Context Free language & Context Free Grammer.

writing Grammer for a language.

1) Language $L = \{\epsilon, a, aa, \ldots\}$

$S \to aS$
$S \to \epsilon$

Generation of string:-

$S \to aS$
$\to aaS$
$\to aaaS$
$\to aaa$

L এর শুরুতে ϵ থাকলে $S \to \epsilon$ দিতে হবে।

2) Language $L = \{a, aa, aaa, \ldots\}$

$S \to aS$
$S \to a$

Generation of string:-

$S \to aS$
$\to aaS$
$\to aaaS$
$\to aaaa$

L এর শুরুতে ϵ না থাকলে $S \to a$ শুরুতে a আছে।

3) Language $L = \{b, ab, aab, aaab, \ldots\}$

$S \to aS$
$S \to b$

Generation of string

$S \to aS$
$\to aaS$
$\to aaaS$
$\to aaab$

(4) Language $L = \{ w \in \{a, b\}^* \}$

$L = \{ \epsilon, a, b, aa, ab, ba, bb, \ldots \}$

$S \rightarrow aS$, $S \rightarrow bS$

$S \rightarrow \epsilon$

Generation of string

$S \rightarrow aS$

$\rightarrow abS$

$\rightarrow abbS$

$\rightarrow \cancel{abbaS}$

$\rightarrow abbabS$

$\rightarrow abbab$

5) Language $L = \{ a^n b^n \mid n \geq 0 \}$

$L = \{ \epsilon, ab, aabb, aaabbb, \ldots a^n b^n \}$

$S \rightarrow \epsilon$

$S \rightarrow aSb$

Generation of string

$S \rightarrow aSb$

$\rightarrow aaSbb$

$\rightarrow aaaSbbb$

$\rightarrow aaabbb$

6) Language $L = \{ a^n b^n \mid n \geq 1 \}$ ϵ হবে না
একবার a & একবার b হবে

$L = \{ ab, aabb, aaabbb, \ldots, a^n b^n \}$

$\Rightarrow S \rightarrow aSb$
$S \rightarrow ab$

Generation of String

$S \rightarrow aSb$
$\rightarrow aaSbb$
$\rightarrow aaaSbbb$
$\rightarrow aaaabbbb$

Conversion of Context Free Grammar to Context Free language

What is the language generated by the following CFG?

$S \to aS \mid Sb \mid \epsilon$

$S \to ~~b~~Sb$

$S \to \epsilon$

$S = \{ \epsilon, a, aa, ab, bb, aaabb ... \}$

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

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

What is the language generated by following CFG?

$S \to aSa$
$S \to bSb$
$S \to a$
$S \to b$

which is the set of ~~all~~ length pallindrome
$L = \{ w \{a, b\} \mid w$ is an odd lengths pallindromes$\}$

1st : $S \to a$
$S \to b$

2nd : $S \to aSa \Rightarrow aaa$

3rd : $S \to bSb \Rightarrow bbb$

4th : $S \to aSa \Rightarrow aba$ rev: aba

5th : $S \to bSb \Rightarrow bab$

~~6th : $S \to$~~ reverse: bab

6th : $S \to aSa$ (aaa)
$\to abSba$ (bbb)
$\to abbba$
reverse abbba

7th $S \to aSa \Rightarrow abSba$
$= ababa$
reverse ababa

(03) Consider the grammer : (Gate-CS-2009)

$S \rightarrow aSb$ $\Rightarrow aba$ $\Rightarrow aaSb$ aaa,

$S \rightarrow bSb$ $\Rightarrow bab$ bbb

$S \rightarrow a$ ababa- - -

$S \rightarrow b$

all are odd length pallindromes

a) All pallindromes

~~b) All~~ odd length pallindromes

c) strings that begin and end with the same symbol.

d) All even length pallindromes.

(04) consider the following context free Grammer:

$G_1 : S \rightarrow aS \mid B, \ B \rightarrow b \mid bB$

$G_2 : S \rightarrow aA \mid bB, \ A \rightarrow aA \mid B \mid \epsilon, \ B \rightarrow bB \mid \epsilon$

which of the following pairs of languages is generated by G_1 and G_2 respectively?

a) $\{a^m b^n \mid m \geq 0 \text{ or } n > 0\}$ and $\{a^m b^n \mid m > 0 \text{ and } n > 0\}$

b) $\{a^m b^n \mid m > 0 \text{ and } n > 0\}$ and $\{a^m b^n \mid m > 0 \text{ or } n \geq 0\}$

c) $\{a^m b^n \mid m \geq 0 \text{ or } n > 0\}$ and $\{a^m b^n \mid m > 0 \text{ and } n > 0\}$

