$S \to AB \mid ab \mid aA$

$A \to a \mid \varepsilon$

$B \to b$

Removal of Null Production:

$S \to AB \mid ab \mid aA \mid B \mid a$

$A \to a$

$B \to b$

By removing null productions, we reduce the number of intermediate states

Removal of Unit Production:

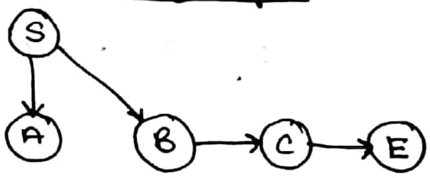
$S \to AB \mid ab \mid aA \mid b \mid a$

$A \to a$

$B \to b$

• <u>Parsing:</u> whether the string is possible for the given language or not

<u>Reachibility Graph:</u>



$S \to AB \mid ab \mid aA \mid B \mid a$

$A \to a$

$B \to b \mid cC$

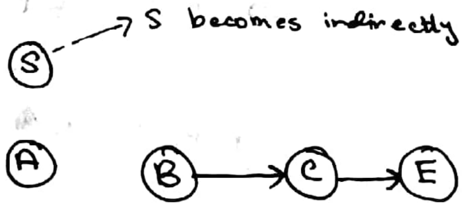
$C \to E$

$S \to AB \mid ab \mid aA$

$A \to a \mid \varepsilon$

$B \to b \mid \varepsilon \mid cC$

$C \to E$



S becomes indirectly nullable

Scanned by CamScanner

Convert to CNF:

1. $S \rightarrow Aa \mid bS \mid \varepsilon$

$A \rightarrow aA \mid bB \mid \varepsilon$

$B \rightarrow aA \mid bC \mid \varepsilon$

$C \rightarrow aC \mid bC$

Remove ε-production —

$S \rightarrow Aa \mid bS \mid a \mid b \mid \varepsilon$

$A \rightarrow aA \mid bB \mid a \mid b$

$B \rightarrow aA \mid bC \mid a$

$C \rightarrow aC \mid bC$

Now, there is no unit production and useless states.

Cannot be done

$S \rightarrow AS$ [$\because S \rightarrow a$ is there]

$S \rightarrow SS$

$S \rightarrow a$

$S \rightarrow b$

$S \rightarrow \varepsilon$

$S \rightarrow AD$	$A \rightarrow DA$	$B \rightarrow DA$	$C \rightarrow DC$
$S \rightarrow ES$	$A \rightarrow EB$	$B \rightarrow EC$	$C \rightarrow EF$
$D \rightarrow a$	$A \rightarrow a \mid b$	$B \rightarrow a$	$F \rightarrow c$
$E \rightarrow b$			
$S \rightarrow \varepsilon$			
$S \rightarrow a \mid b$			

Scanned by CamScanner

2. $S \to aA \mid aBB$

$A \to aAA \mid \varepsilon$

$B \to bB \mid bbC$

$C \to B$

Removing Null Production:

$S \to aA \mid aBB \mid a$

$A \to aAA \mid aA \mid a$

$B \to bB \mid bbC$

$C \to B$

No Unit Productions.

B and C are useless symbols.

$\therefore$ $S \to aA \mid a$

$A \to aAA \mid aA \mid a$

$S \to DA$	$A \to DE$
$D \to a$	$E \to AA$
$S \to a$	$A \to DA$
	$A \to a$

3. $S \to aAbB$

$A \to abAB \mid aAA \mid a$

$B \to bBaA \mid bBB \mid b$

4. $S \to bA \mid aB$

$A \to bAA \mid aS \mid a$

$B \to aBB \mid bS \mid b$

Scanned by CamScanner

Remove unit production –

1) D → a | b | Da | Db | D0 | D1

F → D | (E)

T → F | T * F

E → T | E + T

3.

