18001A0545

1) Derive string $W = abaabaa$ from production rules

$S \rightarrow a$ (1)

$S \rightarrow aAS$ (2)

$A \rightarrow bs$ (3)

Given variables (Non-terminals) are $\{S, A\}$

Terminals are $\{a, b\}$

Production rules are $S \rightarrow a$
$S \rightarrow aAS$
$A \rightarrow bs$

Consider production rule (2)

		from
S	$\rightarrow aAS$	
S	$\rightarrow absS$	$\rightarrow (3)$
S	$\rightarrow abaaAS$	$\rightarrow 1 \& 2$
S	$\rightarrow abaabSS$	$\rightarrow 3$
S	$\rightarrow abaabaa$	$\rightarrow 1$

Finally $W = abaabaa$ string is derived

Derive LDT and RDT for the following aabaa string

$S \rightarrow aAS \mid aSS \mid \varepsilon$

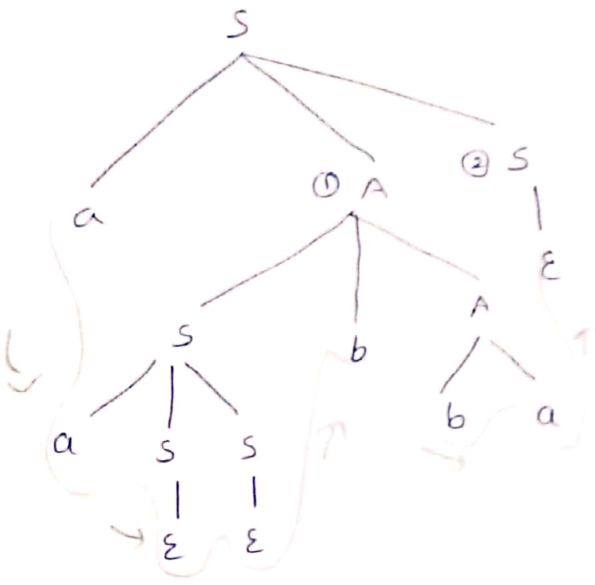
$A \rightarrow SbA \mid ba$

Given: Variables are $\{S, A\}$

Non Terminals are $\{a, b\}$

Production rules are $S \rightarrow aAS$
$S \rightarrow aSS$
$S \rightarrow \varepsilon$
$A \rightarrow SbA$
$A \rightarrow ba$

Left Derivation Tree (LDT) is obtained by applying production rules to left most variable in each step

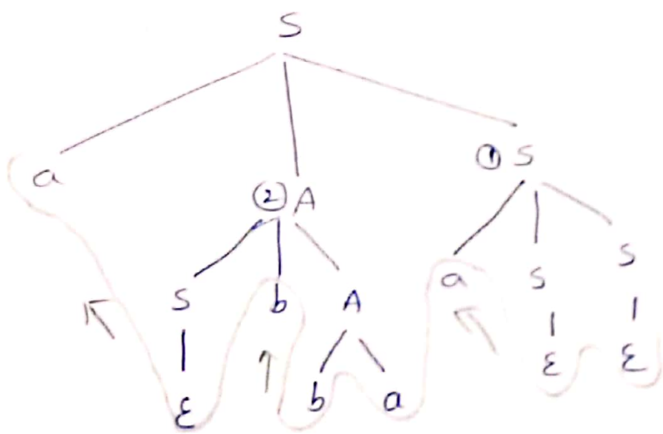


Final string is

aabba

LDT

Right Derivation Tree (RDT) is obtained by applying productions to right most variable in each step.



Final string is

aabba

3. Derive string ab from production rules

$S \rightarrow AB$
$A \rightarrow a$
$B \rightarrow b$

Given :- Non terminals are $\{S, A, B\}$

Terminals are $\{a, b\}$

Production rules are

$S \longrightarrow AB \rightarrow (1)$

$A \rightarrow a \rightarrow (2)$

$B \rightarrow b \rightarrow (3)$

$S \rightarrow AB$

$S \rightarrow aB \quad \rightarrow (2)$

$S \rightarrow ab \quad \rightarrow (3)$

string ab is derived

Derive $a^m b^n$ from production rules $S \rightarrow AB$
$A \rightarrow aA/a$
$B \rightarrow bB/b$

