

Theoretical Computer Science

Unit 3: Grammars

Faculty Name: Ms. Namita Pulgam

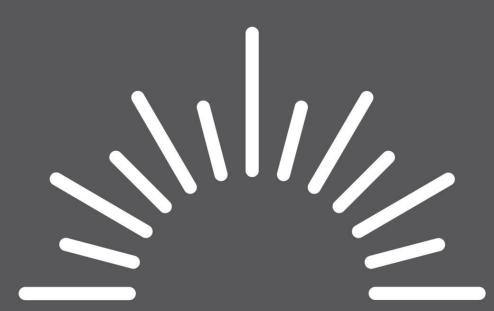
Index

Lecture 25 -Conversion of CFG to CNF	3
Lecture 26 - Conversion of CFG to GNF (Part-1)	25
Lecture 27 – Conversion of CFG to GNF(Part-2)	36



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



- 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 $X \to a$ or $X \to aY$ where $X, Y \in N$ (Non terminal) and $a \in T$ (Terminal).
- The rule $S \to \mathcal{E}$ is allowed if S does not appear on the right side of any rule.
- Example:
 - 1. $X \rightarrow \epsilon$
 - 2. $X \rightarrow a \mid aY$
 - 3. $Y \rightarrow b$



- Generates context-free languages.
- The productions must be in the form $A \to Y$ where $A \in N$ (Non terminal) and $Y \in (T \cup N)^*$ (String of terminals and non-terminals).
- Languages generated by these grammars are be recognized by a nondeterministic pushdown automaton.

• Example:

- 1. $S \rightarrow X a$
- $2. X \rightarrow a$
- $3. X \rightarrow aX$
- 4. $X \rightarrow abc$
- $5. X \rightarrow \epsilon$

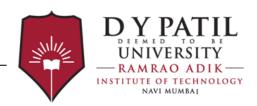


- Generates context-sensitive languages.
- The productions must be in the form

$$\alpha \land \beta \rightarrow \alpha \lor \beta$$

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

- The strings α and β may be empty, but γ must be non-empty.
- The rule $S \rightarrow \mathbf{E}$ 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 \rightarrow AbBc
 - 2. $A \rightarrow bcA$
 - $3. B \rightarrow b$



- 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 $\alpha \rightarrow \beta$

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. $S \rightarrow ACaB$
- 2. Bc \rightarrow acB
- 3. $CB \rightarrow DB$
- 4. aD \rightarrow Db



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 ε and which is generated by a grammar in which all productions are of the form $A \to BC$ or $A \to a$ 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. ε then only the following ε production is allowed in CNF.
 - $S \rightarrow E$ where S is the start symbol



Conversion of CFG to CNF

- Eliminate useless symbols, unit productions and E 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 \rightarrow 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. $A \rightarrow PQaR$

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

- \rightarrow Replace 'a' by C_1 , we will get $A \rightarrow PQC_1R$
- This can be broken as:

$$A \rightarrow PC_{2}$$

$$C_{2} \rightarrow QC_{3}$$

$$C_{3} \rightarrow C_{1}R$$

$$C_{1} \rightarrow a$$



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:

$$S \rightarrow R_1 SR_1 | R_2 SR_2 | R_1 | R_2 | R_1 R_1 | R_2 R_2$$



Step 3: Identify non CNF productions

$$S \rightarrow R_1 SR_1 \mid R_2 SR_2$$

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

- $-S \rightarrow R_1 R_3$
- $-R_3 \rightarrow SR_1$
- $-S \rightarrow R_2R_4$
- $-R_4 \rightarrow SR_2$

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

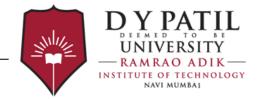
$$S \rightarrow R_1 R_3 | R_2 R_4 | R_1 R_1 | R_2 R_2 | a | b$$

$$R_3 \rightarrow SR_1$$

$$R_4 \rightarrow SR_2$$

$$R_1 \rightarrow a$$

$$R_2 \rightarrow b$$



Example 2:

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

$$S \rightarrow bA \mid aB$$

$$A \rightarrow bAA \mid aS \mid a$$

$$B \rightarrow aBB \mid bS \mid b$$

Solution:

Step 1: Identify non CNF productions

 $S \rightarrow bA \mid aB$

 $A \rightarrow bAA \mid aS$

 $B \rightarrow aBB \mid bS$

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.



