D Y PATIL
DEEMED TO BE
UNIVERSITY
RAMRAO ADIK
INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

Theoretical Computer Science

Unit 3: Grammars

Faculty Name : Ms. Namita Pulgam

Index

Lecture No 25:

Conversion of CFG to CNF

Chomsky Hierarchy

- There are four types of grammars − Type 0, Type 1, Type 2 and Type 3.
- The following table shows how they differ from each other :

Grammar Type	Grammar Accepted	Language Accepted	Automaton
Type 0	Unrestricted grammar	Recursively enumerable language	Turing Machine
Type 1	Context-sensitive grammar	Context-sensitive language	Linear-bounded automaton
Type 2	Context-free grammar	Context-free language	Pushdown automaton
Type 3	Regular grammar	Regular language	Finite state automaton

Type 3

- Generates regular languages.
- Grammar must have a single non-terminal on the left-hand side and a right-hand side consisting of a single terminal or single terminal followed by a single non-terminal.
- The productions must be in the form **X ⟶ a or X ⟶ aY**

  where **X, Y ∈ N** (Non terminal) and **a ∈ T** (Terminal).
- The rule **S ⟶ ε** is allowed if **S** does not appear on the right side of any rule.
- **Example :**

  1. X ⟶ ε
  2. X ⟶ a | aY
  3. Y ⟶ b

D Y PATIL DEEMED TO BE UNIVERSITY
RAMRAO ADIK INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

Type 2

- Generates context-free languages.
- The productions must be in the form **A** $\longrightarrow$ **γ**

  where **A ∈ N** (Non terminal) and **γ ∈ (T ∪ N)*** (String of terminals and non-terminals).
- Languages generated by these grammars are be recognized by a non-deterministic pushdown automaton.
- **Example :**

  1. S $\longrightarrow$ X a
  2. X $\longrightarrow$ a
  3. X $\longrightarrow$ aX
  4. X $\longrightarrow$ abc
  5. X $\longrightarrow$ ε

Type 1

- Generates context-sensitive languages.
- The productions must be in the form

$$\alpha A \beta \longrightarrow \alpha \gamma \beta$$

where $A \in N$ (Non-terminal) and $\alpha, \beta, \gamma \in (T \cup N)^*$ (Strings of terminals and non-terminals).

- The strings α and β may be empty, but γ must be non-empty.
- The rule $S \longrightarrow \varepsilon$ is allowed if S does not appear on the right side of any rule.
- The languages generated by these grammars are recognized by a linear bounded automaton.
- **Example :**

1. $B \longrightarrow AbBc$
2. $A \longrightarrow bcA$
3. $B \longrightarrow b$

Type 0

- Generates recursively enumerable languages.
- The productions have no restrictions.
- They are any phase structure grammar including all formal grammars.
- They generate the languages that are recognized by a Turing machine.
- The productions can be in the form of $\alpha \longrightarrow \beta$

  where, α is a string of terminals and non-terminals with at least one non-terminal and α cannot be null and β is a string of terminals and non-terminals.
- **Example :**

  1. $S \longrightarrow ACaB$
  2. $Bc \longrightarrow acB$
  3. $CB \longrightarrow DB$
  4. $aD \longrightarrow Db$

D Y PATIL DEEMED TO BE UNIVERSITY
RAMRAO ADIK INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

Normal Forms

- Context free grammars can be written in certain standard forms known as normal forms.
- These normal forms impose certain restrictions on the productions in the CFG.
- Complex CFG can be reduced to simple forms after modifying them or rewriting them using these normal forms.
- Two normal forms are :

  1. Chomsky Normal Form (CNF)
  2. Greibach Normal Form (GNF)

Chomsky Normal Form

- Any Context free language without Є and which is generated by a grammar in which all productions are of the form A → BC or A → a where A, B and C are non terminals and 'a' is a terminal symbol is said to be in Chomsky Normal Form.
- We can have either two non terminals or a single terminal on the RHS of every production.
- If the language has an empty string i.e. Є then only the following Є production is allowed in CNF.