In grammar

Given Non terminals are $\{S, A, B\}$

Terminals are $\{a, b\}$

Production rules are

$S \rightarrow AB \quad \rightarrow (1)$

$A \rightarrow aA \quad \rightarrow 2$

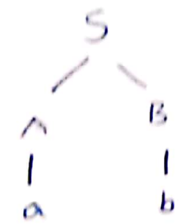
$A \rightarrow a \quad \rightarrow 3$

$B \rightarrow bB \quad \rightarrow 4$

$B \rightarrow b \quad \rightarrow 5$

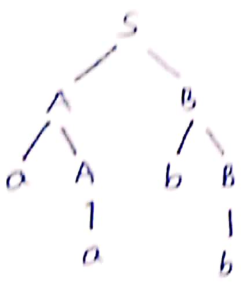
case - 1

consider



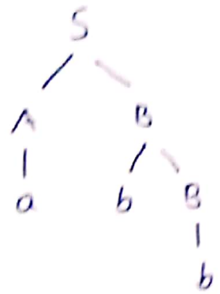
ab

case - 2



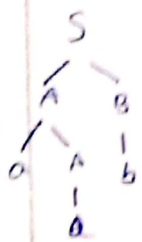
a^2b^2

or



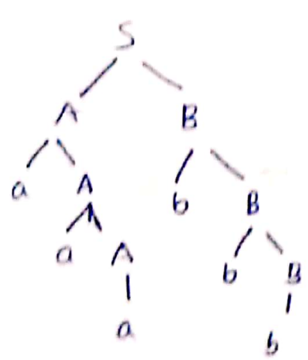
ab^2

or



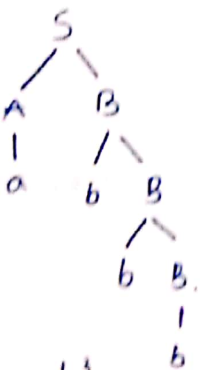
a^2b

case - 3



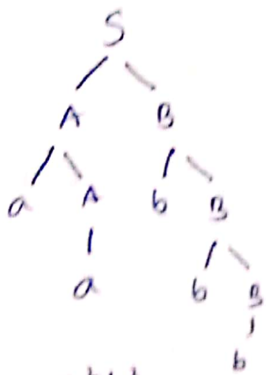
a^3b^3

or



ab^3

or



a^2b^3

Similarly we can derive multiple strings

General form will be $a^m b^n$ where $m, n \geq 0$

5 Show that context free grammar $S \to SS/a/b$ is ambiguous

Given grammar consists

Non terminals are $\{S\}$

Terminals are $\{a, b\}$

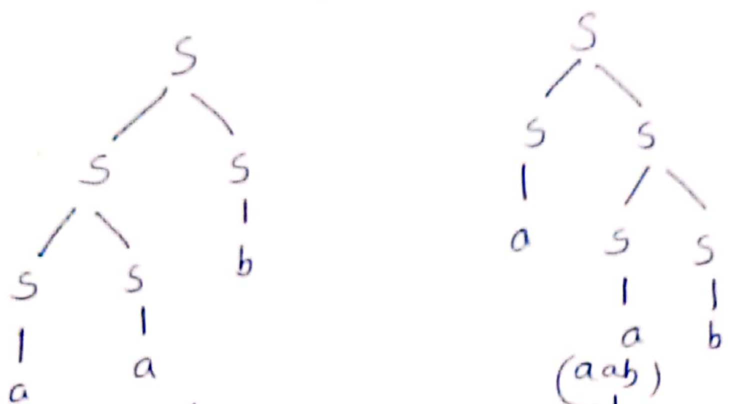
Production rules are $S \to SS$ →1

$S \to a$ →2

$S \to b$ →3

Ambiguous grammar: A grammar is said to be ambiguous if it is having more than one form

