, 10, 01, 010, 01011, ... }

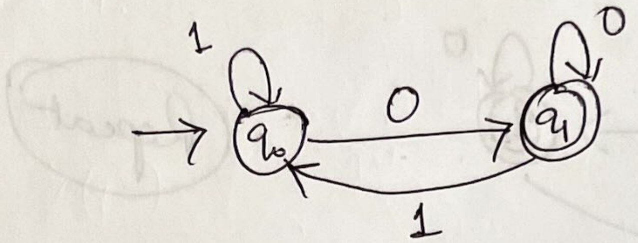


Q: Design an FA which has atmost 3 a's and any no. of b.

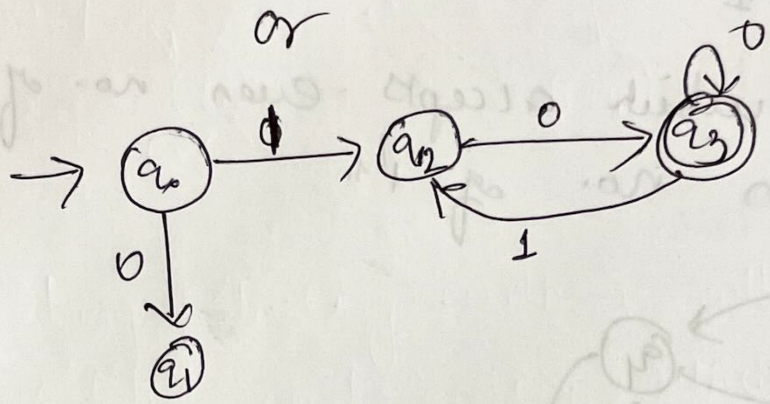
$\Sigma = \{a, b\}$ L = {a, aa, aaa, ϵ, ab, b,



Divisible by 2

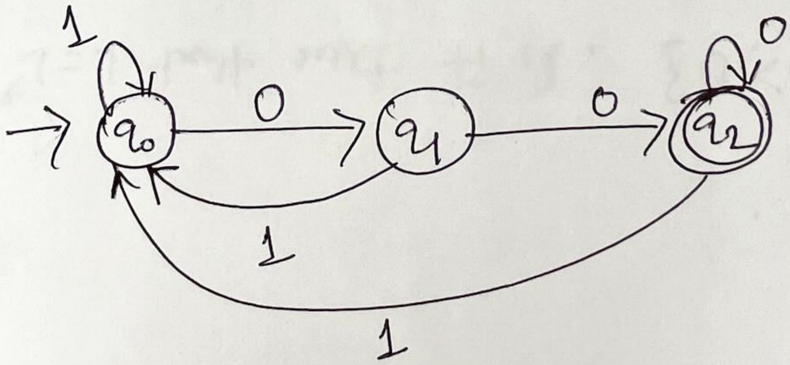


or

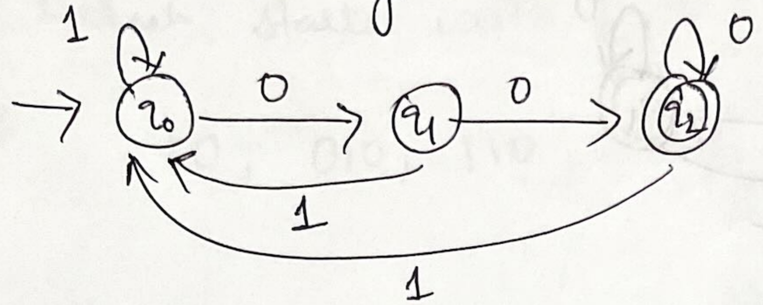


Divisible by 4

4	1<u>00</u>
8	10<u>00</u>
12	11<u>00</u>
16	100<u>00</u>
20	101<u>00</u>

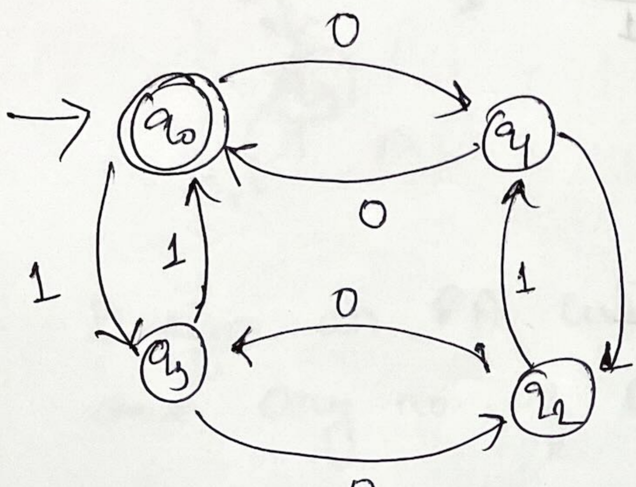


Q: Design an FA to accept the string that always ends with 00.



Repeat

Q:- Design FA which accepts even no. of 0's and even no. of 1's.



q0

q1

q2

q3

$n=0$ $m=1 \Rightarrow b$
$n=1$ $m=1 \Rightarrow ab$

Q1:-

a) $L = a^n b^m$; $n \geq 0$ $m <= 4$

b) Odd no. of 1's followed by an even no. of 0's.
100, 10000, 1110...

c) String over 0 & 1, no. of 0's divisible by 3 and no. of 1's divisible by 5.
00011111
000000111111111

Q2: a) Write strings over a and b such that the fourth symbol from right end is a and fifth symbol is b?
.... b a a a a
or or or
b b b

b) Strings of a's and b's containing no more than 3 a's.
b, ab, ba, a, aab, baa, aba ...

c) which are the strings of a's and b's whose length is divisible by 3?
aba, aaa, baa, abb ...

d) Write the set of all strings over a and b containing symbol a at second position from the right end?
.... a a a
or
b

e) Write strings over 0 & 1 with a substring 010.
010, 0010, 00101, 001010, ...

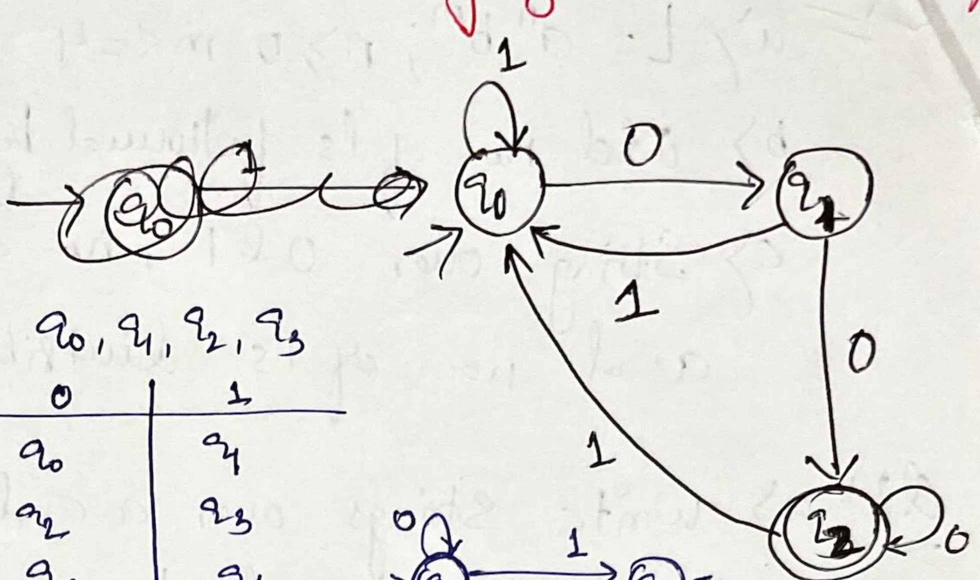
f) Set of all strings over 0 & 1 whose decimal no. is not divisible by 5.

g) " decimal no. divisible by 5.

Q: Design an automaton that recognizes numbers divisible by 4.

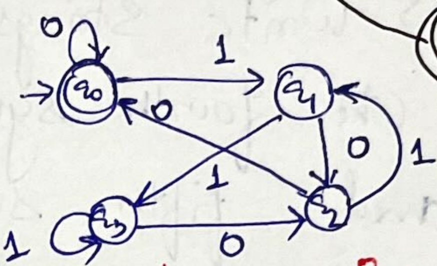
Soln:

4	100
8	1000
12	1100
16	10000
......	



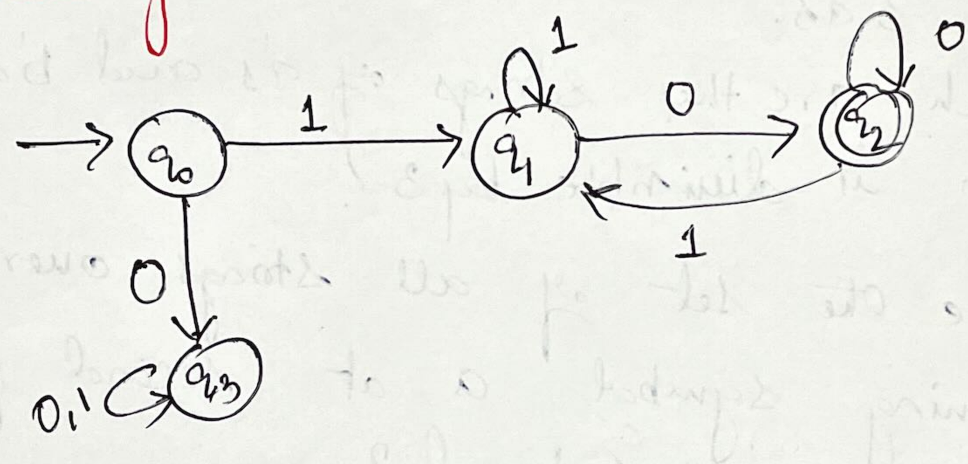
q_0, q_1, q_2, q_3

	0	1
$\rightarrow {}^*q_0(00)$	q_0	q_1
$q_1(01)$	q_2	q_3
$q_2(10)$	q_0	q_1
$q_3(11)$	q_2	q_3



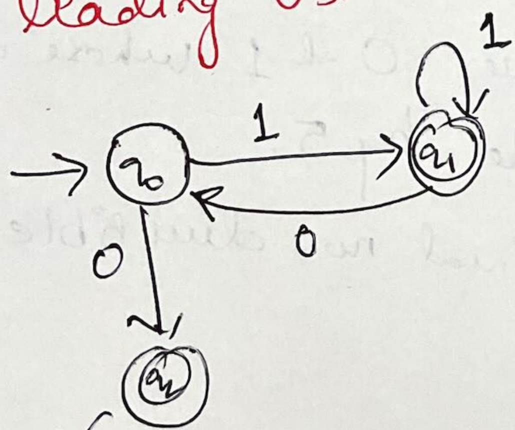
Q: Design a DFA for even numbers without leading 0's.

Soln:



Q: Design a DFA for odd numbers without leading 0's.

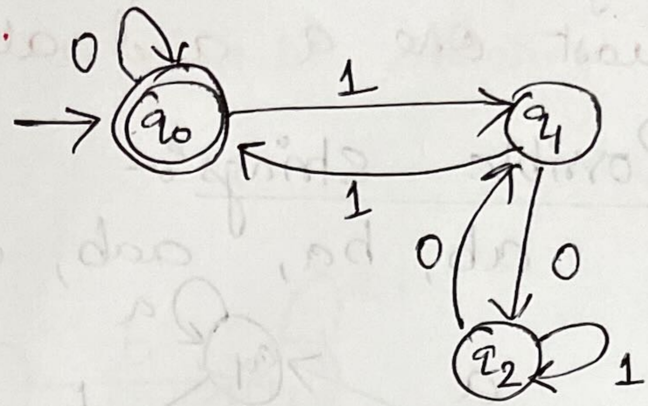
Soln:



Q: Design a DFA that accepts all binary numbers divisible by 3.