~~S → aAbB~~

S → XY

~~X → aAb~~

X → aA

Y → bB

X → CA

Y → DB

C → a

D → b

A → EF

E → ab | E → CD

F → AB

A → XA

A → a

B → YX

B → YB

B → b

4.

S → CA

S → DB

C → b

D → a

A → EA

E → CA

~~A → a~~

A → DS

A → a

B → FB

F → DB

B → CS

B → b

Remove unit production

Solution:

1) D → a | b | Da | Db | D0 | D1

F → a | b | Da | Db | D0 | D1 | (E)

T → a | b | Da | Db | D0 | D1 | (E) | T * F

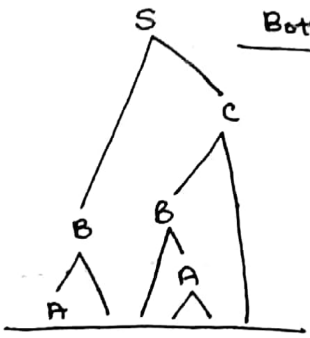
Scanned by CamScanner

$E \rightarrow a \mid b \mid Da \mid Db \mid D0 \mid D1 \mid (E) \mid T*F \mid E+T$ (Ans)

GNF

$X \rightarrow aY \mid b$, $\alpha \in \Sigma, \beta \in \Sigma, Y \in V^* \cup \Sigma$

Bottom-Up



$S \rightarrow BC$
$A \rightarrow ab \mid aa$
$B \rightarrow Ab \mid bA$
$C \rightarrow Ba$

(ab) bbaaa

$S \rightarrow BC$
BBa
$BbAa$ → Top-Down Parsing
$Abbaaa$
$\boxed{abbbaaa}$

CNF
$X \rightarrow AB \mid \alpha$

GNF
$X \rightarrow \alpha A \mid \alpha$

Convert to GNF —

Steps:
1) Convert to CNF
2) Rename all the variables $A \in V^*$ using $A_1, A_2, \dots, A_n$
3) $A_i \rightarrow A_j A_k$ 4) $A_i \rightarrow \beta A_j \mid i < j$
$A_i \rightarrow \alpha$

1. $S \rightarrow XY1 \mid 0$
$X \rightarrow 00X \mid Y \mid 1$
$Y \rightarrow 1X1$

$S \rightarrow XY1 \mid 0$
$X \rightarrow 00X \mid 1X1 \mid 1$
$Y \rightarrow 1X1$

Removing Unit Production

$S \rightarrow XZ$
$Z \rightarrow YW$
$W \rightarrow 1$
$S \rightarrow 0$

$X \rightarrow AX$
$A \rightarrow BB$
$B \rightarrow 0$

$X \rightarrow WP$
$P \rightarrow XW$
$X \rightarrow 1$

$Y \rightarrow WF$

Scanned by CamScanner

~~Y ⇒ W.P~~

i) $A_1 \rightarrow A_2A_3$

ii) $A_3 \rightarrow A_4A_5$

iii) $A_5 \rightarrow 1$

iv) $A_1 \rightarrow 0$

v) $A_2 \rightarrow A_6A_2$

vi) $A_6 \rightarrow A_7A_7$

vii) $A_7 \rightarrow 0$

viii) $A_2 \rightarrow A_5A_8$

ix) $A_8 \rightarrow A_2A_5$

x) $A_2 \rightarrow 1$

xi) $A_4 \rightarrow A_5A_8$

GNF—

i) $A_1 \rightarrow 1A_3$

vi) $A_6 \rightarrow 0A_7$

viii) $A_2 \rightarrow 1A_8$

ix) $A_8 \rightarrow 1A_5$

xi) $A_4 \rightarrow 1A_8$

iii) $A_5 \rightarrow 1$

iv) $A_1 \rightarrow 0$

vii) $A_7 \rightarrow 0$

x) $A_2 \rightarrow 1$

• No of production in CNF $\geq$ No. of productions in GNF

ii) $A_3 \rightarrow 1A_8A_5$

v) $A_2 \rightarrow 0A_7A_2$

2. Convert to GNF—

$S \rightarrow AA \mid a$

$A \rightarrow SS \mid b$

$A_i \rightarrow A_jA_k$

$i < j, k$

OR

$i < j$

$i \not< k$

$\boxed{\begin{array}{l} A_1 \rightarrow A_2A_2 \\ A_1 \rightarrow a \end{array}}$