S → Є where S is the start symbol

Conversion of CFG to CNF

- Eliminate useless symbols, unit productions and ϵ productions.
- For every production of the form $A \rightarrow \alpha$ where $|\alpha| >= 2$ replace terminals of non CNF productions by some variables (i.e. non terminals) and add new productions for these variables.
- Every non CNF production deriving more than two non terminals can be broken into cascade of productions each deriving a string of two non terminals.

Examples:

1. **A → PQRS**

   where A, P, Q, R and S are non terminals

→ This can be broken as:

$A \rightarrow PC_1$

$C_1 \rightarrow QC_2$

$C_2 \rightarrow RS$

2. **A → PQaR**

   where A, P, Q and R are non terminals and 'a' is terminal

→ Replace 'a' by C_1, we will get $A \rightarrow PQC_1R$

→ This can be broken as:

$A \rightarrow PC_2$

$C_2 \rightarrow QC_3$

$C_3 \rightarrow C_1R$

$C_1 \rightarrow a$

Example 1:

Q. Convert following context free grammar to equivalent chomsky normal form .

$S \rightarrow aSa \mid bSb \mid a \mid b \mid aa \mid bb$

Solution :

Step 1 : Identify non CNF productions

$S \rightarrow aSa \mid bSb \mid aa \mid bb$

Step 2 : If productions are starting with some terminals then replace that with some non terminal

- Hence replace a by R_1 and b by R_2 .
- Add productions $R_1 \rightarrow a$ and $R_2 \rightarrow b$ to grammar.
- Updated grammar is as follows :

$S \rightarrow R_1SR_1 \mid R_2SR_2 \mid R_1 \mid R_2 \mid R_1R_1 \mid R_2R_2$

Example 1: (Cont..)

Step 3 : Identify non CNF productions

$S \rightarrow R_1SR_1 \mid R_2SR_2$

Step 4 : In productions only two non terminals are allowed so break them as shown:

- $S \rightarrow R_1R_3$
- $R_3 \rightarrow SR_1$
- $S \rightarrow R_2R_4$
- $R_4 \rightarrow SR_2$

Step 5 : Thus the Final Grammar is in CNF having following productions :

$S \rightarrow R_1R_3 \mid R_2R_4 \mid R_1R_1 \mid R_2R_2 \mid a \mid b$

$R_3 \rightarrow SR_1$

$R_4 \rightarrow SR_2$

$R_1 \rightarrow a$

$R_2 \rightarrow b$

Example 2:

Q. Convert following context free grammar to equivalent chomsky normal form .

S → bA | aB

A → bAA | aS | a

B → aBB | bS | b

Solution :

Step 1 : Identify non CNF productions

S → bA | aB

A → bAA | aS

B → aBB | bS

Step 2 : If productions are starting with some terminals then replace that with some non terminal

- Hence replace a by R_1 and b by R_2 .
- Add productions $R_1 \rightarrow$ a and $R_2 \rightarrow$ b to grammar.

Example 2: (Cont..)

- Updated grammar is as follows :

$S \rightarrow R_2A \mid R_1B$

$A \rightarrow R_2AA \mid R_1S \mid a$

$B \rightarrow R_1BB \mid R_2S \mid b$

$R_1 \rightarrow a$

$R_2 \rightarrow b$

Step 3 : Identify non CNF productions

$A \rightarrow R_2AA$

$B \rightarrow R_1BB$

Step 4 : In productions only two non terminals are allowed so break them as shown:

- $A \rightarrow R_2R_3$
- $R_3 \rightarrow AA$
- $B \rightarrow R_1R_4$
- $R_4 \rightarrow BB$

Example 2: (Cont..)