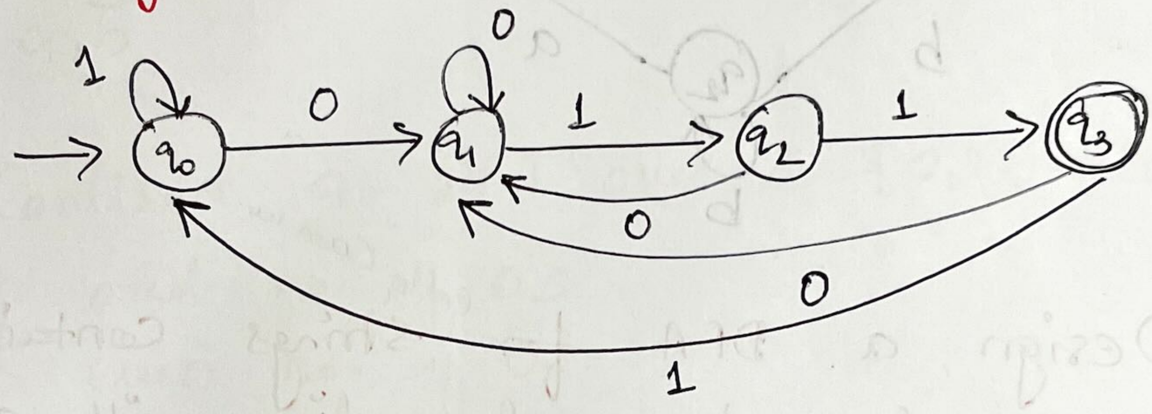
Sol^n:-

3 - 11
6 - 110
9 - 1001
12 - 1100
- - - -



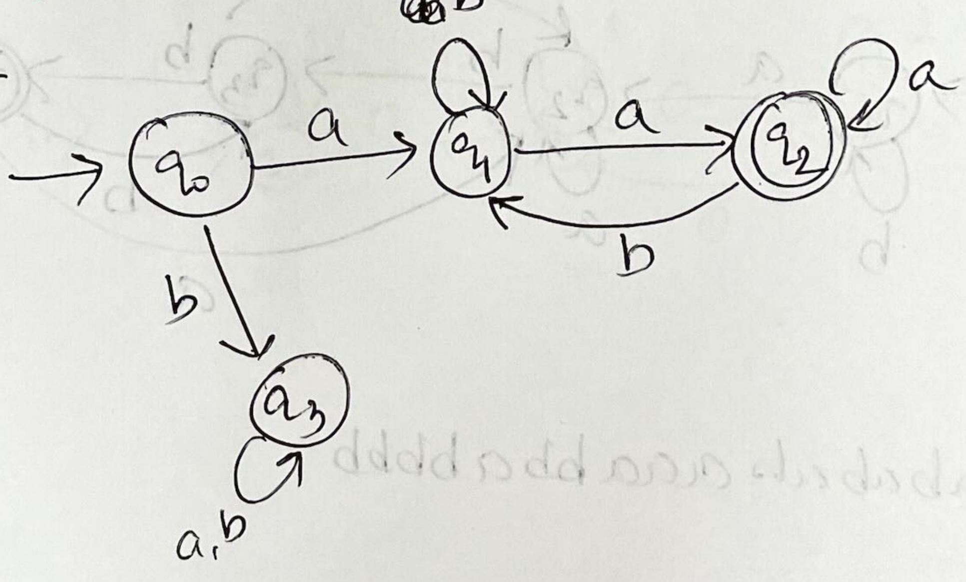
Q: Design a DFA to accept strings of 0's and 1's ending with 011.

Sol^n:-



Q:- String starting with a followed by any no. of a's and b's and ending with a.

Sol^n:-

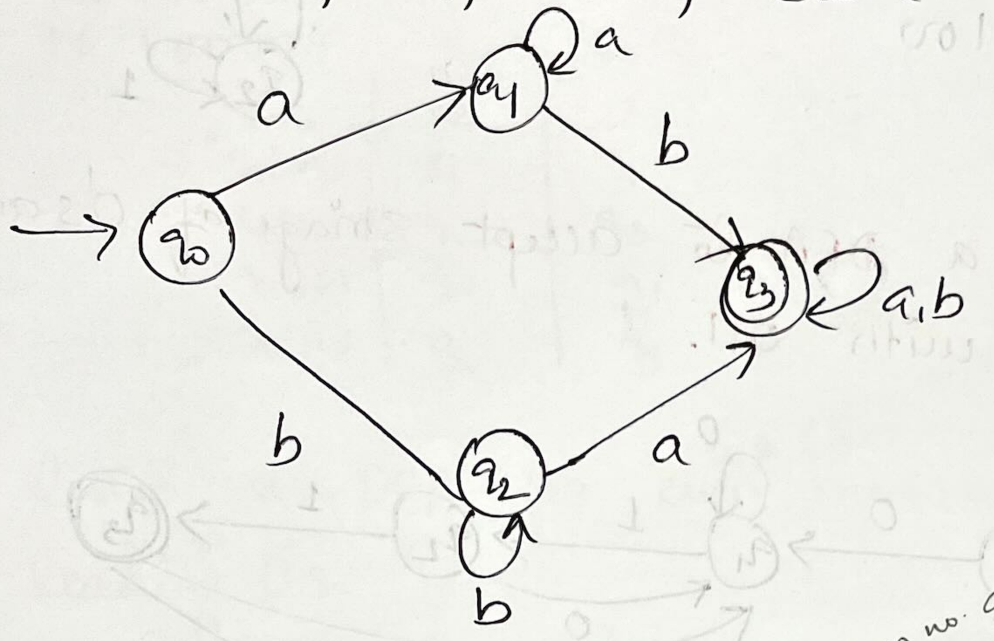


A - 8
SB - 4
AJP - 4
BT - 3
15/4
18

Q:- Design a DFA for strings Containing atleast one a and atleast one b.

Solⁿ:- Possible strings:-

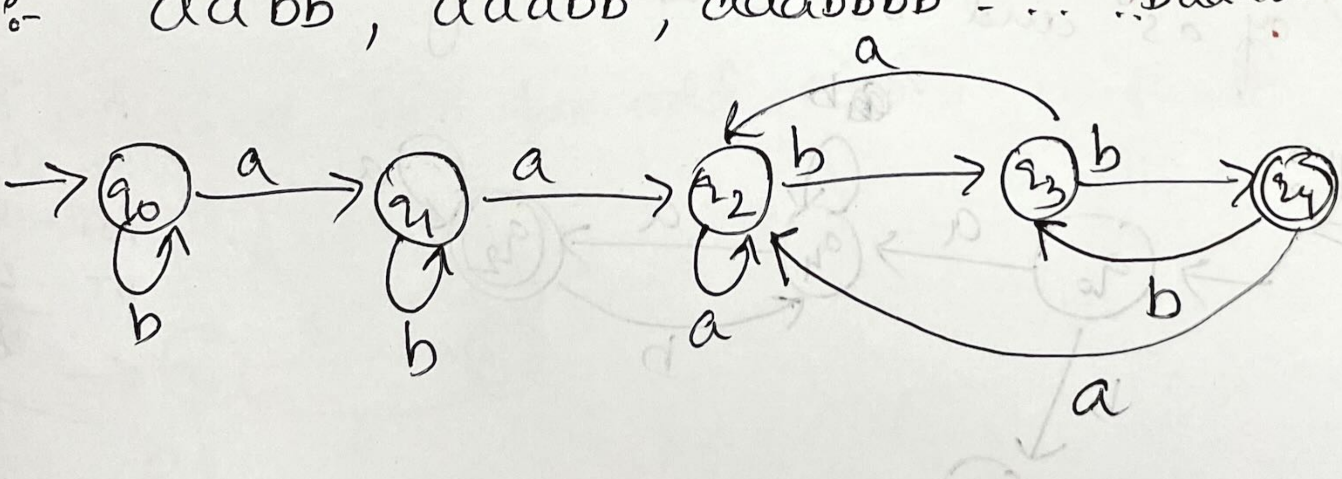
ab, ba, aab, aaba



Q:- Design a DFA for strings containing atleast two a's and ending with an even no. of b's.

ending even no. a

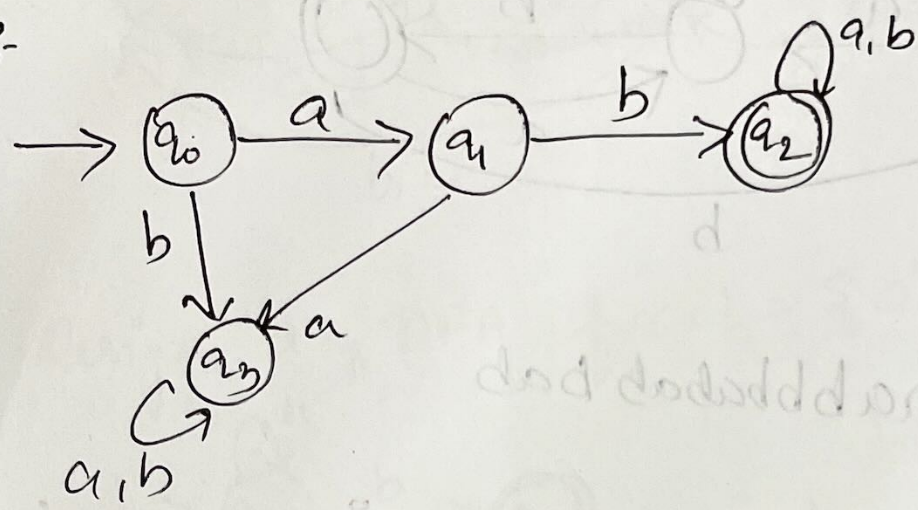
Solⁿ:- aabb, aaabb, aaabbbb ...... baaabbbb



eg:- abababaaabbabbbb

Q:- Design a DFA which accepts the set of all strings on $\Sigma = \{a, b\}$ starting with the prefix ab.

Soln:-

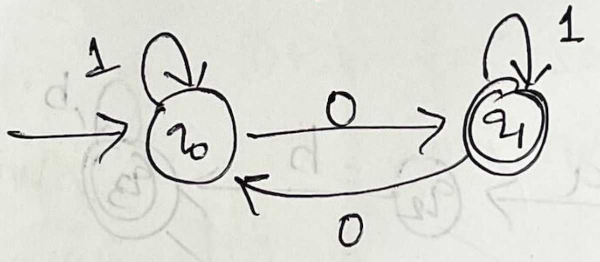


Q:- Construct a DFA over $\Sigma = \{0, 1\}$ accepting

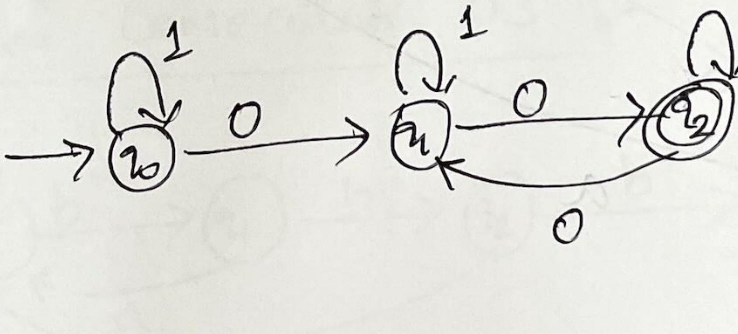
i) odd no. of 0's

ii) even no. of 0's.