$A_2 \rightarrow A_1A_1$

$\boxed{A_2 \rightarrow b}$

$A_2 \rightarrow A_2A_2A_1$

This type of production is known as left recursion.

Left Recursion needs to be removed.

Removal of left recursion:

$\boxed{A_2 \rightarrow bA_3}$

$A_3 \rightarrow A_2A_1A_3 \mid \epsilon$

$\boxed{\begin{array}{l} A_3 \rightarrow A_2A_1A_3 \\ A_3 \rightarrow A_2A_1 \end{array}}$

$X \rightarrow X\beta \mid \alpha$

$X \rightarrow X\beta$

$X \rightarrow \alpha A$

$A \rightarrow \beta A \mid \epsilon$

$A \rightarrow \beta A \mid \beta$

~~Y~~ → W.P

i) $A_1 \rightarrow A_2A_3$
ii) $A_3 \rightarrow A_4A_5$
iii) $A_5 \rightarrow 1$
iv) $A_1 \rightarrow 0$
v) $A_2 \rightarrow A_6A_2$
vi) $A_6 \rightarrow A_7A_7$
vii) $A_7 \rightarrow 0$
viii) $A_2 \rightarrow A_5A_8$
ix) $A_8 \rightarrow A_2A_5$
x) $A_2 \rightarrow 1$
xi) $A_4 \rightarrow A_5A_8$

GNF —

i) $A_1 \rightarrow 1A_3$
vi) $A_6 \rightarrow 0A_7$
viii) $A_2 \rightarrow 1A_8$
ix) $A_8 \rightarrow 1A_5$
xi) $A_4 \rightarrow 1A_8$
iii) $A_5 \rightarrow 1$
iv) $A_1 \rightarrow 0$
vii) $A_7 \rightarrow 0$
x) $A_2 \rightarrow 1$

• No of production in CNF $\geq$ No. of productions in GNF

ii) $A_3 \rightarrow 1A_8A_5$
v) $A_2 \rightarrow 0A_7A_2$

2. Convert to GNF —

$S \rightarrow AA \mid a$
$A \rightarrow SS \mid b$

$A_i \rightarrow A_jA_k$
$i < j, k$
OR
$i < j$
$i \not< k$

$A_1 \rightarrow A_2A_2$
$A_1 \rightarrow a$

$A_2 \rightarrow A_1A_1$
$A_2 \rightarrow b$

$A_2 \rightarrow A_2A_2A_1$

This type of production is known as left recursion.
Left Recursion needs to be removed.

Removal of left recursion:
$A_2 \rightarrow bA_3$
$A_3 \rightarrow A_2A_1A_3 \mid \epsilon$

$A_3 \rightarrow A_2A_1A_3$
$A_3 \rightarrow A_2A_1$

$X \rightarrow X\beta \mid \alpha$
$X \rightarrow X\beta$
$X \rightarrow \alpha A$
$A \rightarrow \beta A \mid \varepsilon$
$A \rightarrow \beta A \mid \beta$

GNF —

$^{4}A_1 \rightarrow bA_2$

$^{4}A_1 \rightarrow a$

$^{4}A_2 \rightarrow b$

$^{4}A_2 \rightarrow bA_3$

$A_2 \rightarrow bA_3 \begin{cases} A_3 \rightarrow bA_1A_3 \\ A_3 \rightarrow bA_3A_1A_3 \end{cases}$

$A_3 \rightarrow A_2A_1 \begin{cases} A_3 \rightarrow bA_1 \\ A_3 \rightarrow bA_3A_1 \end{cases}$

Sequence Detector

Overlapped Sequence Detector
i/p — 1 0 1 0 0 0 1 0 1 0 1 0 0 0 1 0 0 1 0 $\boxed{1010}$
o/p — 0 0 0 1 0 0 0 0 1 0 1 0 0 0 0 0 0 0

If $\boxed{011}$, then non-overlapped.