- 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:

$$-A \rightarrow R_2R_3$$

$$-R_3 \rightarrow AA$$

$$-B \rightarrow R_1 R_4$$

$$-R_4 \rightarrow BB$$



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

$$S \rightarrow R_2A \mid R_1B$$

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

$$B \rightarrow R_1 R_4 \mid R_2 S \mid b$$

$$R_1 \rightarrow a$$

$$R_2 \rightarrow b$$

$$R_3 \rightarrow AA$$

$$R_4 \rightarrow BB$$



Example 3:

Q. Convert following CFG to equivalent CNF.

 $s \rightarrow ABA$

 $A \rightarrow aA \mid \epsilon$

 $B \rightarrow bB \mid \epsilon$

Solution:

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

 $S \rightarrow ABA$

 $A \rightarrow aA$

 $B \rightarrow bB$

Step 2: Identify nullable non terminals

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

 $A \rightarrow \varepsilon$ and $B \rightarrow \varepsilon$, $S \rightarrow ABA \rightarrow \varepsilon$ (i.e. $S \rightarrow \varepsilon$)



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 $A \rightarrow E$)

$$S \rightarrow \epsilon$$
 (Deleting ABA, since $A \rightarrow \epsilon$ and $B \rightarrow \epsilon$

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

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

- 2. Consider production : $A \rightarrow aA$
 - $A \rightarrow a$ (Deleting A, since $A \rightarrow \epsilon$)
- 3. Consider production : $B \rightarrow bB$

$$B \rightarrow b$$
 (Deleting B, since $B \rightarrow \epsilon$)

Step 5:

Final Simplified or Reduced Grammar is:

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

$$A \rightarrow aA \mid a$$

$$B \rightarrow bB \mid b$$



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

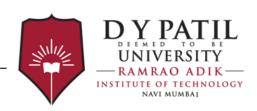
$$S \rightarrow ABA \mid aA \mid bB$$

$$A \rightarrow aA$$

$$B \rightarrow bB$$

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

- Hence replace a by \boldsymbol{R}_1 and b by \boldsymbol{R}_2 .
- Add productions $R_1 \rightarrow$ a and $R_2 \rightarrow$ b to grammar.



-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:

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

 $A \rightarrow R_1A | a$
 $B \rightarrow R_2B | b$



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 E-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,

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

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

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



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:

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

 $A \rightarrow R_1A \mid R_2A \mid a \mid b$
 $B \rightarrow R_2B \mid R_1A \mid b \mid a$
 $R_1 \rightarrow a$



 $R_2 \rightarrow b$

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:

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

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

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

$$R_1 \rightarrow a$$

$$R, \rightarrow b$$

$$R_3 \rightarrow BA$$



Example 5:

Q. Begin with the following grammar:

 $S \rightarrow ABC \mid BaB$

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

 $B \rightarrow bBb \mid a \mid D$

 $C \rightarrow CA \mid AC$

 $D \rightarrow \epsilon$

i) Eliminate **E** 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 **E**-productions we get:

- Nullable non-terminals are B and D.

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

(Putting 'B' as **E** and generate all subsets)

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

(Putting 'B' as **E** and generate all subsets)

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

(Putting 'B' as **E** and generate all sub

 $C \rightarrow CA \mid AC$

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

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

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



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:

$$S \rightarrow BR_1B \mid BR_1 \mid R_1B \mid a$$

$$B \rightarrow R_2BR_2 | R_2R_2 | a$$

$$R_1 \rightarrow a$$

$$R_2 \rightarrow b$$

Step 6: Identify non CNF productions

$$S \rightarrow BR_1B$$

$$B \rightarrow R_2BR_2$$



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:

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

$$B \rightarrow R_2R_4 | R_2R_2 | a$$

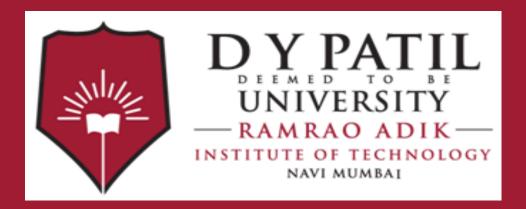
$$R_1 \rightarrow a$$

$$R_2 \rightarrow b$$

$$R_3 \rightarrow R_1 B$$

$$R_4 \rightarrow BR_2$$





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: ε and ε where 'A' is non terminal, 'a' is terminal symbol and ' ε ' 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 $S \rightarrow E$ is allowed in GNF if S is a start symbol.



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:

$$A_{1} \rightarrow A_{2}A_{3} \mid A_{4}A_{4}$$

$$A_{4} \rightarrow b \mid A_{1}A_{4}$$

$$A_{2} \rightarrow b$$

$$A_{3} \rightarrow a$$

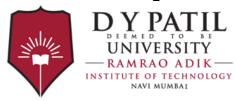
Here, $A_4 \rightarrow A_1 A_4$ is starting with lower non-terminal A_1 so substitute productions of A_1 . Updated production of A_4 is:

$$A_{4} \rightarrow b \mid A_{1}A_{4}$$

$$A_{4} \rightarrow b \mid \mathbf{A_{2}A_{3}}A_{4} \mid \mathbf{A_{4}} \mathbf{A_{4}} A_{4}$$

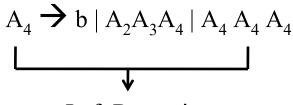
$$A_{4} \rightarrow b \mid \mathbf{b}A_{3}A_{4} \mid A_{4} A_{4} A_{4}$$

(Replacing A_2 by its production $A_2 \rightarrow b$)



Conversion of CFG to GNF (Cont..)