Soln:- i)

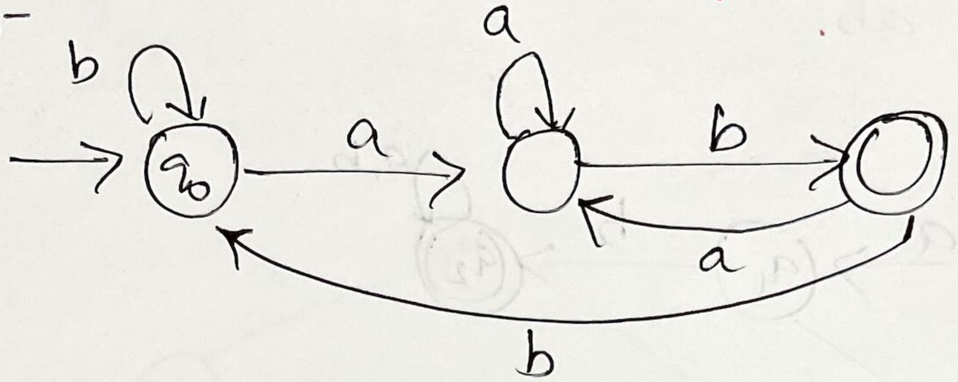


ii)



Q:- Design a DFA accepting all string over $\Sigma = \{a, b\}$ ending in ab.

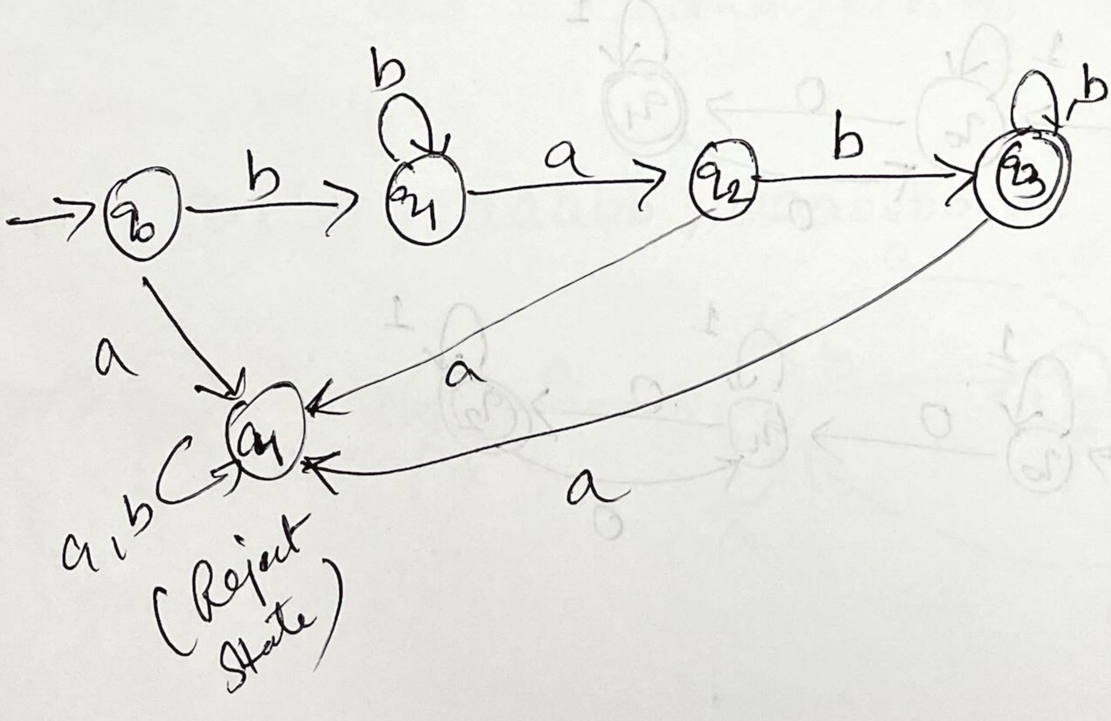
Sol^n:-



eg:- aabbbababbab

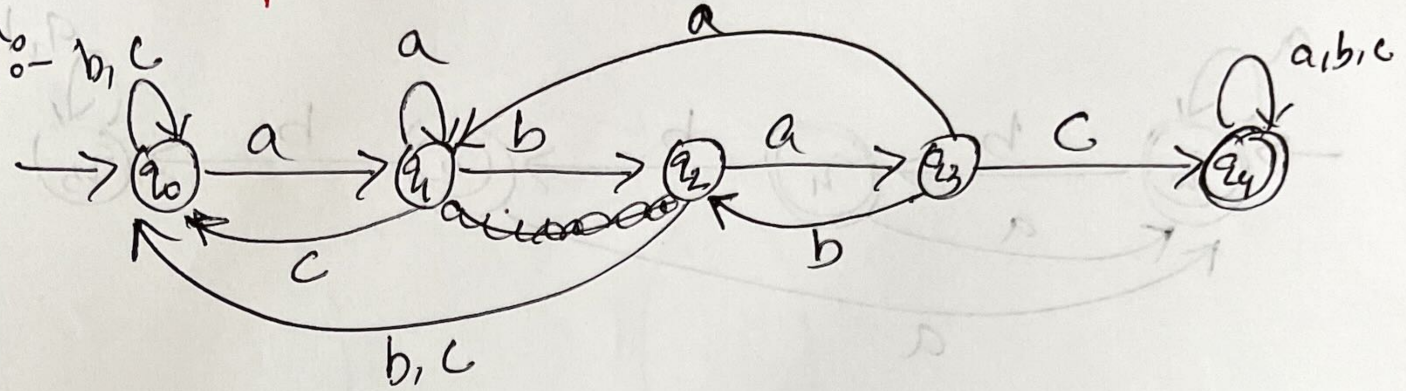
Q:- Construct a DFA which will recognize the language $L = \{b^m a b^n : m, n > 0\}$.

Sol^n:- L = bab, bbab, babb, bbbabbb ....



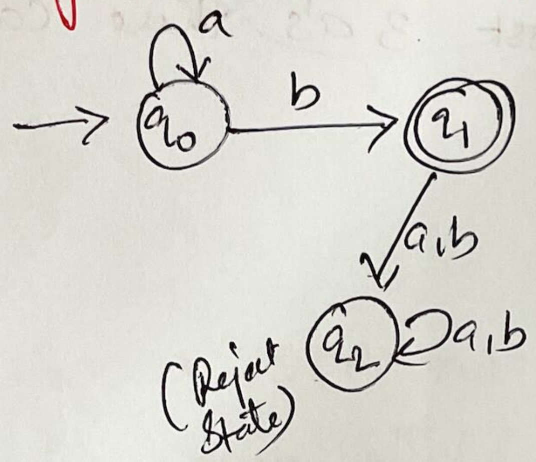
Q:- Design a DFA for Strings Containing the pattern abac. over $\Sigma = \{a, b, c\}$

Solⁿ:-



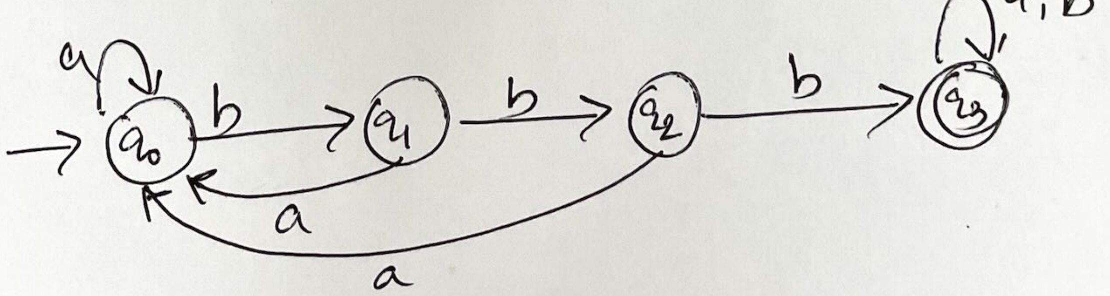
Q:- Design a DFA for $L = \{a^n b : n \geq 0\}$

Solⁿ:-

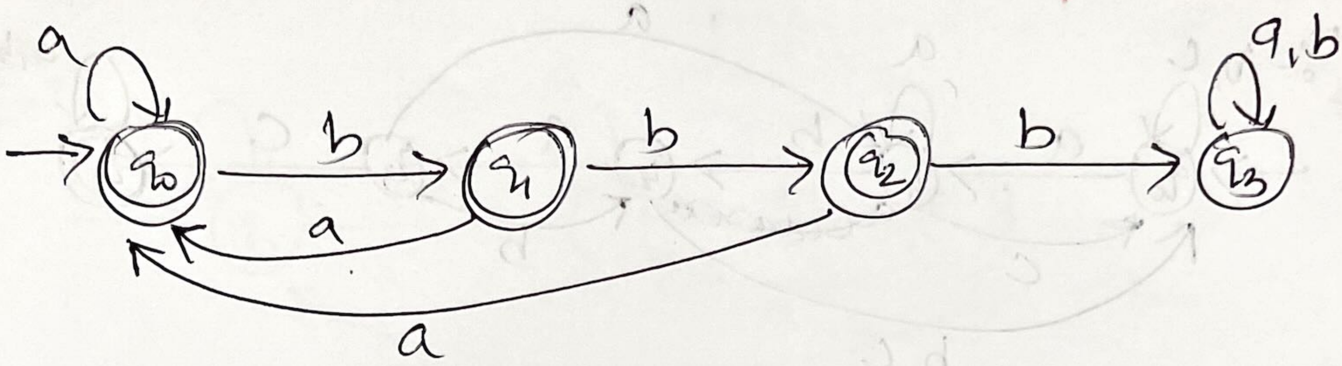


Q:- Design a DFA for Set of strings over $\Sigma = \{a, b\}$ which does not contain three Consecutive b's.

Solⁿ:- Three Consecutive b's :-



Not Containing three Consecutive b's

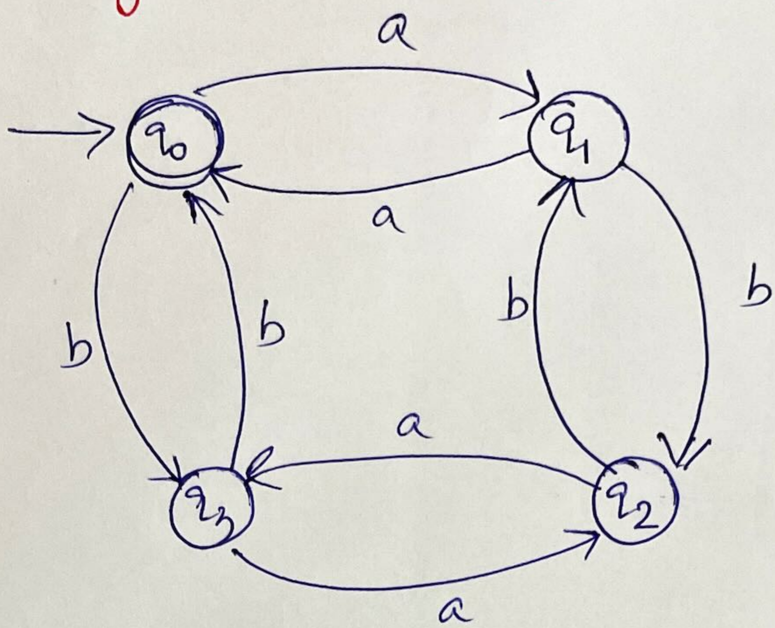


Q:- Design a DFA with input symbols a and b that contain atmost 3 a's. There can be any no. of b's.

Sol^n:-

Q:- Design a DFA over the input symbol $\Sigma = \{a, b\}$ having even no. of a and b.