D) $\{a^m b^n \mid m \geq 0 \text{ and } n > 0\}$ and $\{a^m b^n \mid m > 0 \text{ or } n > 0\}$

Using $S \Rightarrow B \Rightarrow b$, b can be generated.
Using $S \Rightarrow B \Rightarrow bB$, bb can be generated
Using $S \Rightarrow aS \Rightarrow aB \Rightarrow ab$ can be generated
Using $S \Rightarrow aS \Rightarrow aB \Rightarrow abB \Rightarrow abb$ can be generated
Using $S \Rightarrow aS$

so, a is = 0 number or a is more than 0 | b is ~~not allow~~ zero, b is greater than 0

$\therefore L(G1) = \{a^m b^n \mid m \geq 0 \text{ and } n > 0\}$

Consider the Grammer G2:

using $S \Rightarrow aA \Rightarrow a$, a can
$S \Rightarrow bB \Rightarrow b$, b can
$S \Rightarrow aA \Rightarrow aa$ can
$S \Rightarrow bB \Rightarrow bb$ "
$S \Rightarrow aA \Rightarrow aB \Rightarrow abB \Rightarrow abb$ can

a is greater than 0
b is greater than 0

$\therefore L(G2) = \{a^m b^n \mid m \geq 0 \text{ and } n \geq 0\}$

(05) $L(G) = \{a^n b^n, n >= 0\}$

$L = \{\epsilon, abb, aabb, aaabbb, \dots\}$

$S \rightarrow aSb \mid \epsilon$

Key points :-

For a given language L(G) there can be more than one grammer which can produce L(G)

The grammer G corresponds to Language L(G) must generate all possible strings of L(G)

The grammer G corresponding to language L(G) must not generate any string which is not part of L(G)

(A) $S \rightarrow AC \mid CB$

$C \rightarrow aCb \mid a \mid b$

$A \rightarrow aA \mid \epsilon$

$B \rightarrow Bb \mid \epsilon$

(B) $S \rightarrow aS \mid Sb \mid a \mid b$

(C) $S \rightarrow AC \mid CB$

$C \rightarrow aCb \mid \epsilon$

$A \rightarrow aA \mid \epsilon$

$B \rightarrow Bb \mid \epsilon$

(D) $S \rightarrow AC \mid CB$

$C \rightarrow aCb \mid \epsilon$

$A \rightarrow aA \mid a$

$B \rightarrow Bb \mid b$.

(A) $S \to AC \to aAC \to aC \to aCb \to \underline{ab}$

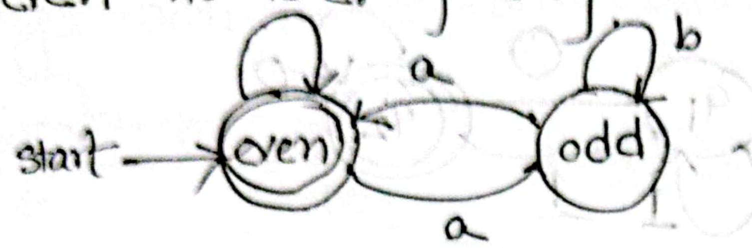
$a = 1,\ b = 1,$ but $i \neq j$

(B) $S \to$ ~~aa~~ $aS \to ab,\quad a = 1, b = 1,\ i \neq j$ ✗

(C) $S \to AC \to C \to aCb = ab$

(D) ✓

Design a DFA that accepts $L = \{s \in \{a,b\}^* : s$ contains an even number of a's$\}$



what is finit Automaton ? Types of finite Automaton.

Finite automa has a set of states and control moves from one state to other by external inputs.

Deterministic Finite automata: can not be in more than one state in a particular time.