To show: Given grammar is ambiguous



(a ab) one form 1st

(aab) 2nd form

∴ string aab can be expressed in 2 forms so given grammar is ambiguous

Hence showed

Find whether given grammar is ambiguous or not

(1) $S \rightarrow \varepsilon$

$S \rightarrow aSb$

$S \rightarrow bSa$

$S \rightarrow SS$

2) $S \rightarrow AA$

$A \rightarrow AAA$

$A \rightarrow bA$

$A \rightarrow Ab$

$A \rightarrow a$

3) $S \rightarrow A$

(1) Given grammar contains

Non terminals $\{S\}$

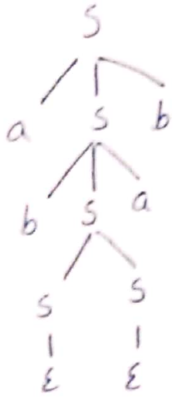
Terminals $\{a, b\}$

Production rules are $S \rightarrow \varepsilon$

$S \rightarrow aSb$

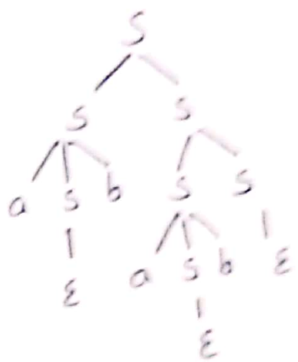
$S \rightarrow bSa$

$S \rightarrow SS$



string is abab

1st form



string is abab

2nd form

Hence given grammar is ambiguous

(2) Given grammar contains

Non terminals are $\{S, A\}$

Terminals are $\{a, b\}$

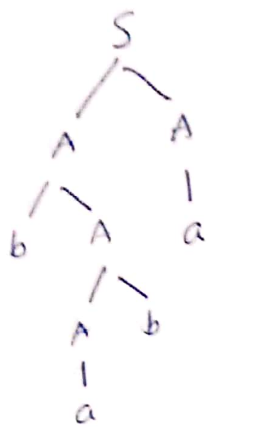
Production rules are $S \rightarrow AA$

$A \rightarrow AAA$

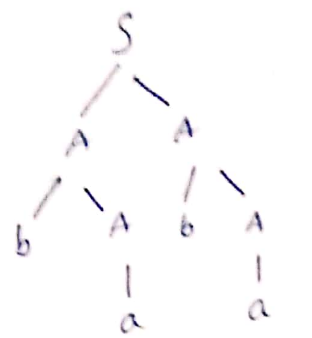
$A \rightarrow bA$

$A \rightarrow Ab$

$A \rightarrow a$



string is baba
1st form



String is baba
2nd form

∴ Given grammar is ambiguous

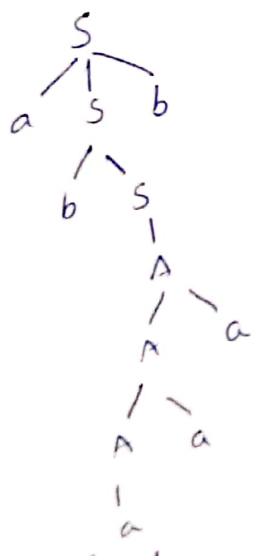
(c) Given grammar contains

$S \rightarrow A$

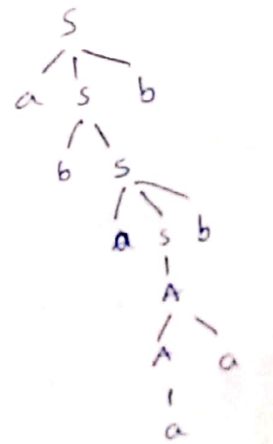
Non terminals are $\{S, A\}$

Terminals are $\{a, b\}$

Production rules are
$S \rightarrow A$
$S \rightarrow aSb$
$S \rightarrow bS$
$A \rightarrow Aa$
$A \rightarrow a$



ab aab



abaaabb

Hence ~~str~~ grammar is unambiguous.

4/ 18001A05[illegible]

Eliminate ε-production rules if any

1) $S \rightarrow Abac$

$A \rightarrow Bc$

$B \rightarrow b / \varepsilon$

$C \rightarrow D | \varepsilon$

$D \rightarrow d$

Given grammar contains

Non terminals are {S, A, B, C, D}

Terminals are {a, b, c, d}

Production rules are

$S \rightarrow Abac$ → 1

$A \rightarrow Bc$ → 2

$B \rightarrow b$ → 3

$B \rightarrow \varepsilon$ → 4

$C \rightarrow D$ → 5

$C \rightarrow \varepsilon$ → 6

$D \rightarrow d$ → 7

Steps:

1) Identify Nullable variables

Nullable variables are A, B, C

2) Create two versions one with nullable variables and other without nullable variables

with	without
$A \rightarrow BC$	$S \rightarrow Abac$
$B \rightarrow \varepsilon$	$B \rightarrow b$
$C \rightarrow \varepsilon$	$C \rightarrow D$
	$D \rightarrow d$
	$A \rightarrow BC$

3) Remove ε production then we get

S → bac

A → B/~~c~~/Bc A → Bc/c/B

B → b

C → D

D → d

4) S → as/bs/ε

Given grammar contains

Non terminals are S

Terminals are a,b

Production rules are S → as

S → bs

S → ε

Steps:

1) Identify Nullable variables S

2) Two versions one with Nullable variable and other without Nullable variables are

with	without
S → ε	S → as
	S → bs

3) Eliminate ε production we get

S → a/b

3) S → ABAC

A → aA/ε

B → bB/ε

C → c

Given grammar contains 18001A0545

Non-terminals are {S, A, B, C}

Terminals are {a, b, c}

Production rules are

$S \rightarrow ABAc$

$A \rightarrow aA \mid \varepsilon$

$B \rightarrow bB \mid \varepsilon$

$C \rightarrow c$

Steps:

1) Identify Nullable variables {A, B}

2) Two versions one with Nullable variables and other without nullable variables in RHS side

with	without
$A \rightarrow \cancel{aA}$	$C \rightarrow c$
$B \rightarrow \cancel{b}B$	
$S \rightarrow ABAc$	

3) Eliminate ε - production we get

$S \rightarrow C$

$A \rightarrow a$

$B \rightarrow b$

$C \rightarrow c$

4) $S \rightarrow bEf$

$E \rightarrow bEc$

$E \rightarrow GGc$

$G \rightarrow b$

$G \rightarrow KL$

$K \rightarrow ckd$

$K \rightarrow \varepsilon$

$L \rightarrow dLe$

$L \rightarrow \varepsilon$

Given grammar contains

Non terminals are {S, E, G, K, L}

Terminals are {b, c, d, e, f}

Production rules are
S → bEf
E → bEc
E → GGc
G → b
G → KL
K → ckd
K → ε
L → dLe
L → ε

Steps:

1) Identify Nullable variables {G, K, L}

2) Two versions with Nullable variables and without Nullable variables on right side

with	without
G → KL	S → bEf
K → ckd	E → bEc
L → dLe	E → GGc
	E → b

3) Eliminate ε- productions we get

S → bEf / bf
E → bEc / bc
E → GGc / c/bc/bbc
E → b
K → cd
L → de

18001A0545

$S \to eSe$
$S \to GH$
$G \to cGb$
$G \to \varepsilon$
$H \to JHd$
$H \to \varepsilon$
$J \to bJ$
$J \to f$

Given grammar contains
Non terminals are $\{S, G, H, J\}$
Terminals are $\{e, b, c, d, f\}$
Production rules are
$S \to eSe$
$S \to GH$
$G \to cGb$
$G \to \varepsilon$
$H \to JHd$
$H \to \varepsilon$
$J \to bJ$
$J \to f$

Steps:

1) Identifying Nullable variables
$\{S, G, H\}$

2) Two versions one with Nullable variables and other without nullable variables on right side

with	without
$S \to GH$	$J \to bJ$
$G \to cGb$	$J \to f$
$H \to JHd$	
$S \to eSe$	

3) Eliminate ε production we get

$S \to ee$
$G \to cb$
$H \to fd$
$J \to bf$
$J \to f$

Scanned with CamScanner

Removal of unit production.