Soln:-



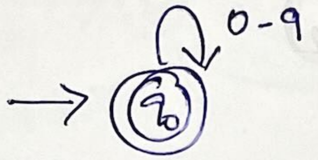
Q:- Design a DFA over the input symbol $\Sigma = \{a, b\}$ having ~~even~~ i) odd no. of a and even no. of b.
ii) even no. of a and odd no. of b.

Soln:- i) odd no. of a and even no.

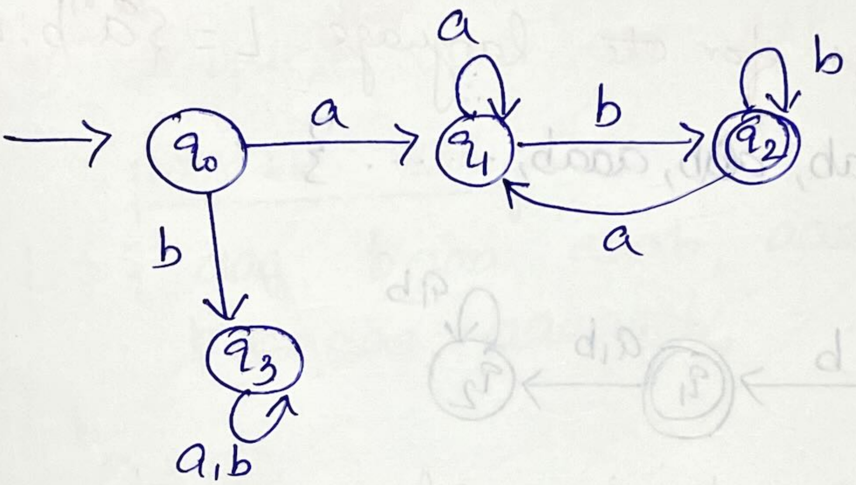
Q:- Determine the DFA with the

a) Set ~~of~~ integers

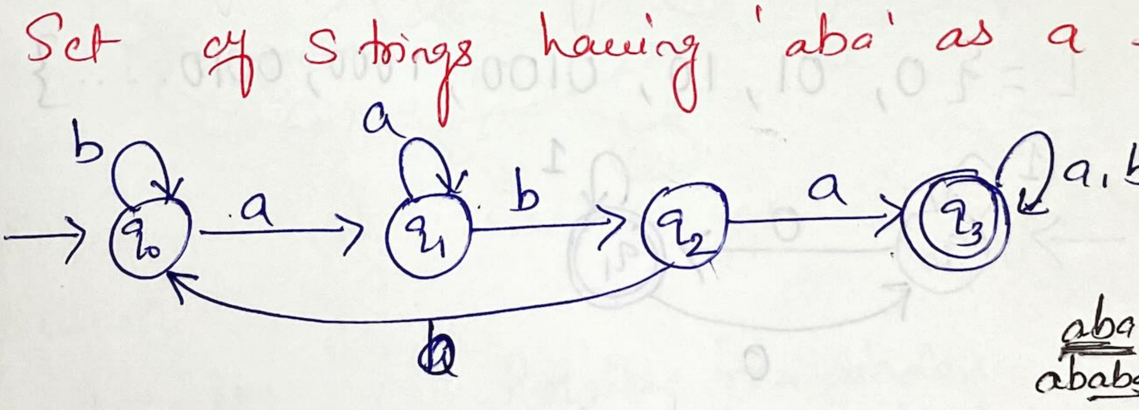
$\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$



b) Set of strings beginning with 'a' and ending with 'b'.

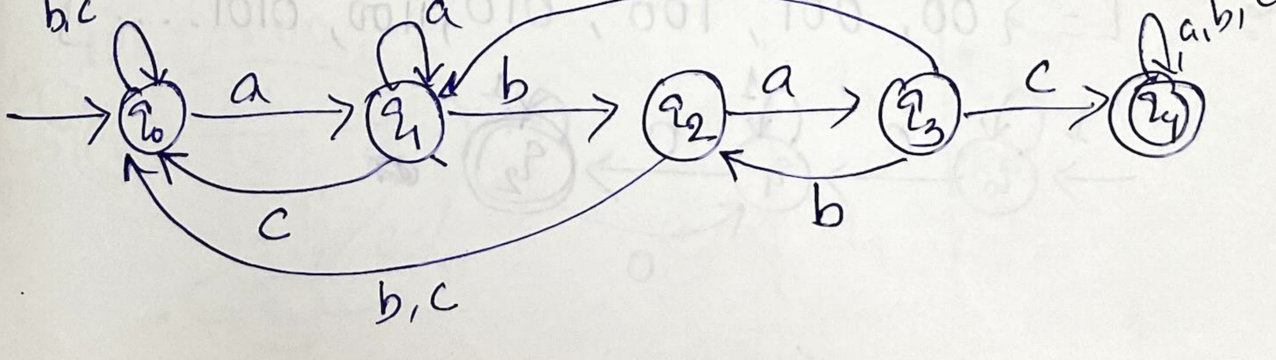


c) Set of strings having 'aba' as a subword.



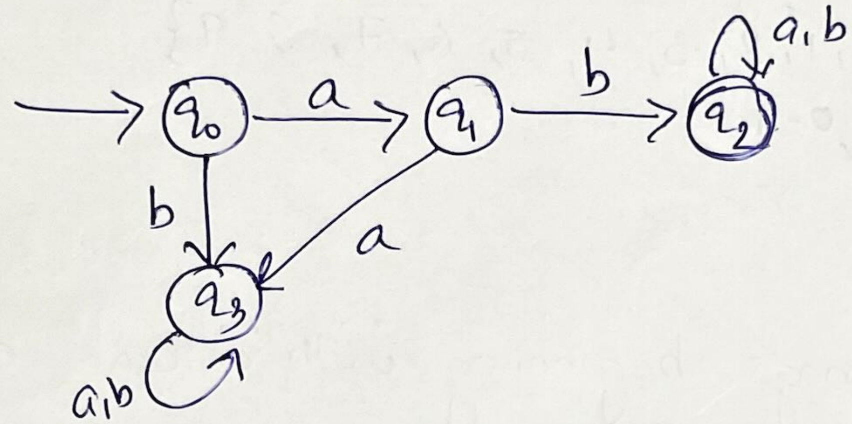
aba, aaba,
ababa

d) Set of strings which contains the pattern abac.



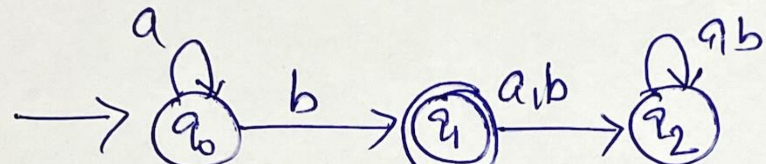
e> Set of all strings starting with the pre
'ab'. $\Sigma = \{a, b\}$

Sol^n:-



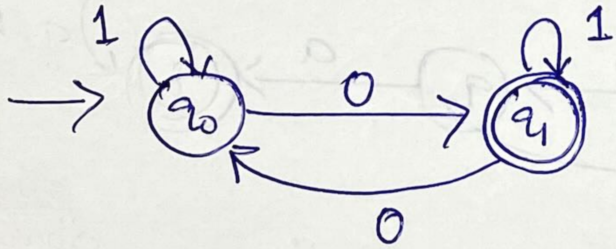
f> Set of strings for the language $L = \{a^n b : n \geq 0\}$

Sol^{n}:- $L = \{b, ab, aab, aaab, \ldots\}$



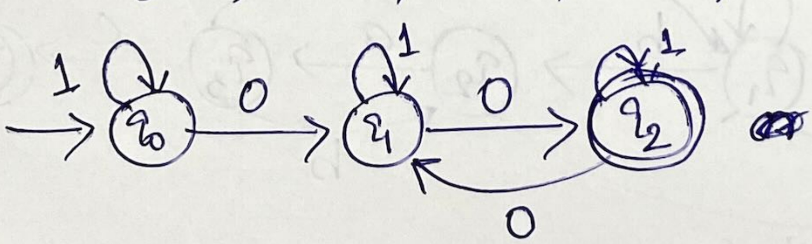
g> Set of strings having odd no. of 0's over $\Sigma = \{0, 1\}$

Sol^n:- $L = \{0, 01, 10, 0100, 1000, 0010 \ldots\}$



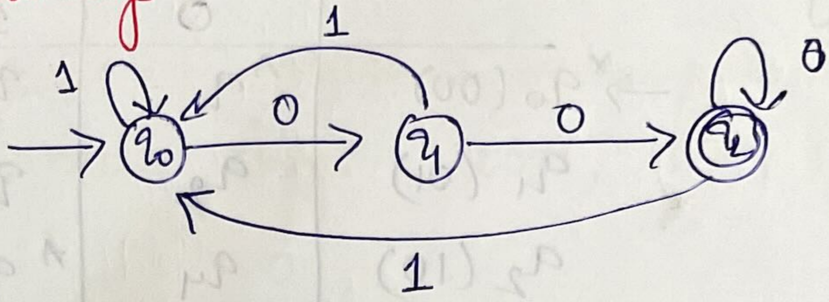
h> Set of strings having even no. of 0's over $\Sigma = \{0, 1\}$

Sol^n:- $L = \{00, 001, 100, 010, 1100, 0101 \ldots\}$



Q: Design DFA to accept the string that always ends with 00.

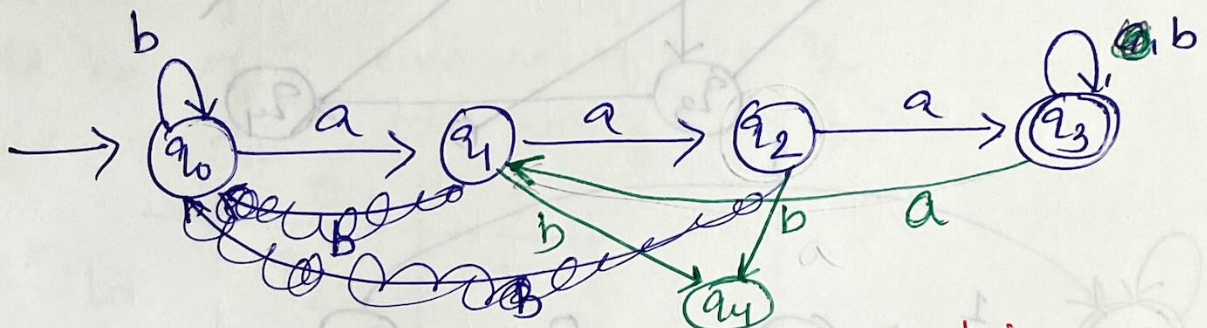
Sol^n:-



Q:- Design DFA to accept the language L, where L = { strings in which 'a' always appears in tripplets } over the set $\Sigma = \{a, b\}$

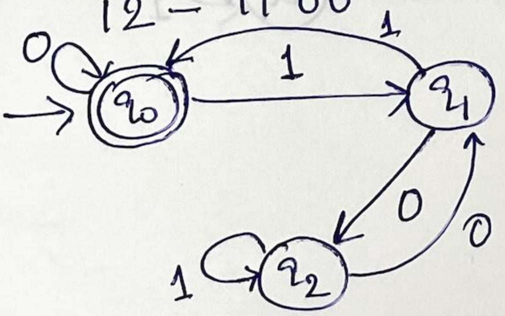
Sol^n: possible strings:-

L = { aaa, baaa, aaab, aaaaaa, aaabaaa, baaaaaa, aaaaaab, . . . }



Q: Design DFA which accepts binary numbers divisible by 3.

Sol^n:- 3 - 101
6 - 110
9 - 1001
12 - 1100



Possible remainders = 0, 1, 2

	0	1
→ q_0 (00)	* q_0	q_1
q_1(01)	q_2	* q_0
q_2(10)	q_1	q_2

22 - 10110
23 - 10111
24 - 11000

Q: Design DFA which accepts binary divisible by 5.