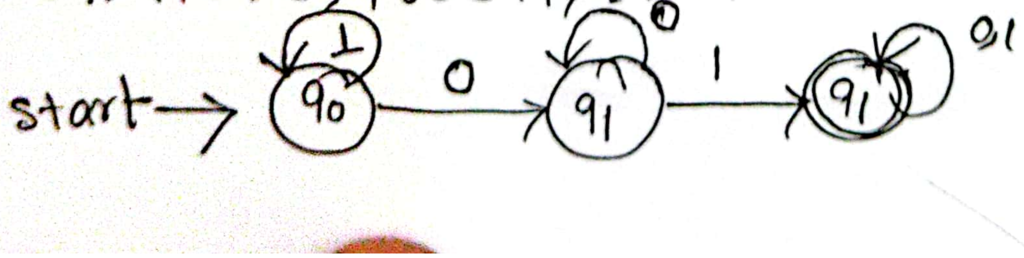
Non Deterministic Finite automata: can be in more than one state in a particular time

$$DFA = (Q, S, d, q_0, F)$$

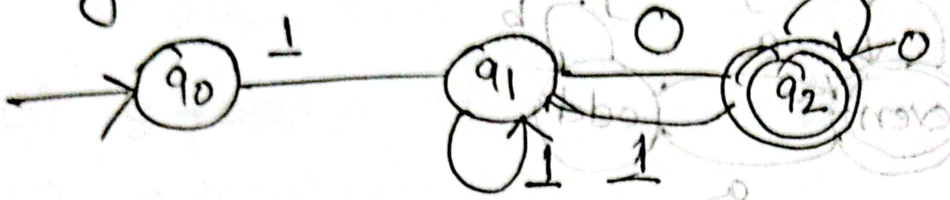
Design a DFA to accept the language.

$L = \{x01y \mid x$ and y are any strings of 0's and 1's$\}$

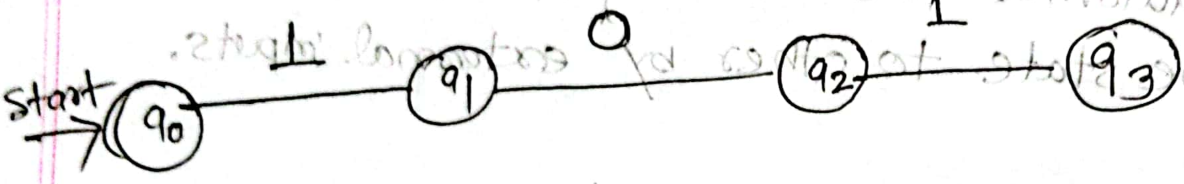
01, 11010, 100011, ...



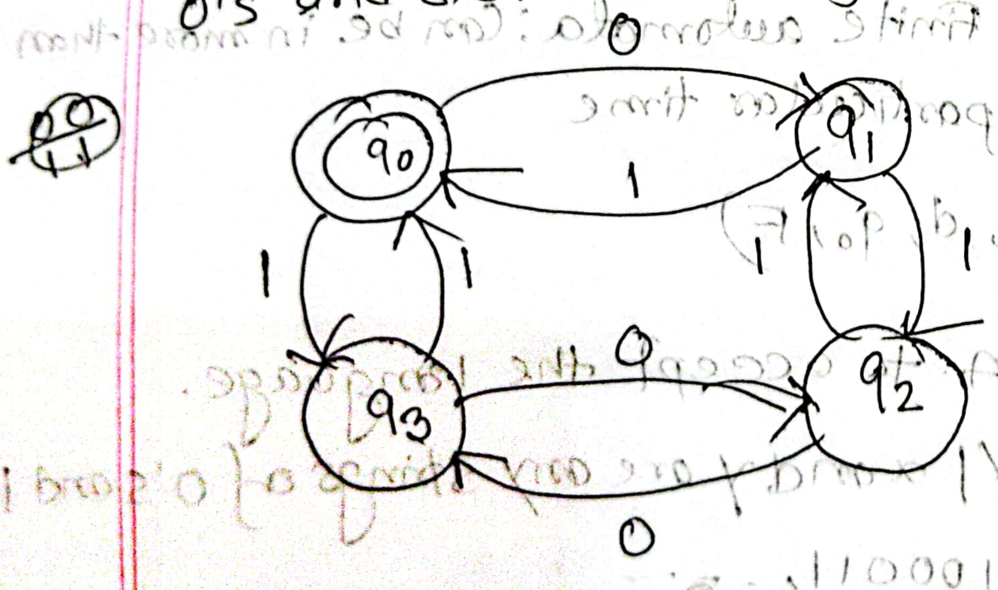
01. Design a DFA with $\Sigma = \{0,1\}$ accepts those strings which starts with 1 and ends with 0