1) $S \rightarrow Aa / B$

$B \rightarrow A / bb$

$A \rightarrow a / bc / B$

Given grammar contains

Non terminals are $\{S, A, B\}$

Terminals are $\{a, b, c\}$

Production rules are $S \rightarrow Aa/B$

$B \rightarrow A/bb$

$A \rightarrow a/bc/B$

Unit production	Non unit production
$S \rightarrow B$	$S \rightarrow Aa$
$B \rightarrow A$	$B \rightarrow bb$
$A \rightarrow B$	$A \rightarrow a/bc$

$S \rightarrow B \rightarrow A$: $\cancel{a}$, bb

$S \rightarrow B$: A (a, bc, B ✗), bb

$B \rightarrow A$: a, bc, B ✗

$A \rightarrow B$: A ✗, bb

$S \rightarrow Aa/bb/a/bc$

$B \rightarrow bb/a/bc$

$A \rightarrow a/bc/bb$

$\therefore$ $S \rightarrow Aa/bb/a/bc$

$B \rightarrow bb/a/bc$

$A \rightarrow a/bc/bb$

8) $S \rightarrow CBA$

$S \rightarrow B$

$A \rightarrow CB$

$A \rightarrow Abbs$

$B \rightarrow aaa$

Given grammar contains

Variables (Non-terminals) are {S, A, B, C}

Terminals are {a, b}

Production rules are
$S \rightarrow CBA$
$S \rightarrow B$
$A \rightarrow CB$
$A \rightarrow Abbs$
$B \rightarrow aaa$

Unit production	Non unit production
$S \rightarrow B$	$S \rightarrow CBA$
	$A \rightarrow CB$
	$A \rightarrow Abbs$
	$B \rightarrow aaa$

$S \rightarrow B \rightarrow aaa$

$S \rightarrow CBA$

$S \rightarrow aaa$

$A \rightarrow CB$

$A \rightarrow Abbs$

$B \rightarrow aaa$

3) $S \rightarrow AB$

$A \rightarrow a$

$B \rightarrow C/b$

$C \rightarrow D$

$D \rightarrow E/bc$

$E \rightarrow d/Ab$

Given grammar contains

Non terminals are $\{S, A, B, C, D, E\}$

Terminals are $\{a, b, c, d\}$

Production rules are

$S \rightarrow AB$
$A \rightarrow a$
$B \rightarrow C/b$
$C \rightarrow D$
$D \rightarrow E/bc$
$E \rightarrow d/Ab$

Unit production	Non unit production
$B \rightarrow C$	$S \rightarrow AB$
$C \rightarrow D$	$A \rightarrow a$
$D \rightarrow E$	$B \rightarrow b/d/Ab$
	$D \rightarrow bc$
	$E \rightarrow d/Ab$

$B \rightarrow C \rightarrow D \rightarrow E \rightarrow d,\ Ab$

$C \rightarrow D \rightarrow E \rightarrow d,\ Ab$

$D \rightarrow E \rightarrow d,\ Ab$

So final productions are

$S \rightarrow AB$
$A \rightarrow a$
$B \rightarrow b/d/Ab$
$C \rightarrow d/Ab$
$D \rightarrow bc/d/Ab$
$E \rightarrow d/Ab$

Removal of useless symbols:

1) $S \rightarrow ABC \mid BaB$

$A \rightarrow aA \mid BaC \mid aaa$

$B \rightarrow bBb \mid a$

$C \rightarrow CA \mid AC$

Given grammar contains
Non terminals are {S, A, B, C}
Terminals are {a, b}
Production rules are $S \rightarrow ABC \mid BaB$
$A \rightarrow aA \mid BaC \mid aaa$
$B \rightarrow bBb \mid a$
$C \rightarrow CA \mid AC$

Rules:

1) Remove symbols not deriving terminal string
2) Remove symbols non reachable from S.

useless symbols are {C, S, A} as C can't derive any terminal string and $S \rightarrow ABC$ / $A \rightarrow BaC$ as it contain C.

After removing useless symbols production rules are

$S \rightarrow BaB$

$A \rightarrow aA \mid \cancel{BaC} \mid aaa$

$B \rightarrow bBb \mid a$

2) $S \rightarrow Sa \mid A \mid C$

$A \rightarrow a$

$B \rightarrow bb$

$C \rightarrow aC$

Ans) Given grammar contains
Non terminals are {S, A, B, C}
Terminals are {a, b}
Production rules are $S \rightarrow Sa \mid A \mid C$
$A \rightarrow a$
$B \rightarrow bb$
$C \rightarrow aC$

Useless Symbols are {S, C, B}

After removing useless symbols we get

$S \rightarrow Sa \mid A$

$A \rightarrow a$

Use 3 rules to simplify given grammar.

$S \to aA \mid bB \mid a$

$A \to aB \mid B \mid \varepsilon$

$B \to b \mid \varepsilon$

Given grammar contains

Non terminals are $\{S, A, B\}$

Terminals are $\{a, b\}$

Production rules are $S \to aA \mid bB \mid a$

$A \to aB \mid B \mid \varepsilon$

$B \to b \mid \varepsilon$

Rule 1:

1) Removal of ε rules.

a) Nullable variables are $\{A, B\}$

b) 2 versions with & without Nullable variables on RHS

with	without
~~$A \to B$~~	$S \to a$
~~$A \to aB$~~	$A \to B$
$S \to aA$	$S \to aA$
$S \to bB$	$S \to bB$
	$A \to aB$

c) Removal of ε productions we get $S \to a \mid b \mid a$

$A \to a$

$B \to b$

Rule 2:-

~~And~~ there is no need of using ~~other two~~ second rule since there are no unit production