Step 5 : Thus the Final Grammar is in CNF having following productions :

$S \rightarrow R_2A \mid R_1B$

$A \rightarrow R_2R_3 \mid R_1S \mid a$

$B \rightarrow R_1R_4 \mid R_2S \mid b$

$R_1 \rightarrow a$

$R_2 \rightarrow b$

$R_3 \rightarrow AA$

$R_4 \rightarrow BB$

Example 3:

Q. Convert following CFG to equivalent CNF.

S → ABA

A → aA | Є

B → bB | Є

Solution :

Step 1 : First we have to remove Є-productions and we get,

S → ABA

A → aA

B → bB

Step 2 : Identify nullable non terminals

- A and B are nullable non terminals
- S is also nullable non terminal as we have production :

A → Є and B → Є , S → ABA → Є (i.e. S → Є)

Example 3: (Cont..)

Step 3 : Following productions consist of nullable non terminals on R.H.S. :

$S \rightarrow ABA$

$A \rightarrow aA$

$B \rightarrow bB$

Step 4 : Add new productions by deleting all possible subsets of nullable non terminals.

1. Consider production : $S \rightarrow ABA$

$S \rightarrow BA$	(Deleting first A, since $A \rightarrow \epsilon$)
$S \rightarrow AB$	(Deleting last A, since $A \rightarrow \epsilon$)
$S \rightarrow AA$	(Deleting B, since $B \rightarrow \epsilon$)
$S \rightarrow A$	(Deleting AB, since $A \rightarrow \epsilon$ and $B \rightarrow \epsilon$)
$S \rightarrow B$	(Deleting both first and last A, since $A \rightarrow \epsilon$)
$S \rightarrow \epsilon$	(Deleting ABA, since $A \rightarrow \epsilon$ and $B \rightarrow \epsilon$)

Note : We will not add $S \rightarrow \epsilon$ to the final grammar.

Example 3: (Cont..)

Step 4 : Add new productions by deleting all possible subsets of nullable non terminals.

2. Consider production : A → aA

   A → a (Deleting A, since A → ϵ)

3. Consider production : B → bB

   B → b (Deleting B, since B → ϵ)

Step 5 :

Final Simplified or Reduced Grammar is :

S → ABA | BA | AB | AA | A | B

A → aA | a

B → bB | b

Example 3: (Cont..)

Step 6 : Removing unit productions $S \rightarrow A \mid B$

- Substitute productions of A and B in S
- After removal of unit productions we get,

$S \rightarrow ABA \mid BA \mid AB \mid AA \mid aA \mid a \mid bB \mid b$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b$

Step 7 : Identify non CNF productions

$S \rightarrow ABA \mid aA \mid bB$

$A \rightarrow aA$

$B \rightarrow bB$

Step 8 : If productions are starting with some terminals then replace that with some non terminal

- Hence replace a by R_1 and b by R_2 .
- Add productions $R_1 \rightarrow a$ and $R_2 \rightarrow b$ to grammar.

Example 3: (Cont..)

-Updated grammar is as follows :

$S \rightarrow ABA \mid BA \mid AB \mid AA \mid R_1A \mid a \mid R_2B \mid b$

$A \rightarrow R_1A \mid a$

$B \rightarrow R_2B \mid b$

$R_1 \rightarrow a$

$R_2 \rightarrow b$

Step 9 : Identify non CNF productions and we have, $S \rightarrow ABA$

Step 10 : In productions only two non terminals are allowed so break them as shown:

- $S \rightarrow AR_3$
- $R_3 \rightarrow BA$

Step 11 : Thus the Final Grammar is in CNF having following productions :

$S \rightarrow AR_3 \mid BA \mid AB \mid AA \mid R_1A \mid a \mid R_2B \mid b$

$A \rightarrow R_1A \mid a$

$B \rightarrow R_2B \mid b$

$R_1 \rightarrow a$

$R_2 \rightarrow b$

$R_3 \rightarrow BA$

D Y PATIL DEEMED TO BE UNIVERSITY RAMRAO ADIK INSTITUTE OF TECHNOLOGY NAVI MUMBAI

Example 4:

Q. Convert the following grammar to CNF form:

$S \rightarrow ABA$

$A \rightarrow aA \mid bA \mid \epsilon$

$B \rightarrow bB \mid aA \mid \epsilon$

Solution :

Step 1 : First we have to simplify the grammar. Deleting ϵ-productions we get:

$S \rightarrow ABA \mid AB \mid BA \mid AA \mid A \mid B$

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

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

Step 2 : Removing unit productions $S \rightarrow A \mid B$

- Substitute productions of A and B in S
- After removal of unit productions we get,

$S \rightarrow ABA \mid BA \mid AB \mid AA \mid aA \mid bA \mid a \mid b \mid bB$

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

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

Example 4: (Cont..)

Step 3 : Identify non CNF productions

$S \rightarrow ABA \mid aA \mid bA \mid bB$

$A \rightarrow aA \mid bA$

$B \rightarrow bB \mid aA$

Step 4 : If productions are starting with some terminals then replace that with some non terminal

- Hence replace a by R_1 and b by R_2 .
- Add productions $R_1 \rightarrow a$ and $R_2 \rightarrow b$ to grammar.
- Updated grammar is as follows :

$S \rightarrow ABA \mid AB \mid BA \mid AA \mid R_1A \mid R_2A \mid a \mid b \mid R_2B$

$A \rightarrow R_1A \mid R_2A \mid a \mid b$

$B \rightarrow R_2B \mid R_1A \mid b \mid a$

$R_1 \rightarrow a$

$R_2 \rightarrow b$

Example 4: (Cont..)

Step 5 : Identify non CNF productions

$S \rightarrow ABA$

Step 6 : In productions only two non terminals are allowed so break them as shown:

- $S \rightarrow AR_3$
- $R_3 \rightarrow BA$

Step 7 : Thus the Final Grammar is in CNF having following productions :

$S \rightarrow AR_3 \mid AB \mid BA \mid AA \mid R_1A \mid R_2A \mid a \mid b \mid R_2B$

$A \rightarrow R_1A \mid R_2A \mid a \mid b$

$B \rightarrow R_2B \mid R_1A \mid b \mid a$

$R_1 \rightarrow a$

$R_2 \rightarrow b$

$R_3 \rightarrow BA$

Example 5:

Q. Begin with the following grammar :

S → ABC | BaB

A → aA | BaC | aaa

B → bBb | a | D

C → CA | AC

D → ϵ

i) Eliminate ϵ productions

ii) Eliminate unit productions

iii) Eliminate useless symbols

iv) Convert grammar into CNF

Solution :

Step 1 : First we have to simplify the grammar. Eliminating ϵ-productions we get:

- Nullable non-terminals are B and D.

S → ABC | AC | BaB | Ba | aB | a (Putting 'B' as ϵ and generate all subsets)

A → aA | BaC | aC | aaa (Putting 'B' as ϵ and generate all subsets)

B → bBb | bb | a (Putting 'B' as ϵ and generate all subsets)

C → CA | AC

Example 5: (Cont..)

Step 2 : Eliminate unit productions

- Given grammar does not contain any unit productions so, no change.

Step 3 : Eliminate useless symbols

- Non terminal C is not generating any string of terminals so it is useless.
- Even A is useless symbol as it is not in sentential form(not reachable from start symbol).
- After removing useless symbols from grammar we get:

$S \rightarrow BaB \mid Ba \mid aB \mid a$

$B \rightarrow bBb \mid bb \mid a$

Step 4 : Identify non CNF productions

$S \rightarrow BaB \mid Ba \mid aB$

$B \rightarrow bBb \mid bb \mid a$

Example 5: (Cont..)