Soln: 5 101

10 1010

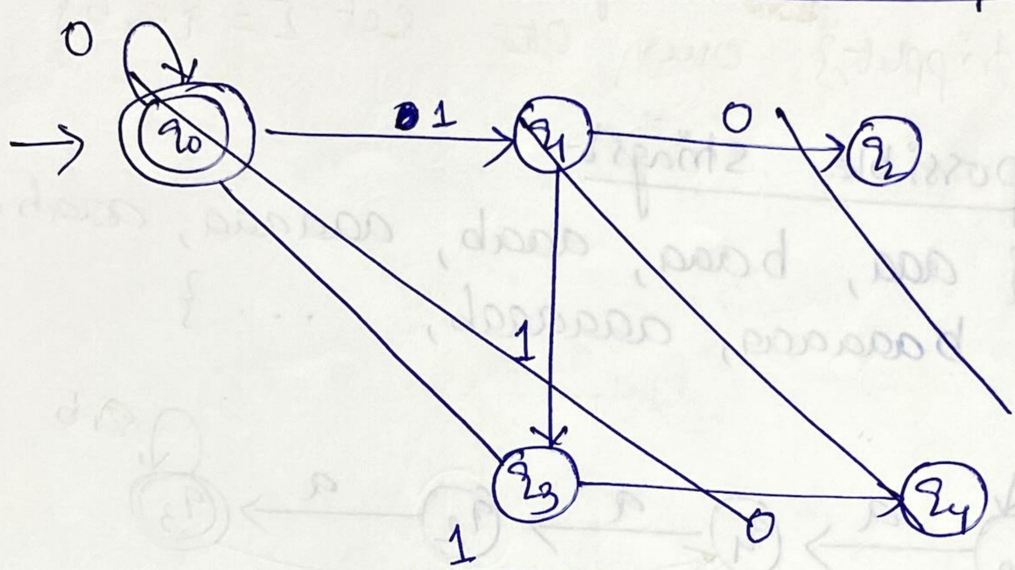
15 1111

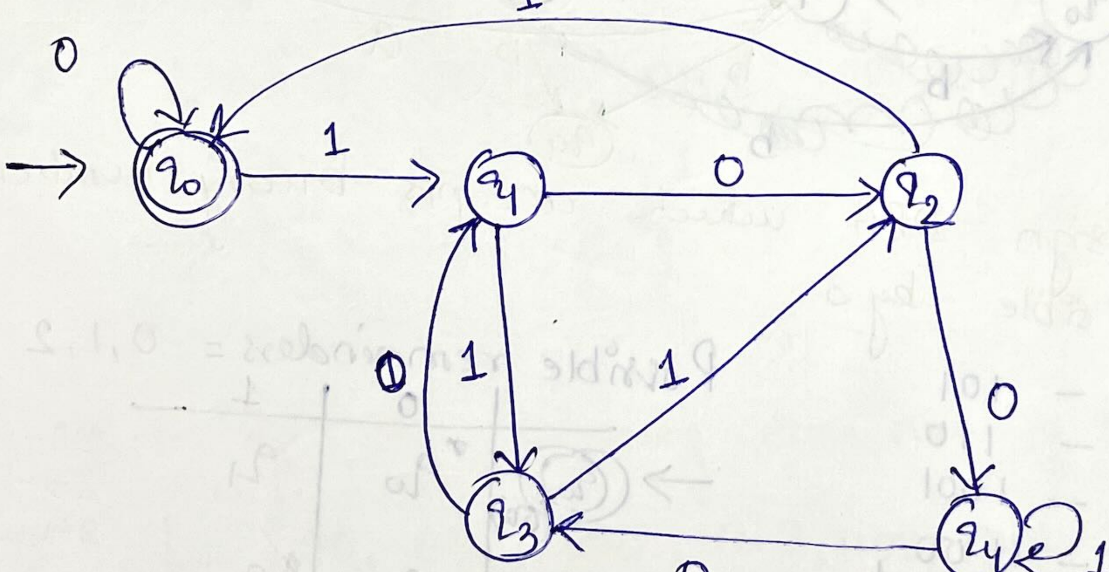
20 10100

25 11001

30 11110

	0	1
$\rightarrow {}^{*}q_0$ (00)	${}^{A}q_0$	q_1
q_1 (01)	q_2	q_3
q_2 (10)	q_4	${}^{A}q_0$
q_3 (11)	q_1	q_2
q_4 (100)	q_3	q_4

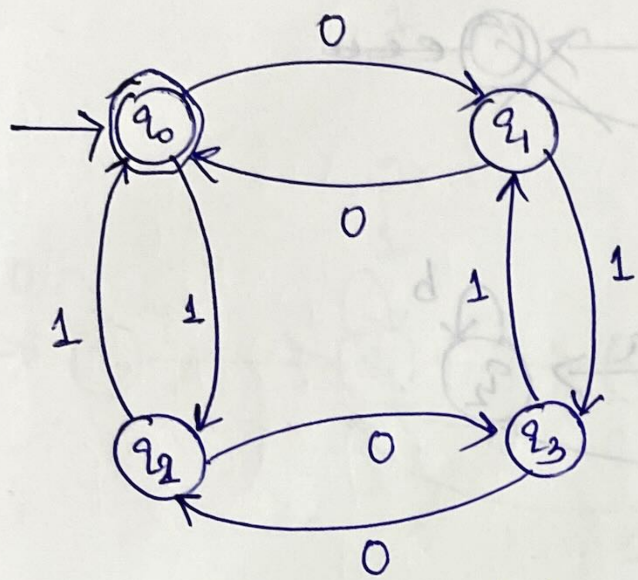




Q: Design DFA which accepts even no of 0's and even no. of 1's.

~~0 1 2 1 4 2 3 7 2 1 3 4 7 5 3 2 7~~

Solⁿ: Possible string:- 0011, 1100, 1010, 0100 ....

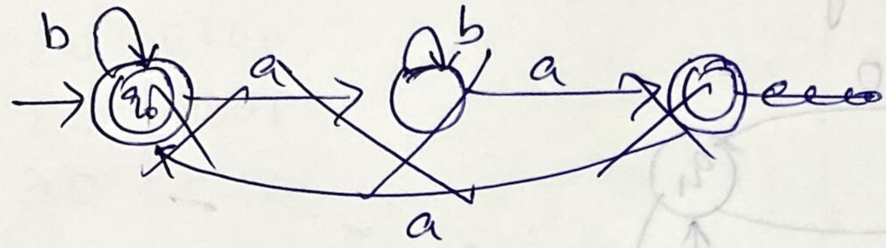


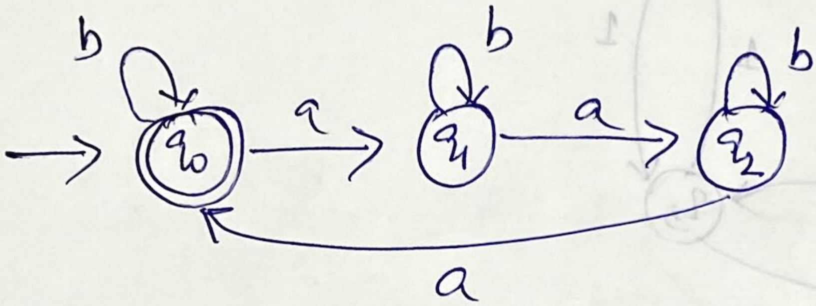
even no. of 0 even no. of 1 – q_0
even no. of 0 odd no. of 1 – q_1
odd no. of 0 even no. of 1 – q_2
odd no. of 0 odd no. of 1 – q_3

	0	1
→* q_0	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

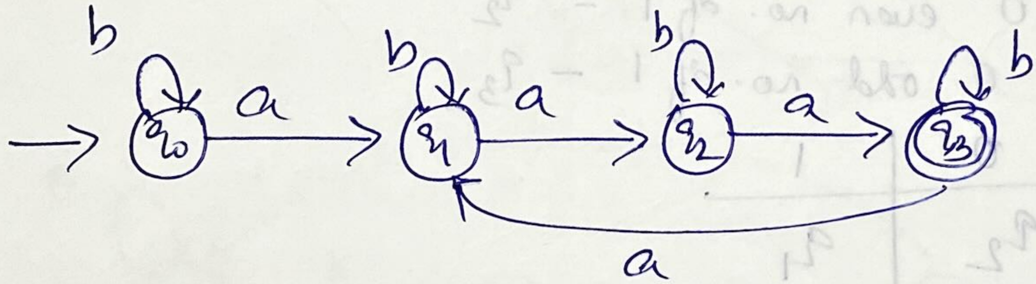
Q: Design DFA to accept strings in L such that the total no. of a's in them are divisible by 3.

Solⁿ: eg: aaa, baaa, aaab, ...



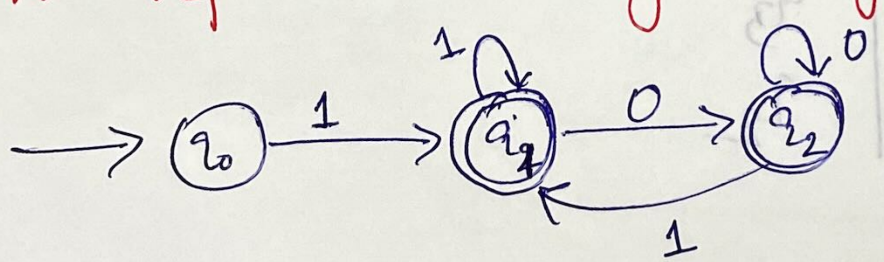


Or

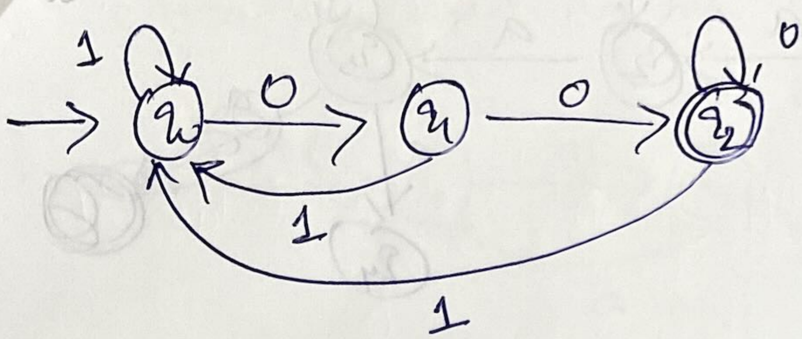


Q: Design a DFA to accept odd and even no. represented using binary notation.

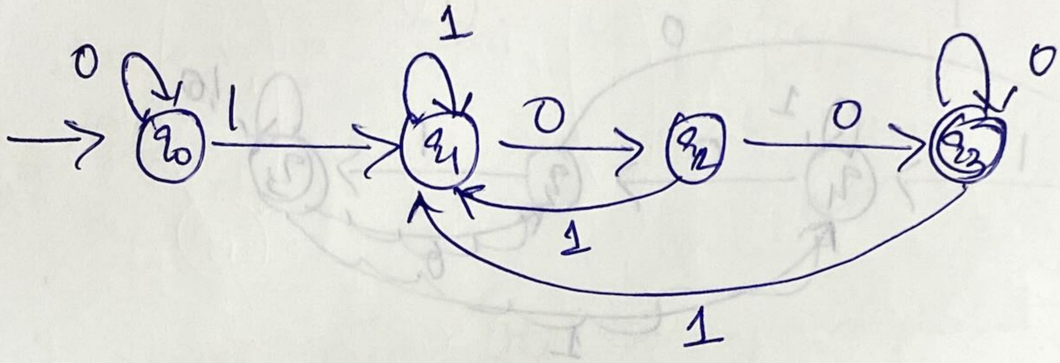
Solⁿ:



Q: Design a DFA for divisibility by 4.