Left Recursive Production:



Left Recursion

- Step 5: Removal of Left Recursive Productions
- Let $A \to A\alpha_1 \mid A\alpha_2 \mid A\alpha_3 \mid \dots \mid \beta_1 \mid \beta_2 \mid \beta_3 \dots$ is a production such that the leftmost symbol for $A\alpha_1 \mid A\alpha_2 \mid A\alpha_3 \mid \dots$ 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 → Aab | Acd | Aef | gh | hi

- From above production we get:

$$\alpha_1 = ab$$

$$\alpha_2 = cd$$

$$\alpha_3 = ef$$

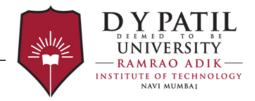
$$\beta_1 = gh$$

$$\beta_2 = hi$$

- 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.

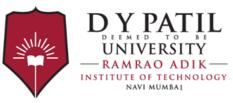
$$S \rightarrow AA \mid 0$$

$$A \rightarrow SS \mid 1$$

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 $A \rightarrow SS$ 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 :

 $A \rightarrow AAS \mid 0S \mid 1$ (Replacing first S with $S \rightarrow AA \mid 0$)



Example 1: (Cont..)

• Step 3: Removal of Left Recursive Production:

- After conversion we get left recursive production i.e. A \rightarrow AAS | 0S | 1
- Comparing this with A \rightarrow A α_1 | A α_2 | A α_3 | | β_1 | β_2 | β_3
- 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 $S \rightarrow AA \mid 0$, replace first A by its production i.e.

$$A \rightarrow 0S \mid 1 \mid 0SB \mid 1B \text{ and we get,}$$

 $S \rightarrow 0SA \mid 1A \mid 0SBA \mid 1BA \mid 0$

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

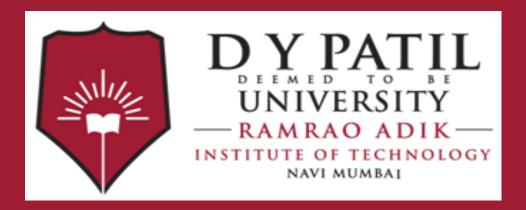
A
$$\rightarrow$$
 0S | 1 | 0SB | 1B and we get,
B \rightarrow 0SS | 1S | 0SBS | 1BS | 0SSB | 1SB | 0SBSB | 1BSB
A \rightarrow 0S | 1 | 0SB | 1B

• Step 6: Final Grammar

$$S \rightarrow 0SA \mid 1A \mid 0SBA \mid 1BA \mid 0$$

 $B \rightarrow 0SS \mid 1S \mid 0SBS \mid 1BS \mid 0SSB \mid 1SB \mid 0SBSB \mid 1BSB$
 $A \rightarrow 0S \mid 1 \mid 0SB \mid 1B$





Thank You

Lecture No 27:

Conversion of CFG to GNF (Part-2)



Example 2:

Q. Convert the following grammar to equivalent GNF.

$$A_1 \rightarrow A_2 A_3$$

$$A_2 \rightarrow A_3 A_1 \mid b$$

$$A_3 \rightarrow A_1 A_2 \mid a$$

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 $A_3 \rightarrow A_1 A_2$ 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 :

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



Example 2: (Cont..)

- Still it is not in required form as it is starting with lower variable A₂
- Using substitution convert production into required form :

$$A_3 \rightarrow A_3 A_1 A_3 A_2 \mid bA_3 A_2 \mid a$$
 (Replacing A_2 by $A_2 \rightarrow A_3 A_1 \mid b$)

- Step 3: Removal of Left Recursive Production:
 - After conversion we get left recursive production i.e. $A_3 \rightarrow A_3 A_1 A_3 A_2 \mid bA_3 A_2 \mid a$
 - Comparing this with A \rightarrow A α_1 | A α_2 | A α_3 | | β_1 | β_2 | β_3
 - From above production we get:

$$\alpha_1 = A_1 A_3 A_2$$
$$\beta_1 = bA_3 A_2$$
$$\beta_2 = a$$

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

$$A_3 \rightarrow bA_3 A_2 | a | bA_3 A_2 B | aB$$

$$B \rightarrow A_1 A_3 A_2 | A_1 A_3 A_2 B$$



Example 2: (Cont..)