Step 5 : If productions are starting with some terminals then replace that with some non terminal

- Hence replace a by R_1 and b by R_2 .
- Add productions $R_1 \rightarrow a$ and $R_2 \rightarrow b$ to grammar.
- Updated grammar is as follows :

$$S \rightarrow BR_1B \mid BR_1 \mid R_1B \mid a$$
$$B \rightarrow R_2BR_2 \mid R_2R_2 \mid a$$
$$R_1 \rightarrow a$$
$$R_2 \rightarrow b$$

Step 6 : Identify non CNF productions

$$S \rightarrow BR_1B$$
$$B \rightarrow R_2BR_2$$

Example 5: (Cont..)

Step 7 : In productions only two non terminals are allowed so break them as shown:

- $S \rightarrow BR_3$
- $R_3 \rightarrow R_1B$
- $B \rightarrow R_2R_4$
- $R_4 \rightarrow BR_2$

Step 8 : Thus the Final Grammar is in CNF having following productions :

$S \rightarrow BR_3 \mid R_1B \mid BR_1 \mid a$

$B \rightarrow R_2R_4 \mid R_2R_2 \mid a$

$R_1 \rightarrow a$

$R_2 \rightarrow b$

$R_3 \rightarrow R_1B$

$R_4 \rightarrow BR_2$

Thank You

Lecture No 26:

Conversion of CFG to GNF (Part-1)

Greibach Normal Form

- A Context free grammar without ϵ productions is in GNF, if every production is in form : $A \rightarrow a\alpha$ where 'A' is non terminal, 'a' is terminal symbol and 'α' is a string of non terminals.
- Thus given grammar is in GNF if its each production contains only one leftmost terminal followed by zero or more non terminals.
- **For Example :**

$$S \rightarrow aBC \mid b$$
$$B \rightarrow aC \mid a$$
$$C \rightarrow bB \mid b$$

- The production $S \rightarrow \epsilon$ is allowed in GNF if S is a start symbol.

D Y PATIL DEEMED TO BE UNIVERSITY
RAMRAO ADIK INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

Conversion of CFG to GNF

- **Step 1 :** Check if the given CFG has any Unit Productions or Null Productions.

Remove if there are any.

- **Step 2 :** Check whether the CFG is already in CNF and convert it to CNF if it is not.
- **Step 3 :** Variables in CNF are ordered as follows :

  a) Start symbol is the lowest variable.

  b) The variable next to start symbol is higher than start symbol and so on.

- **Step 4 :** Non GNF productions of a variable should be in required form as follows :

  a) Left most symbol of right side of a non GNF production should be higher variable.

  b) For last variable the leftmost symbol should be itself.

Conversion of CFG to GNF (Cont..)

- Alter the rules so that the Non-Terminals are in ascending order, such that,
- If the production is of the form $A_i \rightarrow A_j$ then, i<j and should never be i >= j
- If the productions are not in the form then using substitutions bring them in required form.

- **For Example :**

$A_1 \rightarrow A_2A_3 \mid A_4A_4$

$A_4 \rightarrow b \mid A_1A_4$

$A_2 \rightarrow b$

$A_3 \rightarrow a$

Here, $A_4 \rightarrow A_1A_4$ is starting with lower non-terminal A_1 so substitute productions of A_1.

Updated production of A_4 is:

$A_4 \rightarrow b \mid A_1A_4$

$A_4 \rightarrow b \mid \mathbf{A_2A_3}A_4 \mid \mathbf{A_4\ A_4}\ A_4$

$A_4 \rightarrow b \mid \mathbf{b}A_3A_4 \mid A_4\ A_4\ A_4$ (Replacing A_2 by its production $A_2 \rightarrow b$)

Conversion of CFG to GNF (Cont..)