Rule 3: Removal of useless symbols

Since the production 'S' is not having variables A, B we can eliminate them. After eliminating we get production rule as

$S \to a/b$

Normal forms :-

18001A0545

Find CNF and GNF equivalent to following grammar

1) $S \to S \wedge S$
$S \to S \vee S$
$S \to \neg S$
$S \to (S)$
$S \to p$
$S \to q$

2) $E \to E + E$
$E \to E * E$
$E \to (E)$
$E \to a$

1) Given grammar contains Non terminals are $\{S\}$

Terminals are $\{\wedge, \vee, \neg, (,), p, q\}$

To find CNF we make of assigning new variables to terminals

let $x_1 = \wedge$, $x_2 = \vee$, $x_3 = \neg$, $x_4 = ($, $x_5 =)$

Then production rules are

$S \to S x_1 S$ → Not in CNF ①
$S \to S x_2 S$ → Not in CNF ②
$S \to x_3 S$ ✓ ③
$S \to x_4 S x_5$ → Not in CNF ④
$S \to p$ ✓ ⑤
$S \to q$ ✓ ⑥

Rules (or) Conditions for GNF
1) $A \to BC$
2) $A \to a$

Production rules 1, 2, 4 are not in CNF we convert them into CNF

$S \to S x_1 S \to x_6 S$

$x_6 \to S x_1$

$S \to S x_2 S \to x_7 S$

$x_7 \to S x_2$

$S \to x_4 S x_5 \to x_8 x_5$

$x_8 \to x_4 S$

Finally CNF is

$S \rightarrow X_6 S$

$S \rightarrow X_7 S$

$X_6 \rightarrow S X_1$

$X_7 \rightarrow S X_2$

$S \rightarrow X_8 X_5$

$X_8 \rightarrow X_4 S$

$X_1 \rightarrow \wedge$

$X_2 \rightarrow \vee$

$X_3 \rightarrow \neg$

$X_4 \rightarrow ($

$X_5 \rightarrow)$

$S \rightarrow X_3 S$

$S \rightarrow P$

$S \rightarrow Q$

Converting to GNF

The given grammar is in CNF and there is no left recursion

Condition

$A \rightarrow a\alpha \quad \alpha \in N^*$

$S \rightarrow \varepsilon$

$S \rightarrow X_6 S \mid X_7 S \mid X_8 X_5 \mid X_3 S$

$X_6 \rightarrow S X_1$

$X_7 \rightarrow S X_2$ are not in GNF

$X_8 \rightarrow X_4 S$

Substitute $X_4 \rightarrow ($, $X_3 \rightarrow \neg$ so we get $X_8 \rightarrow (S$, $S \rightarrow \neg S$

Substitute $X_8 \rightarrow (S$ in $S \rightarrow X_8 X_5$ we get

$S \rightarrow X_6 S \mid X_7 S \mid (S X_5 \mid \neg S$

$X_8 \rightarrow (S$

$X_6 \rightarrow S X_1$

$X_7 \rightarrow S X_2$

Now $S \to X_6 S \,/\, X_7 S$

$X_6 \to S X_1$

$X_7 \to S X_2$

Now $\cancel{X_6} \to \cancel{X_7 S} X_1 \,/\, \cancel{X_7 S} X_1$

$\cancel{X_7} \to \neg S X_2 \,/\, (S X_5 X_2$

$S \to \neg S X_2 S \,/\, (S X_5 X_2 S \,/\, \neg S X_1 S \,/\, (S X_5 S_1 S$

$X_6 \to \neg S X_1 \,/\, (S X_5 S_1$

GNF is

$S \to \neg S X_2 S \,/\, (S X_5 X_2 S \,/\, \neg S X_1 S \,/\, (S X_5 S_1 S$

$X_1 \to \wedge,\ X_2 \to \vee,\ X_3 \to \neg,$

$X_4 \to (,\ X_5 \to),\ S \to P,$

$S \to q,\ S \to (S X_5 \,/\, \neg S$

$X_7 \to \neg S X_2 \,/\, (S X_5 X_2$

$X_6 \to \neg S X_1 \,/\, (S X_5 S_1$

2) $E \to E + E$

$E \to E * E$

$E \to (E)$

$E \to a$

Given grammar contains Non terminals $\{E\}$

Terminals are $\{+, *, (,), a\}$

To convert CNF

$E \to E + E$

$E \to E * E$

$E \to (E)$

are not in CNF

So, to convert substitute terminals with new variables

as $X_1 \to +$ $\quad X_3 \to ($ $\quad$ we get

$X_2 \to *$ $\quad X_4 \to)$

$E \to E X_1 E$ $\quad X_1 \to +$ $\quad X_4 \to)$

$E \to E X_2 E$ $\quad X_2 \to *$ $\quad E \to a$

$E \to X_3 E X_4$ $\quad X_3 \to ($

$E \to E X_1 E$ — $E \to X_5 E$, $X_5 \to E X_1$

$E \to E X_2 E$ — $E \to X_6 E$, $X_6 \to E X_2$

$E \to X_3 E X_4$ — $E \to X_7 X_4$, $X_7 \to X_3 E$

∴ Final production rules in CNF are

$E \to X_5 E$	$E \to X_7 X_4$
$X_5 \to E X_1$	$X_7 \to X_3 E$
$X_1 \to +$	$X_3 \to ($
$E \to X_6 E$	$X_4 \to)$
$X_6 \to E X_2$	$E \to a$
$X_2 \to *$	

<u>GNF:</u>

$E \to E + E$
$E \to E * E$
$E \to (E)$
$E \to a$

} Not in GNF. but converted present

$E \to X_5 E$.	$E \to X_7 X_4$
$X_5 \to E X_1$	$X_7 \to X_3 E$.
$X_1 \to +$.	$X_3 \to ($.
$E \to X_6 E$.	$X_4 \to)$.
$X_6 \to E X_2$.	$E \to a$.