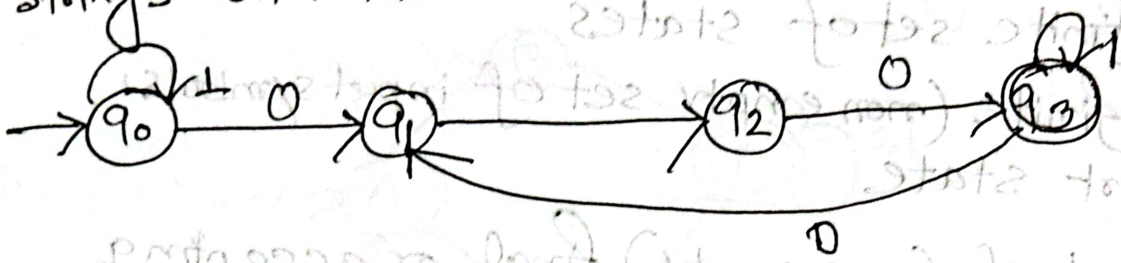
02: Design a FA $\Sigma = \{0,1\}$ accept only input 101



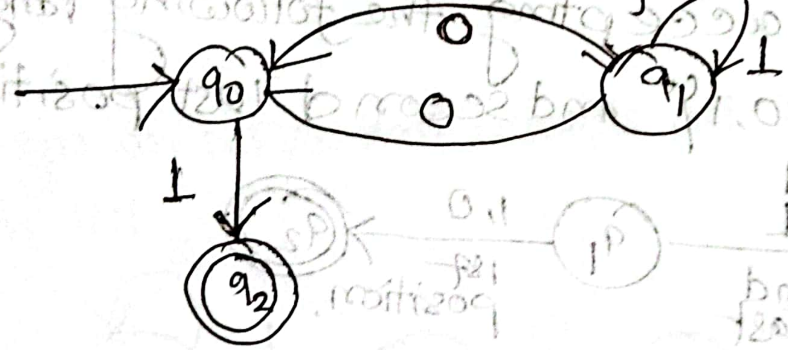
03 Design a DFA with $\Sigma = \{0,1\}$ accepts even number of 0's and even number of 1's



04. Design a FA with $\Sigma = \{0,1\}$ accepts the set of all strings with three consecutive 0's.



05. Design a FA with $\Sigma = \{0,1\}$ accepts the strings with an even number of 0's followed by single 1



Non deterministic Finite automata:

An NFA A is a 5-tuple $A = (Q, S, d, q_0, F)$ where

Q = a finite set of states

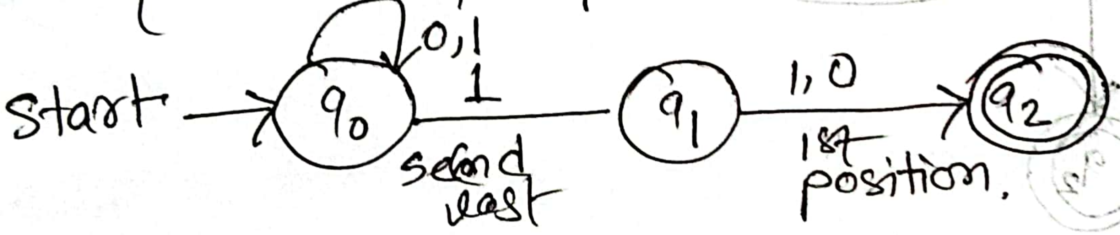
S = a finite (non empty set of input symbols.

q_0 = start state

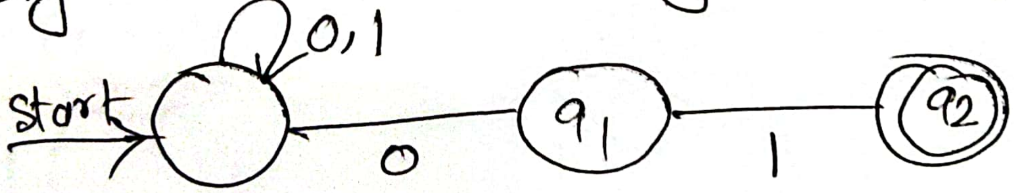
F = a set of (non empty) final or accepting states.