Left Recursive Production :

$$A_4 \rightarrow b \mid A_2A_3A_4 \mid A_4\ A_4\ A_4$$

Left Recursion

- **Step 5 : Removal of Left Recursive Productions**

- Let $A \rightarrow A\alpha_1 \mid A\alpha_2 \mid A\alpha_3 \mid \ldots\ldots\ldots\ldots\ldots\ldots \mid \beta_1 \mid \beta_2 \mid \beta_3 \ldots\ldots\ldots\ldots$ is a production such that the leftmost symbol for $A\alpha_1 \mid A\alpha_2 \mid A\alpha_3 \mid \ldots\ldots\ldots\ldots\ldots\ldots$ is A.

- Introduce new variable B and above productions can be replaced by following set of productions :

$$A \rightarrow \beta_i$$
$$A \rightarrow \beta_i B$$
$$B \rightarrow \alpha_i$$
$$B \rightarrow \alpha_i B$$

Conversion of CFG to GNF (Cont..)

- **Example on Removal of Left Recursive Production :**

  $A \rightarrow Aab \mid Acd \mid Aef \mid gh \mid hi$

  - Comparing this with $A \rightarrow A\alpha_1 \mid A\alpha_2 \mid A\alpha_3 \mid \ldots\ldots\ldots\ldots\ldots\ldots \mid \beta_1 \mid \beta_2 \mid \beta_3 \ldots\ldots\ldots\ldots$
  - From above production we get :

    $\alpha_1 = ab$

    $\alpha_2 = cd$

    $\alpha_3 = ef$

    $\beta_1 = gh$

    $\beta_2 = hi$

  - Let B be new variable to be introduced then above productions can be replaced by :

    $A \rightarrow gh \mid hi \mid ghB \mid hiB$

    $B \rightarrow ab \mid cd \mid ef \mid abB \mid cdB \mid efB$

Example 1:

Q. Convert following context free grammar to equivalent Greibach normal form .

S → AA | 0

A → SS | 1

Solution :

- **Step 1 : Given CFG is not having any Unit Productions or Null Productions and it is already in CNF.**
- **Step 2 : Check order of variables present in CNF.**

  - S is the start symbol.

  - S is lowest variable while ‘A’ is higher than ‘S’.

  - Hence the production A → SS is in **non GNF** form as **leftmost symbol of right side has lower order than left side variable.**

  - Using substitution convert production into required form :

A → **AA**S | **0**S | 1   (Replacing first S with S → AA | 0)

Example 1: (Cont..)

- **Step 3 : Removal of Left Recursive Production :**
  - After conversion we get left recursive production i.e. $A \rightarrow AAS \mid 0S \mid 1$
  - Comparing this with $A \rightarrow A\alpha_1 \mid A\alpha_2 \mid A\alpha_3 \mid \ldots\ldots\ldots\ldots\ldots\ldots \mid \beta_1 \mid \beta_2 \mid \beta_3 \ldots\ldots\ldots\ldots$
  - From above production we get :

$$\alpha_1 = AS$$

$$\beta_1 = 0S$$

$$\beta_2 = 1$$

  - Let B be new variable to be introduced then above productions can be replaced by :

$$A \rightarrow 0S \mid 1 \mid 0SB \mid 1B$$

$$B \rightarrow AS \mid ASB$$

- **Step 4 :** Now to bring remaining productions in GNF substitute updated production of A in S and B.

Example 1: (Cont..)

- **Step 5 : Perform substitution**

  - Production of S is S → AA | 0, replace first A by its production i.e.

    A → 0S | 1 | 0SB | 1B and we get,

    S → **0S**A | **1**A | **0SB**A | **1B**A | 0

  - Production of B is B → AS | ASB, replace first A by its production i.e.