So we get GNF as

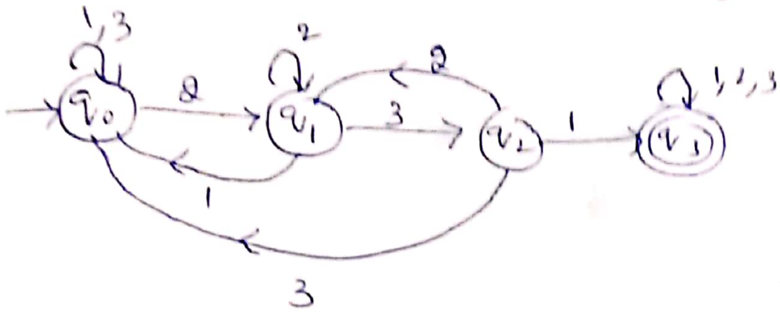
$X_1 \to +$	$X_7 \to (E$	$X_5 \to a X_1$
$X_4 \to)$	$X_6 \to a X_2$	$E \to a X_1 E$
$E \to a$	$E \to a X_2 E$	$E \to (E)$
$X_3 \to ($		

Finite Automata:

18B01A0545

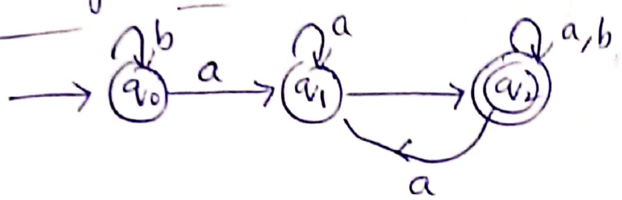
1) construct DFSA which accepts set of strings in $\{1,2,3\}^*$ containing $\{2,3,1\}$ as substring

$L = \{231, 1231, 2231, 3231, 2311 \ldots\}$



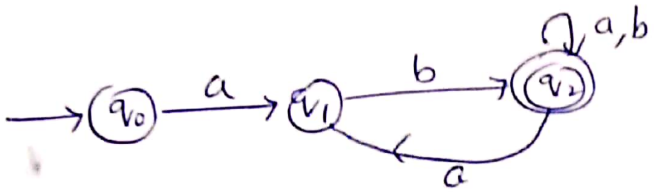
2) DFA which accepts set of strings containing $\{ab\}$ as substring, containing $\{2,3,1\}$ as substring.

substring ab

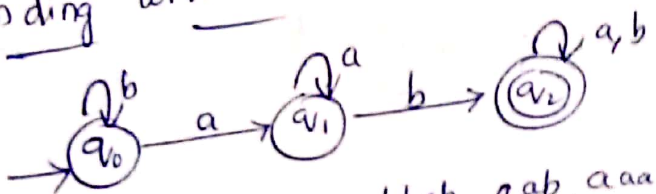


starting with ab

$L = \{ab, abaa, abb, \ldots\}$



Ending with ab

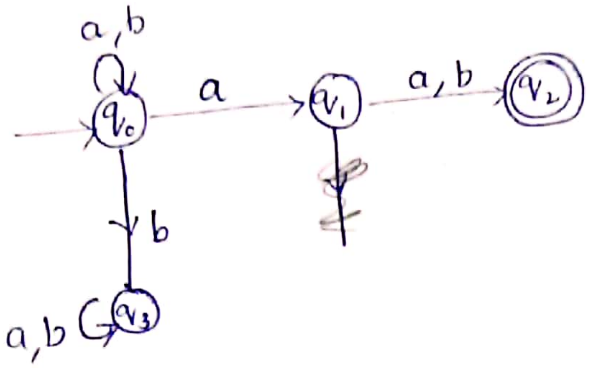


$L = \{ab, bab, bbab, aab, aaab \ldots\}$

DFSA and NFSA

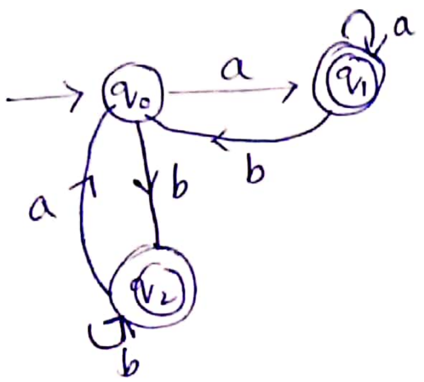
Construct DFSA whose string accepts ~~third~~ second symbol from RHS is 'a'.

L = { aa, ab, aab, bab, ... infinite ... }



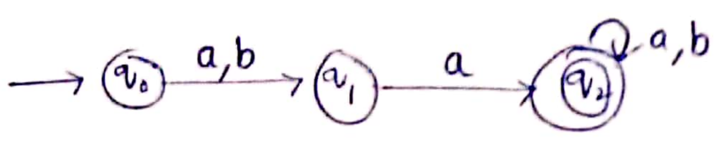
Construct DFA which accepts set of strings which starts and ends with same symbols.