Design an NFA accepting the following language

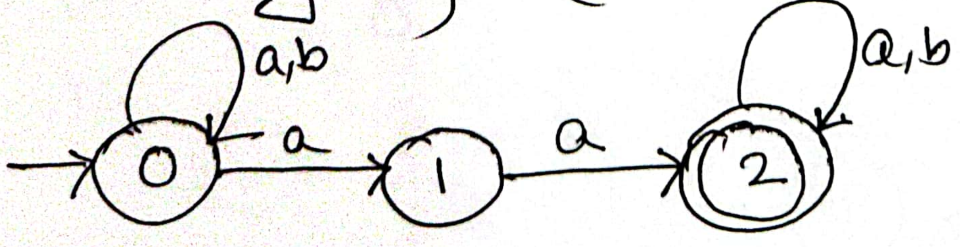
$L = \{ w \mid w \in \{0,1\}^*$ and second last position is



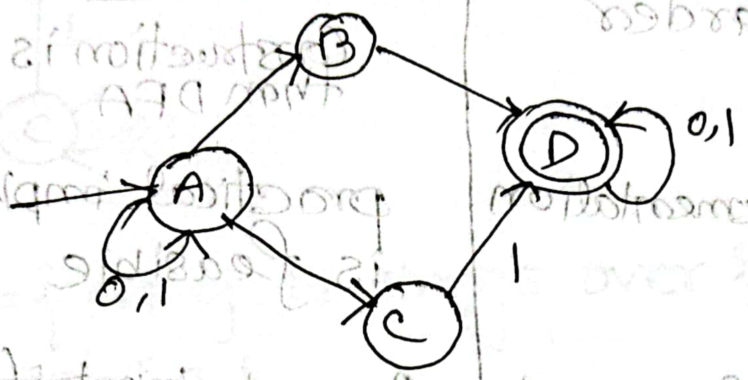
Design an NFA accepting all strings that end in 0



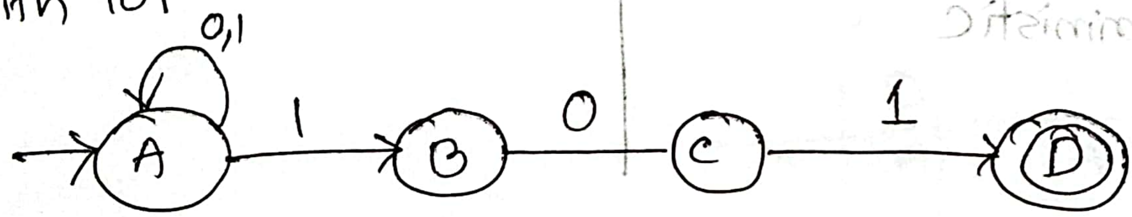
Design an NFA $L = \{ w : \in \{a,b\}$: w contains the substring aa $\}$ $(a+b)^* aa (a+b)$



Design an NFA that accepts any binary string that contains 00 or 11 as a substring.



Design an NFA that accepts all binary strings that end with 101



Difference between NFA to DFA?

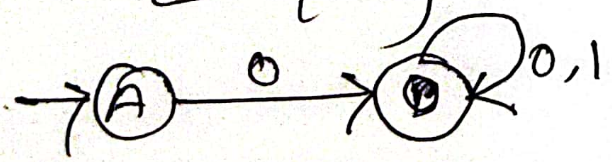
DFA	NFA
Sometimes harder construct	Construction is easier than DFA
practical implementation is Feasible	practical implementation is feasible
Accepts input if the last state is in F	Accepts input if one of the last states is in F.
All transitions are deterministic	All transition are non deterministic.

Conversion from NFA to DFA :-

L = {set of all strings over (0,1) that starts with 0}

DFA তে φ থাকতে পারবে না।

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



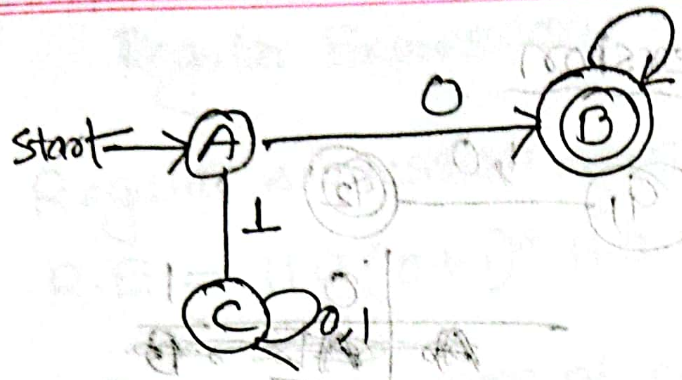
	0	1
A	B	φ
B	B	B

DFA:

	0	1
A	B	C
B	B	B
C	C	C

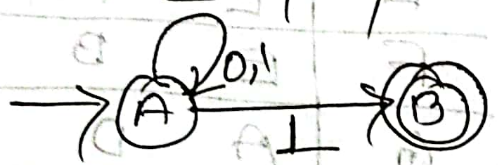
$\boxed{\text{C Dead state}}$

Subset construction method



#2. L = { set of all strings over {0,1} that ends with '1' }

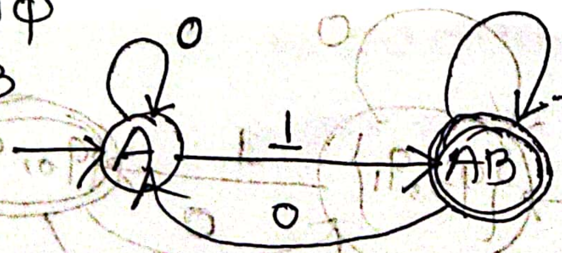
Σ = {0,1}



	0	1
A	A	{B, A}
B	Φ	Φ

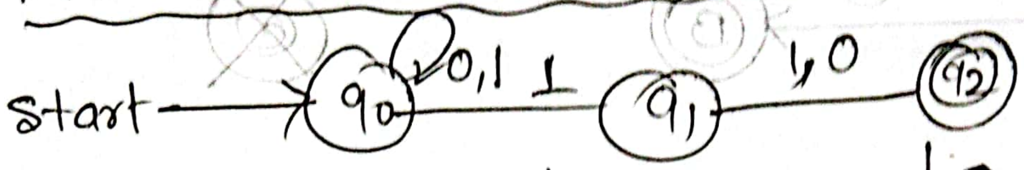
AB → 0
AUB
AUΦ
= A
ABU Φ
ABU Φ
= AB

	0	1
A	{A}	{AB}
AB	{A}	{AB}



Subset Construction method.

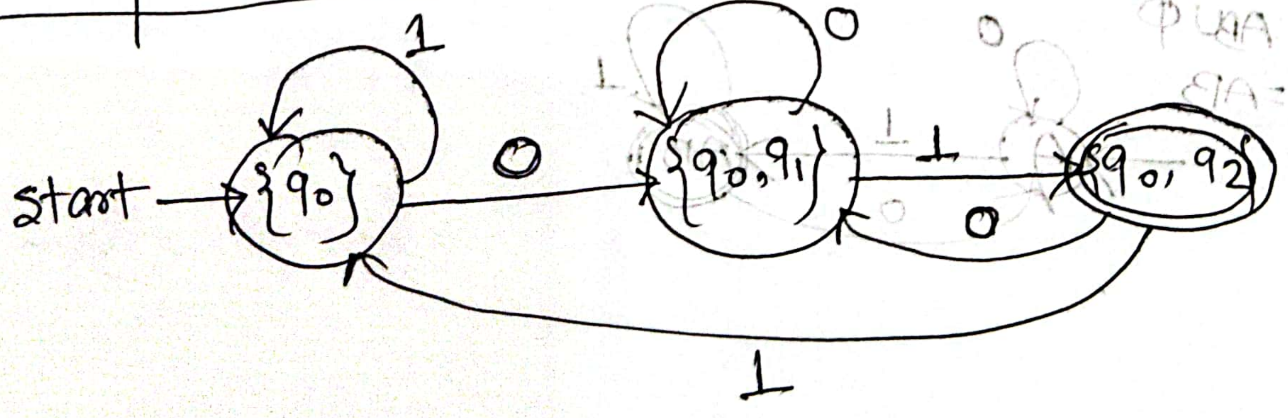
NFA to DFA conversion



		0	1
A	q_0	$q_0 q_1$	q_0
B	q_1	ϕ	q_2
C	q_2	ϕ	ϕ
D	$q_0 q_1$	$q_0 q_1$	$q_0 q_2$
E	$q_0 q_2$	$q_0 q_1$	q_0
F	$q_0 q_1 q_2$	$q_0 q_1$	$q_0 q_2$
	$q_0 q_1 q_2$	$q_0 q_1$	$q_0 q_2$

	0	1
~~A~~	~~A~~	~~B~~
B	~~EA~~	DB
C	A	D
D	A	A
E	E	F
F	E	B
G	A	D
H	E	F

	0	1
B	E	B
E	E	F
F	E	B



Regular Expression:

Regular expression of strings begin and end with 110

$R.E = 110(0+1)^* 110$

Regular expression of strings containing exactly three consecutive 1's.

$R.E = (0+1)^* 111 (0+1)^*$

$G: S \rightarrow 0S1 | A$
$A \rightarrow 0A | \epsilon$

A grammar for $L = \{0^m 1^n \mid m > n\}$

CFG?

$S = 0S1$
$= 00S11$
$= 000S111$
$= 0000A111$
$= 00000\epsilon 1111$
$= 000001111$