    A → 0S | 1 | 0SB | 1B and we get,

    B → **0S**S | **1**S | **0SB**S | **1B**S | **0S**SB | **1**SB | **0SB**SB | **1B**SB

    A → 0S | 1 | 0SB | 1B

- **Step 6 : Final Grammar**

  S → 0SA | 1A | 0SBA | 1BA | 0

  B → 0SS | 1S | 0SBS | 1BS | 0SSB | 1SB | 0SBSB | 1BSB

  A → 0S | 1 | 0SB | 1B

D Y PATIL DEEMED TO BE UNIVERSITY
RAMRAO ADIK INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

Thank You

Lecture No 27:

Conversion of CFG to GNF (Part-2)

Example 2:

Q. Convert the following grammar to equivalent GNF.

$$A_1 \rightarrow A_2A_3$$
$$A_2 \rightarrow A_3A_1 \mid b$$
$$A_3 \rightarrow A_1A_2 \mid a$$

Solution :

- **Step 1 : Given CFG is not having any Unit Productions or Null Productions and it is already in CNF.**
- **Step 2 : Check order of variables present in CNF.**

  - A_1 is lowest variable while A_3 is higher variable.

  - Hence the production $A_3 \rightarrow A_1A_2$ is in **non GNF** form as **leftmost symbol of right side has lower order than left side variable.**

  - Using substitution convert production into required form :

$A_3 \rightarrow A_2A_3\,A_2 \mid a$ (Replacing A_1 by $A_1 \rightarrow A_2A_3$)

Example 2: (Cont..)

- Still it is not in required form as it is starting with lower variable A_2
- Using substitution convert production into required form :

$A_3 \rightarrow A_3A_1A_3\,A_2 \mid bA_3\,A_2 \mid a$ (Replacing A_2 by $A_2 \rightarrow A_3A_1 \mid b$)

- **Step 3 : Removal of Left Recursive Production :**
  - After conversion we get left recursive production i.e. $A_3 \rightarrow A_3A_1A_3\,A_2 \mid bA_3\,A_2 \mid a$
  - Comparing this with $A \rightarrow A\alpha_1 \mid A\alpha_2 \mid A\alpha_3 \mid \ldots\ldots\ldots\ldots \mid \beta_1 \mid \beta_2 \mid \beta_3 \ldots\ldots\ldots$
  - From above production we get :

$$\alpha_1 = A_1A_3\,A_2$$
$$\beta_1 = bA_3\,A_2$$
$$\beta_2 = a$$

  - Let B be new variable to be introduced then above productions can be replaced by :

$$A_3 \rightarrow bA_3\,A_2 \mid a \mid bA_3\,A_2\,B \mid aB$$
$$B \rightarrow A_1A_3\,A_2 \mid A_1A_3\,A_2B$$

Example 2: (Cont..)

- **Step 4 :** Now to bring remaining productions in GNF substitute updated production of A_3 in A_1 , A_2 and B.
- **Step 5 : Perform substitution**
  - Production of A_2 is $A_2 \rightarrow A_3A_1 \mid b$, replace A_3 by its production i.e.

    $A_3 \rightarrow bA_3A_2 \mid a \mid bA_3A_2B \mid aB$ and we get,

    $A_2 \rightarrow \mathbf{bA_3A_2}A_1 \mid \mathbf{a}A_1 \mid \mathbf{bA_3A_2B}A_1 \mid \mathbf{aB}A_1 \mid b$
  - Production of A_1 is $A_1 \rightarrow A_2A_3$, replace A_2 by its production i.e.

    $A_2 \rightarrow bA_3A_2A_1 \mid aA_1 \mid bA_3A_2BA_1 \mid aBA_1 \mid b$ and we get,

    $A_1 \rightarrow \mathbf{bA_3A_2A_1}A_3 \mid \mathbf{aA_1}A_3 \mid \mathbf{bA_3A_2BA_1}A_3 \mid \mathbf{aBA_1}A_3 \mid \mathbf{b}A_3$
  - Production of B is $B \rightarrow A_1A_3A_2 \mid A_1A_3A_2B$, replace A_1 by its production i.e.