JK FF —

J	K	Present State	Next State
0	0	0 1	0 1
0	1	0 1	0 0
1	0	0 1	1 1
1	1	0 1	1 0

- Design a sequence detector to detect the sequence 11011 using JK Flip Flop. (Overlapping is accepted)

0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1

Overlapping → 1 1 1 1

Non Overlapping → 1 1

Scanned by CamScanner

Step 1: Calculate the number of states.

∴ N = 6,

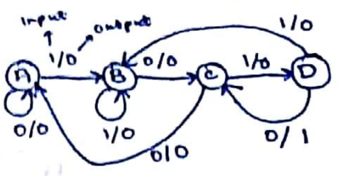
\# States = 5 in ~~10~~11011

Example:

1010

101010

10101010

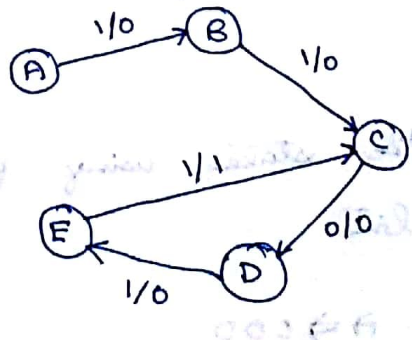


∴ No of bits = No. of States

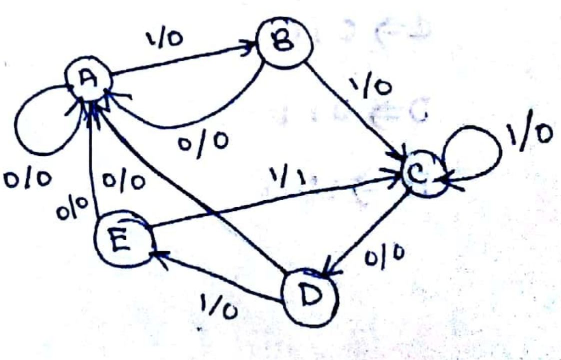
Step 2: Name the states and describe what for the states are waiting and which sequence they have already met.

	already read	waiting for
A	–	11011
B	1	1011
C	11	011
D	110	11
E	1101	1

Step 3: Draw the transition diagram with the acceptable sequence only.



Step 4: Include all the other transitions / breaking sequence



Step 5: Draw the state table

	0		1	
A	A	0	B	0
B	A	0	C	0
C	D	0	C	0
D	A	0	E	0
E	A	0	C	1

- 3 bits are required generate 5 states.

∴ 3 Flip Flops are requ

Step 6: Derive the Number of Flip Flops.

$$2^{p-1} \leq N \leq 2^p$$ [p = no. of FFs required]

N = 5

p = 3 → No. of flip flops.

Step 7: Rename the states using p number of binary bits.

A ⇒ 000

B ⇒ 001

C ⇒ 010

D ⇒ 011

E ⇒ 100

Scanned by CamScanner

Notice that in state table, for 0 input, output states are A or D. And for input 1, output states are B, C or E.

$\therefore$ we divide the states in an appropriate manner.

A, D $\Rightarrow$ even number

B, C, E $\Rightarrow$ odd number

A $\Rightarrow$ 000

B $\Rightarrow$ 001

C $\Rightarrow$ 011

D $\Rightarrow$ 100

E $\Rightarrow$ 101

- Rewrite the state ~~sit~~ table using the new names.

Scanned by CamScanner