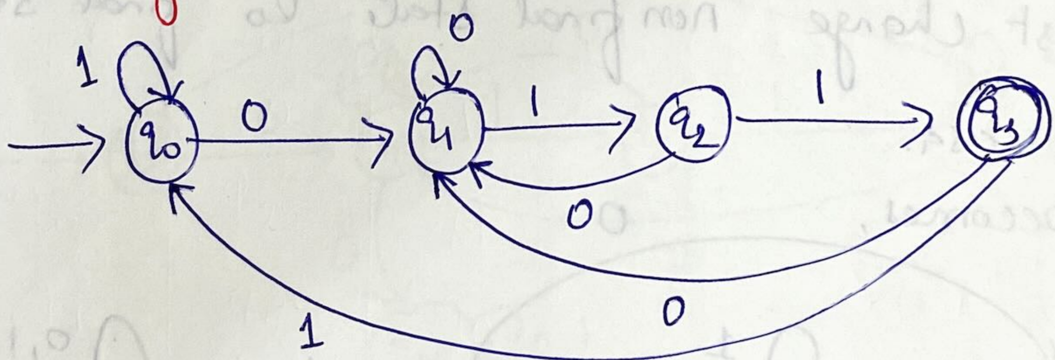


Or



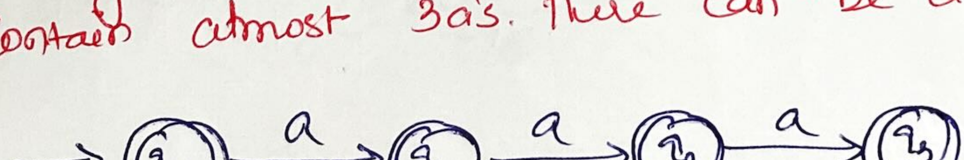
Q: Design a DFA to accept the string of 0's and 1's ending with 011.

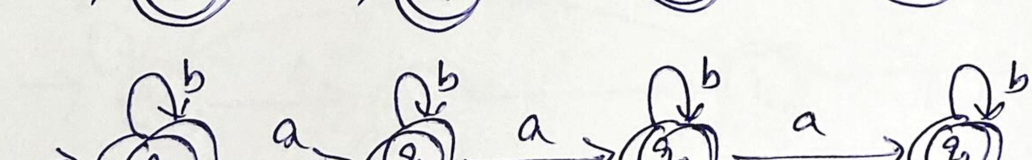
Sol^n:

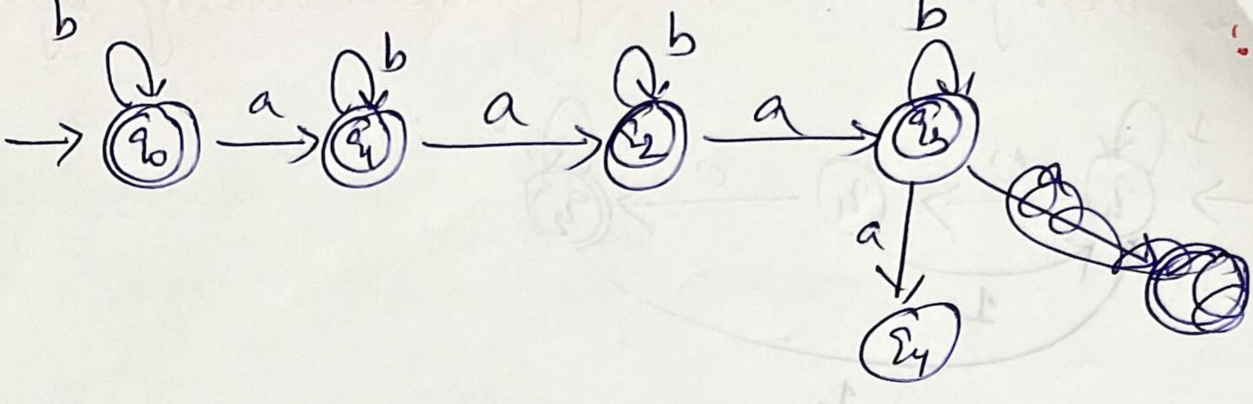


Q: Design a DFA with i/p symbols a and b that contain atmost 3 a's. There can be any no. of b's.

Sol^n:

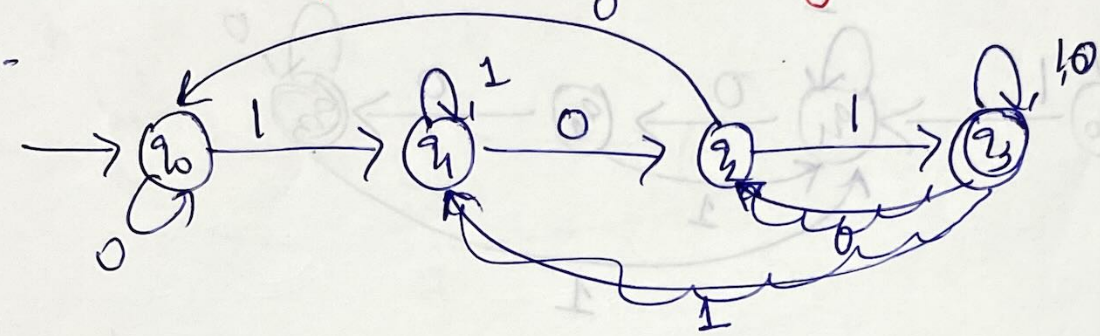






Q:- Construct a DFA that accepts all the strings on $\{0,1\}$ except those containing the substring 101.

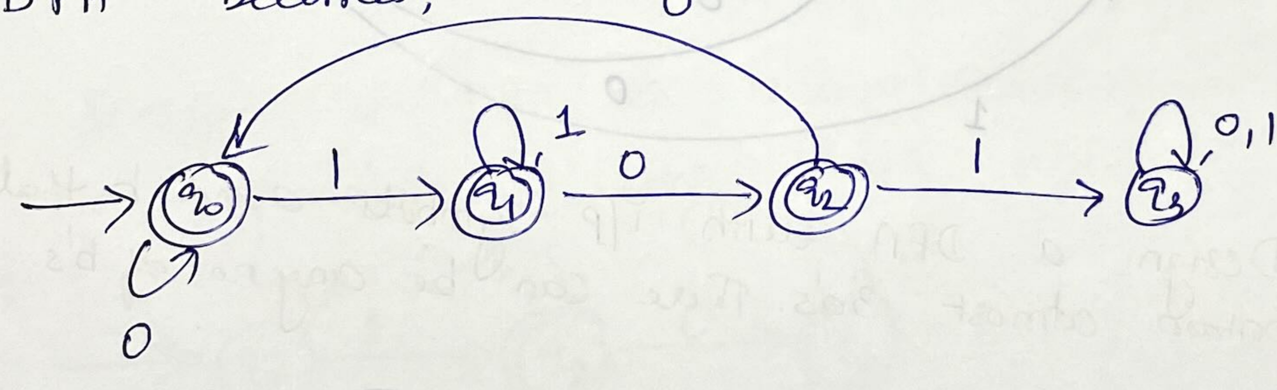
Sol^n:-



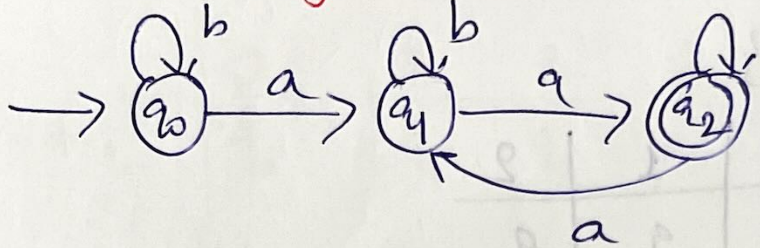
DFA accepting 101 as substring

Now, just change non final state to final state and vice versa.

DFA becomes,



Q:- DFA to accept strings over $\Sigma(a\ b)$ containing even no. of a's.



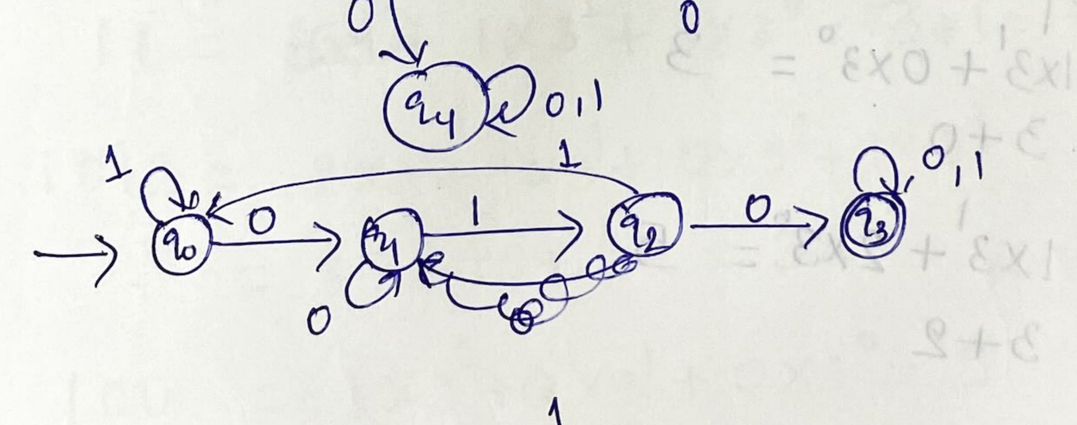
abaaab

Q: Design a DFA to accept

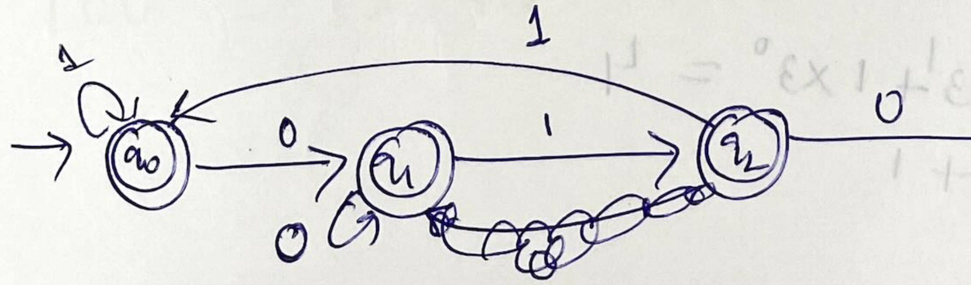
i) Binary strings in which every 0 is followed by 11.

ii) Strings over the binary alphabet $\Sigma = \{0,1\}$ that do not contain the substring 010.

Solⁿ: i)



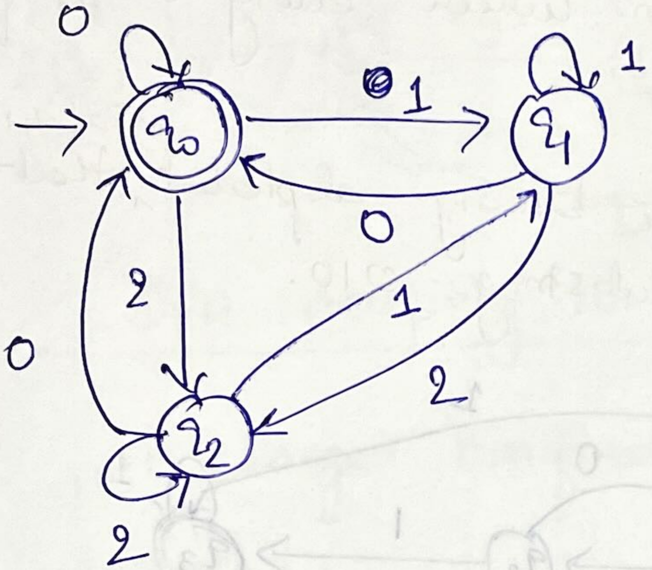
ii)



Q: Design a DFA for ternary no. divi. by 3.