- Step 4: Now to bring remaining productions in GNF substitute updated production of A₃ in A₁, A₂ and B.
- Step 5: Perform substitution
 - Production of A_2 is $A_2 \rightarrow A_3 A_1 \mid b$, 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,

$$A_2 \rightarrow bA_3A_2A_1 \mid aA_1 \mid bA_3A_2BA_1 \mid aBA_1 \mid b$$

- Production of A_1 is $A_1 \rightarrow A_2A_3$, replace A_2 by its production i.e.

$$A_2 \rightarrow bA_3 A_2 A_1 \mid aA_1 \mid bA_3 A_2 BA_1 \mid aBA_1 \mid b$$
 and we get,

$$A_1 \rightarrow bA_3 A_2 A_1 A_3 | aA_1 A_3 | bA_3 A_2 BA_1 A_3 | aBA_1 A_3 | bA_3$$

- Production of B is B \rightarrow A₁A₃A₂ | A₁A₃A₂B, replace A₁ by its production i.e.

$$A_1 \rightarrow bA_3 A_2 A_1 A_3 | aA_1 A_3 | bA_3 A_2 BA_1 A_3 | aBA_1 A_3 | bA_3$$

$$B \rightarrow bA_{3}A_{2}A_{1}A_{3}A_{3}A_{2} | aA_{1}A_{3}A_{3}A_{2} | bA_{3}A_{2} | bA_{3}A_{2}BA_{1}A_{3}A_{3}A_{2} | aBA_{1}A_{3}A_{3}A_{2} |$$

$$\mathbf{b}\mathbf{A_3}\mathbf{A_3}\mathbf{A_2} \mid \mathbf{b}\mathbf{A_3}\mathbf{A_2}\mathbf{A_1}\mathbf{A_3}\mathbf{A_3}\mathbf{A_2}\mathbf{B} \mid \mathbf{a}\mathbf{A_1}\mathbf{A_3}\mathbf{A_3}\mathbf{A_2}\mathbf{B} \mid$$

Example 2: (Cont..)

• Step 6: Final Grammar

$$A_{1} \rightarrow bA_{3}A_{2}A_{1}A_{3} | aA_{1}A_{3} | bA_{3}A_{2}BA_{1}A_{3} | aBA_{1}A_{3} | bA_{3}$$

$$A_{2} \rightarrow bA_{3}A_{2}A_{1} | aA_{1} | bA_{3}A_{2}BA_{1} | aBA_{1} | b$$

$$A_{3} \rightarrow bA_{3}A_{2} | a | bA_{3}A_{2}B | aB$$

$$B \rightarrow bA_{3}A_{2}A_{1}A_{3}A_{3}A_{2} | aA_{1}A_{3}A_{3}A_{2} | bA_{3}A_{2}BA_{1}A_{3}A_{3}A_{2} | aBA_{1}A_{3}A_{3}A_{2} |$$

$$bA_{3}A_{3}A_{2} | bA_{3}A_{2}A_{1}A_{3}A_{3}A_{2} | aA_{1}A_{3}A_{3}A_{2}B | aA_{1}A_{3}A_{3}A_{2}B | bA_{3}A_{2}BA_{1}A_{3}A_{3}A_{2}B |$$

$$aBA_{1}A_{3}A_{3}A_{2} | bA_{3}A_{2}A_{1}A_{3}A_{3}A_{2}B$$



Example 3:

Q. Convert following grammar to equivalent GNF.

 $S \rightarrow 0.1S \mid 0.1$

 $S \rightarrow 10S \mid 10$

 $S \rightarrow 00 \mid \varepsilon$

Solution:

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

 $S \rightarrow 01S \mid 01 \mid 10S \mid 10 \mid 00$

- 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 \rightarrow 0 and B \rightarrow 1 to grammar.
 - Updated grammar is as follows:

 $S \rightarrow ABS \mid AB \mid BAS \mid BA \mid AA$



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, R₁ and R₂.

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

$$R_1 \rightarrow 1S$$

$$R_2 \rightarrow 0S$$

• Step 7: Final Grammar is in GNF is:

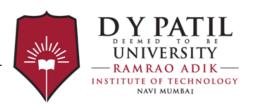
$$S \rightarrow 0B \mid 1A \mid 0A \mid 0R_1 \mid 1R_2$$

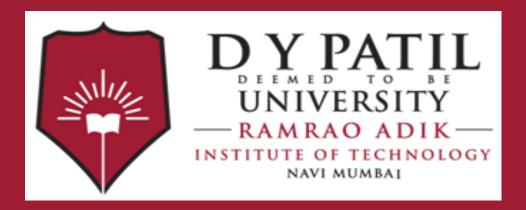
$$A \rightarrow 0$$

$$B \rightarrow 1$$

$$R_1 \rightarrow 1S$$

$$R_2 \rightarrow 0S$$





Thank You