    $A_1 \rightarrow bA_3A_2A_1A_3 \mid aA_1A_3 \mid bA_3A_2BA_1A_3 \mid aBA_1A_3 \mid bA_3$

    $B \rightarrow \mathbf{bA_3A_2A_1A_3}A_3A_2 \mid \mathbf{aA_1A_3}A_3A_2 \mid \mathbf{bA_3A_2BA_1A_3}A_3A_2 \mid \mathbf{aBA_1A_3}A_3A_2 \mid$
    $\mathbf{bA_3}A_3A_2 \mid \mathbf{bA_3A_2A_1A_3}A_3A_2B \mid \mathbf{aA_1A_3}A_3A_2B \mid$
    $\mathbf{bA_3A_2BA_1A_3}A_3A_2B \mid \mathbf{aBA_1A_3}A_3A_{2B} \mid \mathbf{bA_3}A_3A_2B$

Example 2: (Cont..)

- **Step 6 : Final Grammar**

$A_1 \rightarrow bA_3\,A_2A_1A_3 \mid aA_1A_3 \mid bA_3\,A_2\,BA_1A_3 \mid aBA_1A_3 \mid bA_3$

$A_2 \rightarrow bA_3\,A_2A_1 \mid aA_1 \mid bA_3\,A_2\,BA_1 \mid aBA_1 \mid b$

$A_3 \rightarrow bA_3A_2 \mid a \mid bA_3\,A_2\,B \mid aB$

$B \rightarrow bA_3\,A_2A_1A_3A_3\,A_2 \mid aA_1A_3A_3\,A_2 \mid bA_3\,A_2\,BA_1A_3A_3\,A_2 \mid aBA_1A_3A_3\,A_2 \mid$

$bA_3A_3\,A_2 \mid bA_3\,A_2A_1A_3A_3\,A_2B \mid aA_1A_3A_3\,A_2B \mid bA_3\,A_2\,BA_1A_3A_3\,A_2B \mid$

$aBA_1A_3A_3\,A_{2B} \mid bA_3A_3\,A_2B$

Example 3:

Q. Convert following grammar to equivalent GNF.

S → 0 1S | 01

S → 10S | 10

S → 00 | ε

Solution :

- **Step 1 : Given CFG is having Null Productions. So first remove ε-productions and we get,**

  S → 01S | 01 | 10S | 10 | 00

- **Step 2 : If productions are starting with some terminals then replace that with some non terminal**

  - Hence replace 0 by A and 1 by B.
  - Add productions A → 0 and B → 1 to grammar.
  - Updated grammar is as follows :

    S → ABS | AB | BAS | BA | AA

Example 3: (Cont..)

- **Step 3 : Identify non CNF productions**

  $S \rightarrow ABS \mid BAS$

- **Step 4 : In productions only two non terminals are allowed so break them as shown:**

  - $S \rightarrow AR_1$
  - $R_1 \rightarrow BS$
  - $S \rightarrow BR_2$
  - $R_2 \rightarrow AS$

- **Step 5 : Thus given Grammar is in CNF having following productions :**

  $S \rightarrow AB \mid BA \mid AA \mid AR_1 \mid BR_2$

  $A \rightarrow 0$

  $B \rightarrow 1$

  $R_1 \rightarrow BS$

  $R_2 \rightarrow AS$

Example 3: (Cont..)

- **Step 6 : Replace first A by '0' and B by '1' in every productions of S, R_1 and R_2 .**

  $S \rightarrow 0B \mid 1A \mid 0A \mid 0R_1 \mid 1R_2$

  $R_1 \rightarrow 1S$

  $R_2 \rightarrow 0S$

- **Step 7 : Final Grammar is in GNF is :**

  $S \rightarrow 0B \mid 1A \mid 0A \mid 0R_1 \mid 1R_2$

  $A \rightarrow 0$

  $B \rightarrow 1$

  $R_1 \rightarrow 1S$

  $R_2 \rightarrow 0S$

Thank You