Soln: $\Sigma = \{0, 1, 2\}$

	0	1	2
$\rightarrow * q_0$	q_0	q_1	q_2
q_1	q_0	q_1	q_2
q_2	q_0	q_1	q_2



	101	2
3	33	0
3	11	2
3	3	0
3	1	

$(10202)_3$

$10 = 0$

$01 = 0\times3^1 + 1\times3^0 = 1$

$1\times3^4 + 0\times3^3 + 2\times3^2 + 0\times3^1 + 2\times3^0$

$81 + 0 + 18 + 0 + 2 = 101$

$00 = 0$

1

2

Ternary no. Decimal

$10 = 1\times3^1 + 0\times3^0 = 3$

$= 3+0$

$12 = 1\times3^1 + 2\times3^0 = 5$

$= 3+2$

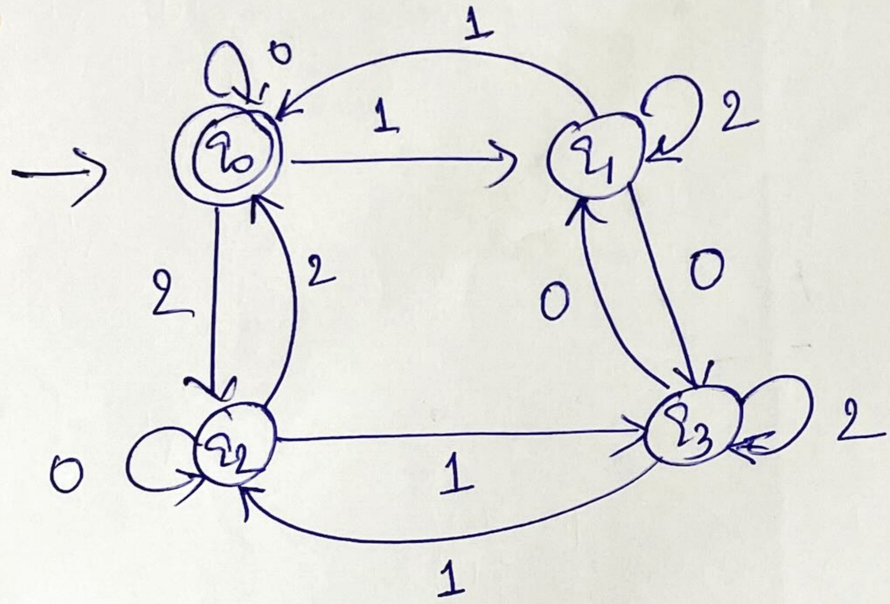
$11 = 1\times3^1 + 1\times3^0 = 4$

$= 3+1$

DFA for divisibility by 4 for ternary no.

$\Sigma = \{0, 1, 2\}$

	0	1	2
$\rightarrow * q_0$	$*q_0$	q_1	q_2
q_1	q_3	$*q_0$	q_1
q_2	q_2	q_3	$*q_0$
q_3	q_1	q_2	q_3



Ternary — Decimal

$11 = 1 \times 3^1 + 1 \times 3^0 = 3 + 1 = 4$

$1012 = 2 \times 3^0 + 1 \times 3^1 + 0 \times 3^2 + 1 \times 3^3$

$= 2 + 3 + 0 + 27 = 32$

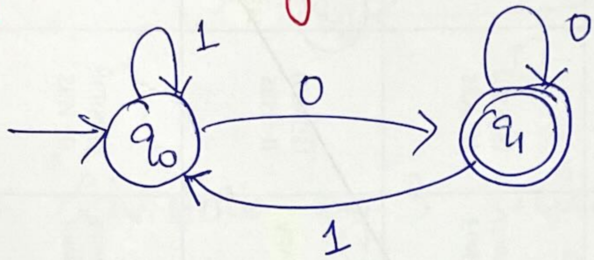
$100 = 1 \times 3^2 + 0 \times 3^1 + 0 \times 3^0 = 9$

Q:- Design a FA which accepts the only input 101 over the input set $\Sigma = \{0,1\}$.

Soln:

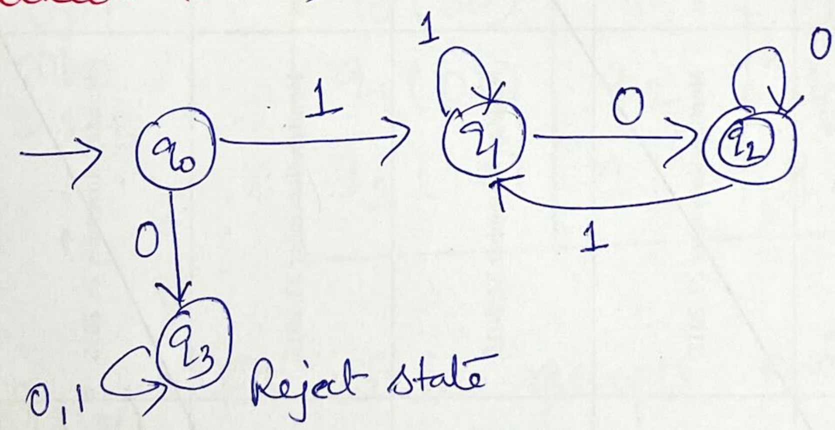
Q: Design a FA which checks whether the given binary no. is even.

Soln:



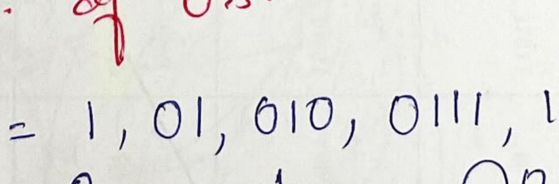
Q: Design FA which accepts only those strings which starts with 1 and ends with 0.

Soln:

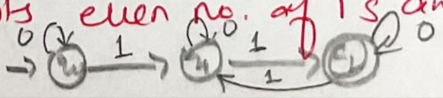


Q:- Design FA which accepts odd no. of 1's and any no. of 0's.

Soln: Strings = 1, 01, 010, 0111, 1011, 10,... 1101110

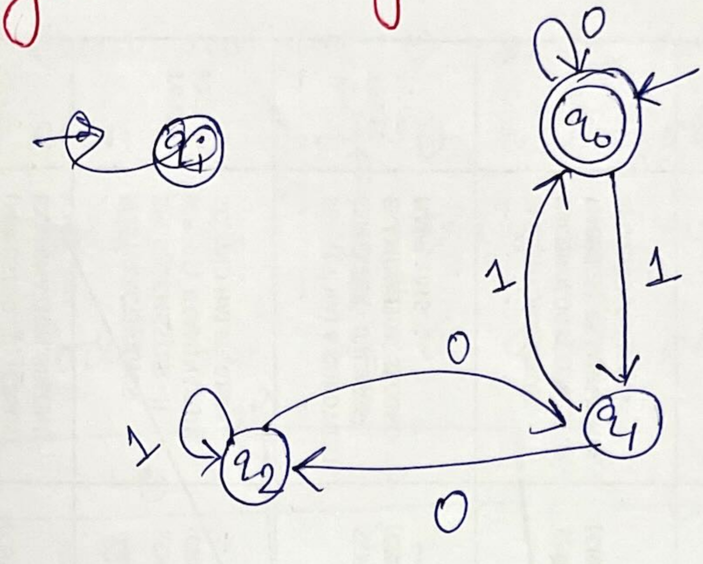


Q: Design FA which accepts even no. of 1's and any string of 0's.



Q: Design FA which checks whether a given binary no. is divisible by three.

Sol^n:



18 - 10010
21 - 10101
22 - 10110
23 - 10111

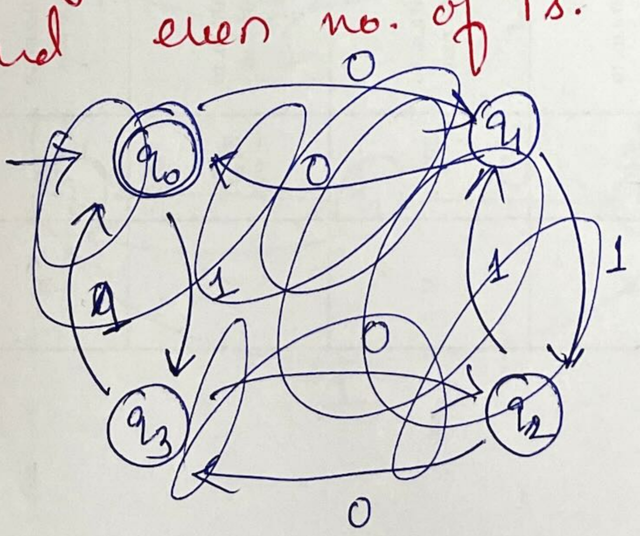
No. divisible by 3 will have three remainders 0, 1, 2. So, we will have three states q_0, q_1 & q_2 wrt remainders.

000
001
010
011

	0	1
*q_0 (00)	*q_0	q_1
q_1 (01)	q_2	*q_0
q_2 (10)	q_1	q_2

Q: Design FA which accepts even no. of 0's and even no. of 1's.

Sol^n:-



This FA will have four different possible combinations :-

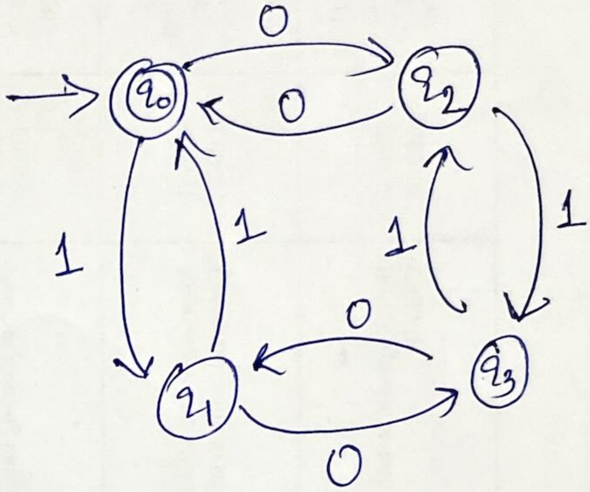
i) even no. of 0 and even no. of 1. – q_0

ii) even no. of 0 and odd no. of 1 – q_1

iii) odd no. of 0 and even no. of 1 – q_2

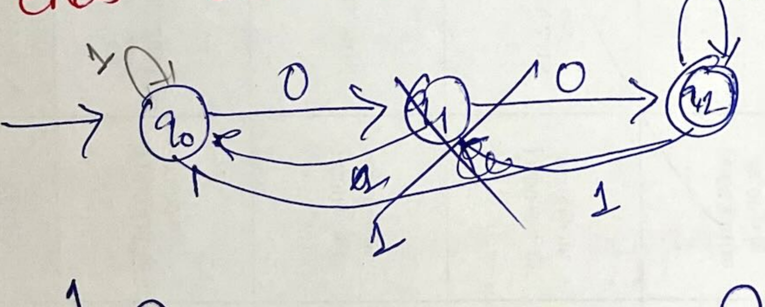
iv) odd no. of 0 and odd no. of 1 – q_3

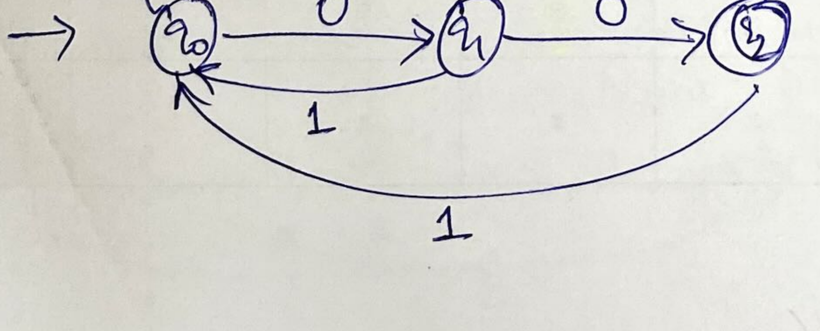
States	I/P 0	1
$\rightarrow {}^*q_0$	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2



Transition dig.