L = { a, b, aa, bb, aba, bab, aaa, bbb ... }



Construct NFA and convert it to DFA for language whose strings contains second symbol from RHS is 'a'.

L = { aa, ba, baa, aaa, ... }



12

1800IA0545

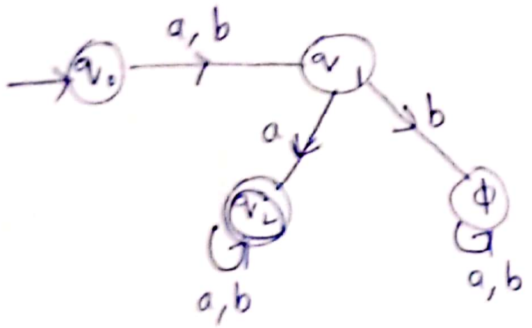
1) Construct state transition table for NFSA

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

2) State transition table for DFA

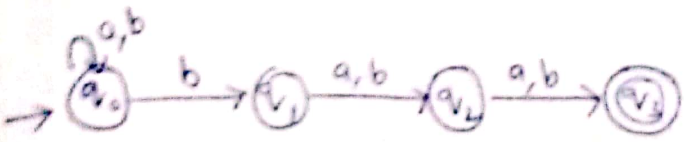
	a	b
$\to q_0$	q_1	q_1
q_1	q_2	ϕ
$* q_2$	q_2	q_2
D	D	D

3) Draw DFSA



Construct NFA and convert it to DFA for the language whose strings accepts 3rd symbol from RHS is b

$L = \{$ baa, bbb, abaa, bbab . $\}$

NFA $\to$ 

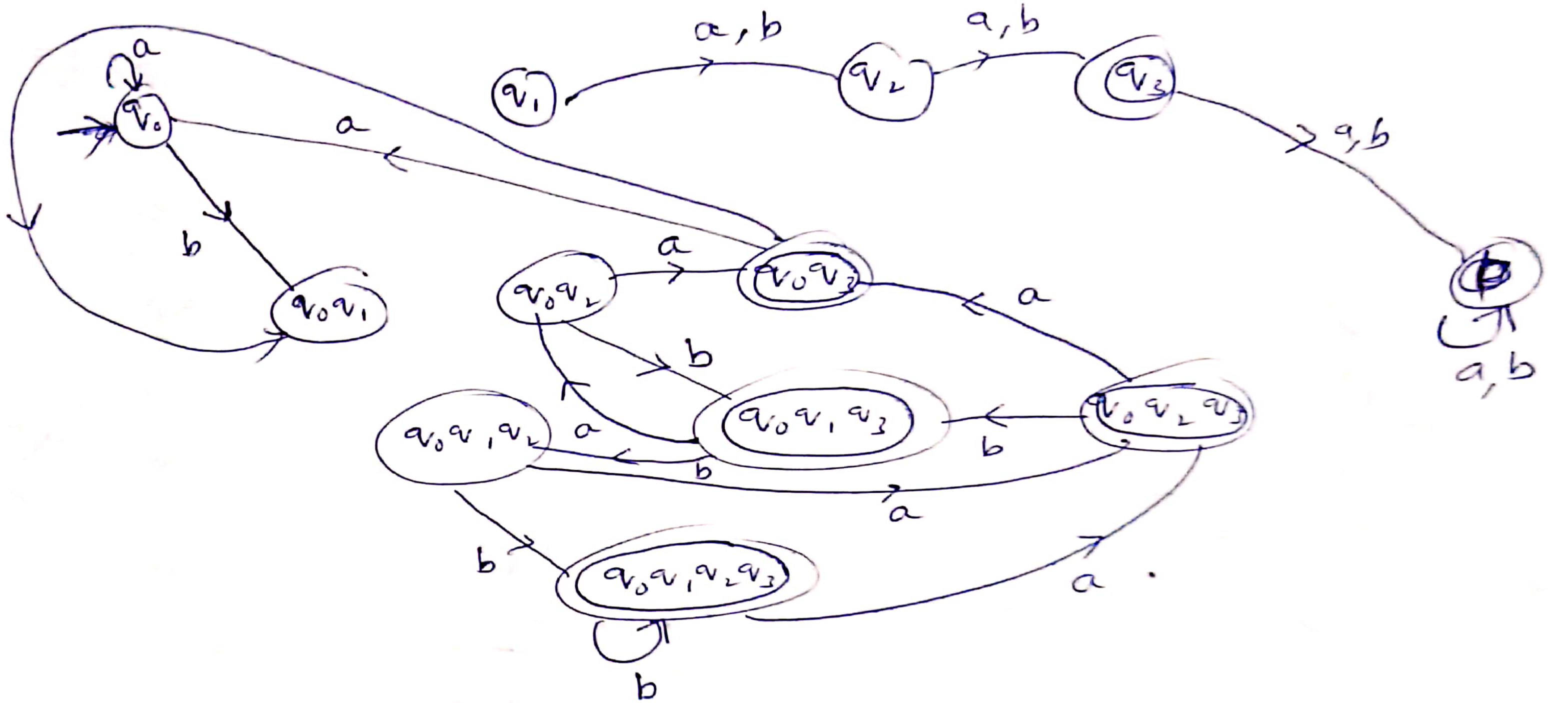
State transition table for NFA:

	a	b
q_0	q_0	$\{q_0 q_1\}$
q_1	q_2	q_2
q_2	q_3	q_3
q_3	D	D

State transition table for DFA:

		a	b
→	q_0	q_0	$[q_0 q_1]$
	q_1	q_2	q_2
	q_2	q_3	q_3
*	q_3	ϕ	ϕ
	$q_0 q_1$	$q_0 q_2$	$[q_0 q_1 q_2]$
	D	ϕ	ϕ
	$q_0 q_2$	$q_0 q_3$	$q_0 q_1 q_3$
	$q_0 q_1 q_2$	$q_0 q_2 q_3$	$q_0 q_1 q_2 q_3$
*	$q_0 q_3$	q_0	$q_0 q_1$
*	$q_0 q_1 q_3$	$q_0 q_2$	$q_0 q_1 q_2$
*	$q_0 q_2 q_3$	$q_0 q_3$	$q_0 q_1 q_3$
*	$q_0 q_1 q_2 q_3$	$q_0 q_2 q_3$	$q_0 q_1 q_2 q_3$

DFSA diagram:-



18001A0545

Construct NFA and convert it to DFA for the language whose strings accepts 3rd symbol from LHS is a

$L = \{ aaa, baa, bba, bbab, aaab \ldots \}$

NFSA:

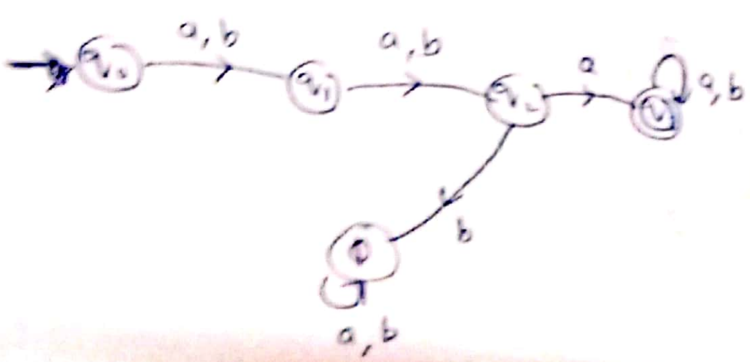
State transition table for NFSA:

	a	b
$\rightarrow q_0$	q_1	q_1
q_1	q_2	q_2
q_2	q_3	ϕ
$* q_3$	q_3	q_3

State transition table for DFSA.

	a	b
$\rightarrow q_0$	q_1	q_1
q_1	q_2	q_2
q_2	q_3	ϕ
$* q_3$	q_3	q_3
ϕ	ϕ	ϕ

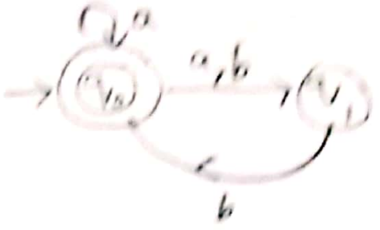
DFSA diagram.



14

18001A0545

Convert the following NFA into DFA using subset construction method

1) 

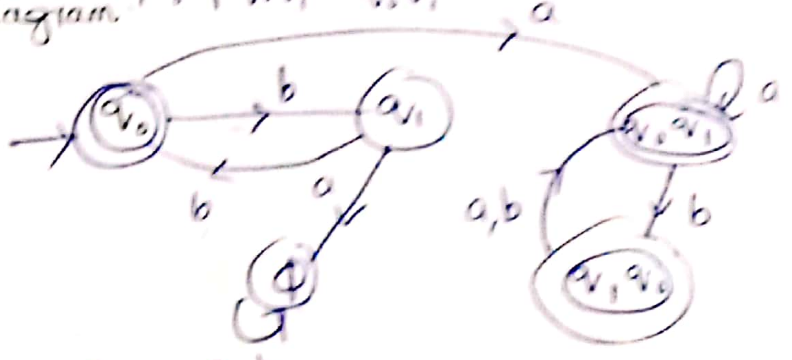
State transition table for NFSA

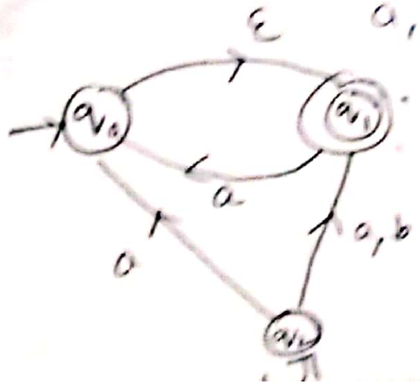
	a	b
q_0	q_0	q_1
q_1	ϕ	q_0

State transition table for DFSA

	a	b
$\rightarrow *\ q_0$	$[q_0,q_1]$	q_1
q_1	ϕ	q_0
ϕ	ϕ	ϕ
$*\ q_0q_1$	q_0q_1	q_1q_0
$*\ q_1q_0$	q_0q_1	q_0q_1

DFSA diagram



2) 

State transition table for NFSA:-

	a	b	ε
→ q_0	ϕ	ϕ	q_1
* q_1	q_0	ϕ	ϕ
q_2	$[q_0 q_1]$	$[q_1 q_2]$	ϕ
$q_0 q_1$			
$q_1 q_2$			

State transition table for DFSA:-

	a	b	ε
→ q_0	ϕ	ϕ	q_1
* q_1	q_0	ϕ	ϕ
q_2	$q_0 q_1$	$q_1 q_2$	ϕ
* $q_0 q_1$	q_0	ϕ	q_1
* $q_1 q_2$	$q_0 q_1$	$q_1 q_2$	ϕ
ϕ	ϕ	ϕ	ϕ

DFSA diagram:-