Q: Design FA to accept the string that always ends with 00.



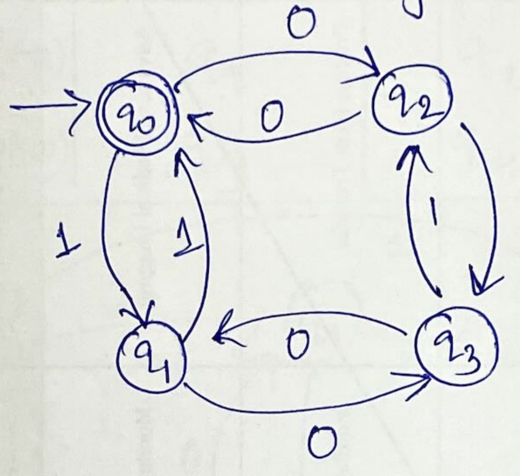


Q: Consider the finite automaton transition table shown below with $F = \{q_0\}$.

States	Inputs 0	1
$\rightarrow * q_0$	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

Find the language accepted by the finite automata.

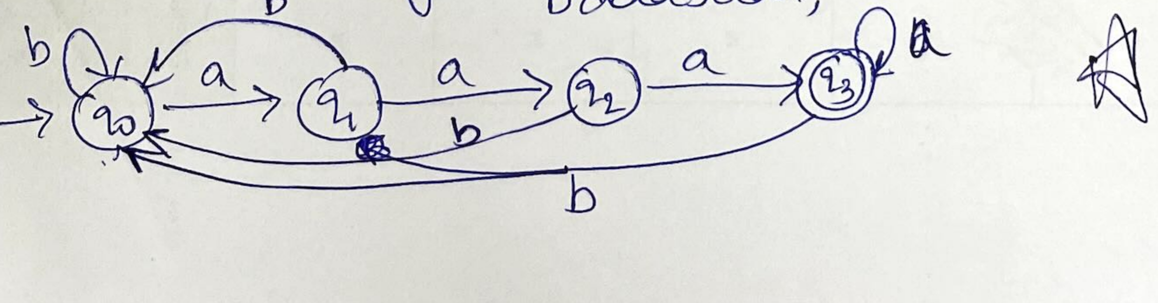
Solⁿ: The transition dig.:-



It shows that the string containing even no. of 1's and 0's are accepted.

Q: Design FA to accept L, where L = {strings in which a ~~language~~ always appears in tripplets} over the set $\Sigma = \{a, b\}$.

Solⁿ: Possible string values - aaa, baaa, aaab, baaab, baaabaaa, baaaaaa....



Q:- Design a DFA which checks whether the given binary no. is divisible by 5.

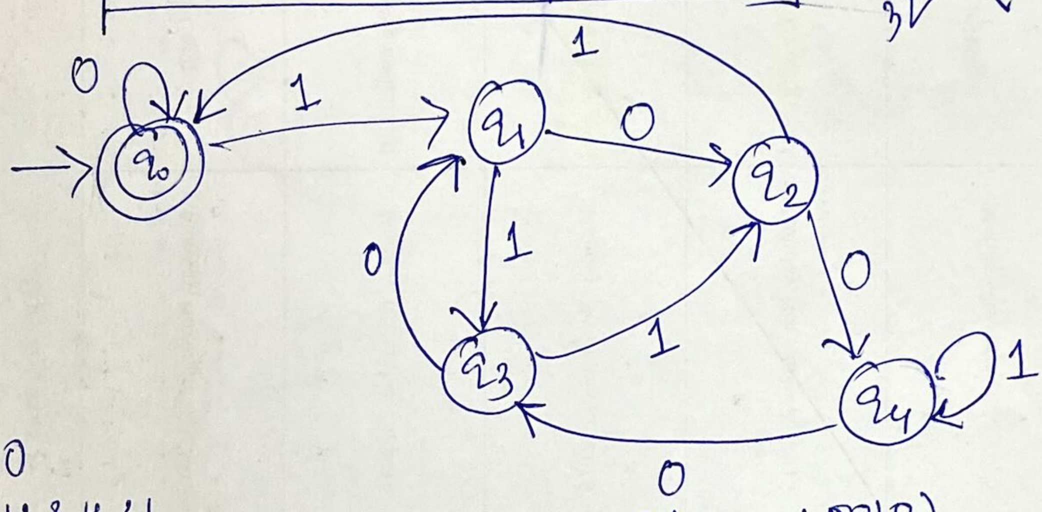
Sol^n:- The no. divisible by 5 will have 5 remainder

0, 1, 2, 3, 4.

$q_0 - 0$, $q_1 - 1$, $q_2 - 2$ $q_3 - 3$ $q_4 - 4$

	0	1
$\rightarrow (q_0)$ (00)	q_0	q_1
q_1(01)	q_2	q_3
q_2(10)	q_4	q_0
q_3(11)	q_1	q_2
q_4(100)	q_3	q_4

32 16 8 4 2 1
1 1 1 1 0



30

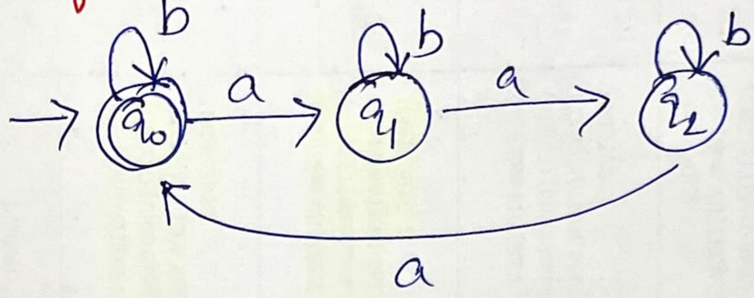
16 1 2 8 4 2 1

1 1 0 0 1 0

$\delta(q_0, 110010) = \delta(q_1, 10010)$
$= \delta(q_3, 0010)$
$= \delta(q_1, 010)$
$= \delta(q_2, 10)$
$= \delta(q_0, 0)$
$= {}^*q_0$

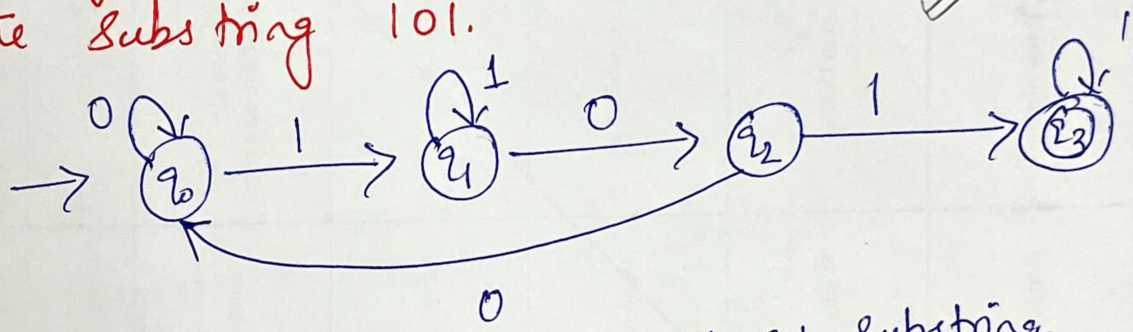
Q: Design FA to accept L where all the strings in L are such that the total no. of a's in them are divisible by 3.

Sol^n:-



Q:- Construct a DFA that accepts all the strings on $\{0, 1\}$ except those containing the substring 101.

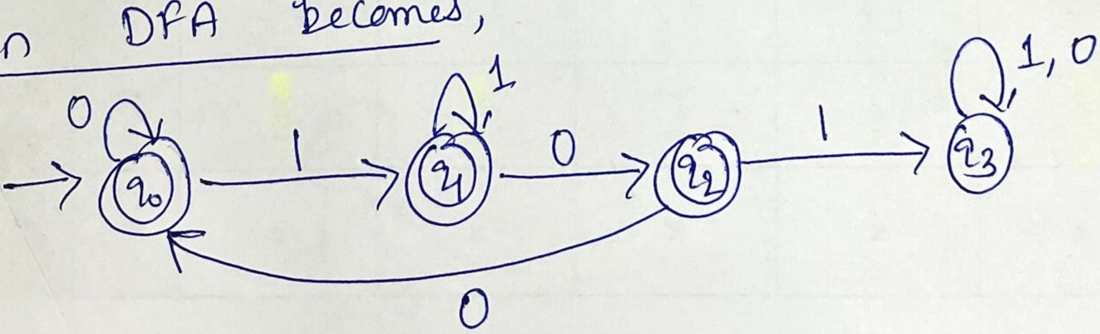
Sol^n:



<u>DFA accepting 101 as substring</u>

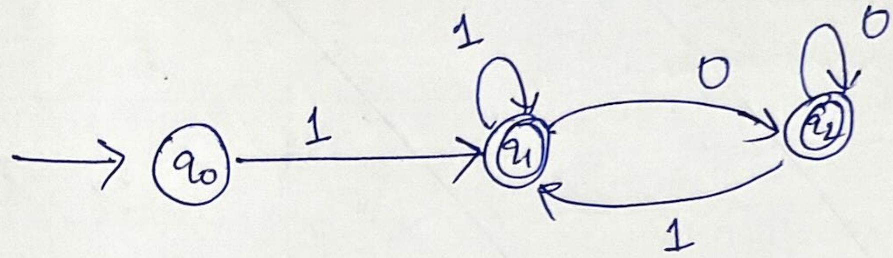
Now, just change non final states to final state and make final state as non-final state.

<u>Then DFA becomes,</u>



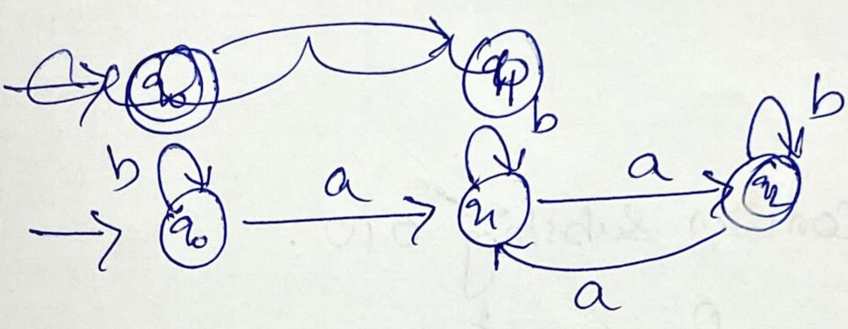
Q:- Design DFA to accept odd and even no. represented using binary notation.

Sol^n:-



Q:- Design DFA to accept strings over the alphabet $\Sigma = \{a, b\}$ containing even no. of a's. (May-14)

Sol^n:-



Q:- Design a DFA to accept

i) Binary strings in which every 0 is followed by 11.

ii) Strings over the binary alphabet that do not contain the substring 010.

Sol^n:- (i)