Normal forms for Context free grammars :→

In a CFG, The R.H.S of a Production can be any string of variables and terminals. When the productions in G satisfy certain restrictions, then G is said to be in a 'normal form'. these are

1. Chomsky Normal form (CNF)
2. Greibach Normal form

Chomsky Normal form :→

in CNF, we have restrictions on the length of R.H.S and the nature of symbols in the RHS of productions.

Definition :- A CFG is in CNF if every production is of the form $A \to a$, or $A \to BC$, and $S \to \Lambda$ is in G if $\Lambda \in L(G)$. When Λ is in $L(G)$, we assume that S does not appear on the RHS of any Production.

for Example :- Consider G whose productions are

$S \to AB / \Lambda$, $A \to a$, $B \to b$. Then G is in CNF

Note :- for a grammar in CNF, the derivation tree has the following property: Every node has almost Two descendants - either two internal vertices or a single leaf.

Reduction to Chomsky Normal form :→

Let us first Consider an example. Let G be $S \to ABC/aC$, $A \to a$, $B \to b$, $C \to c$.

Except $S \to aC/ABC$, All the other Productions are in the form required for CNF.

The terminal a in $S \to aC$ can be replaced by a new variable D. By adding a new Production $D \to a$, The effect of applying $S \to aC$ can be achieved by $S \to DC$ and $D \to a$. $S \to ABC$ is not in the required form, Hence This Production can be replaced by $S \to AE$ and $E \to BC$. Thus an equivalent grammar is

$S \to AE/DC$
$E \to BC$
$A \to a$
$B \to b$
$C \to c$
$D \to a$

Reduction to Chomsky Normal form :→

For Every CFG, there is an equivalent grammar G_2 in CNF.

* Construction of a grammar in CNF.

Step 1 :– Elimination of null Productions and unit Productions

Step 2 :– Elimination of terminals on RHS : we define $G_1 = (V_N', \Sigma, P_1, S')$, where P_1 and V_N' are constructed as follows :

(i) All the productions in P of The form $A \to a$ or $A \to BC$ are included in P_1, All The variables in V_N are included in V_N'.

(ii) Consider $A \rightarrow X_1 X_2 \cdots X_n$ with some terminal on RHS. If X_i is a terminal, say a_i, add a new variable C_{a_i} to V_N' and $C_{a_i} \rightarrow a_i$ to P_1.

In production $A \rightarrow X_1 X_2 \cdots X_n$, every terminal on RHS is replaced by the corresponding new variable and the variables on the RHS are retained. The resulting production is added to P_1. Thus we get $G_1 = (V_N', \Sigma, P_1, S)$

Step 3:- Restricting the number of variables on RHS

for any production in P_1, the RHS consists of either a single terminal (or $\wedge$ in $S \rightarrow \wedge$) or two or more variables. we define $G_2 = (V_N'', \Sigma, P_2, S)$ as follows:

(i) All productions in P_1 are added to P_2 if they are in the required form. all the variables in V_N' are added to V_N''.

(ii) Consider $A \rightarrow A_1 A_2 \cdots A_m$, where $m \geq 3$. we introduce new productions $A \rightarrow A_1 C_1$, $C_1 \rightarrow A_2 C_2$, $\cdots$; $C_{m-2} \rightarrow A_{m-1} A_m$ and new variables $C_1, C_2 \cdots C_{m-2}$. These are added to P'' and V_N'', respectively.

Thus, we get G_2 in CNF.

Ex:- Find a grammar in CNF equivalent to

$$S \to aAbB$$
$$A \to aA \mid a$$
$$B \to bB \mid b$$

Soln:- As there are no unit productions or null productions, we need not carry out step 1. we proceed to step 2.

Step 2:- Let $G_1 = (V_N', \{a, b\}, P_1, S\}$, where P_1 and V_N' are constructed as follows

(i) $A \to a$, $B \to b$ are added to P_1.

(ii) $S \to aAbB$, $A \to aA$, $B \to bB$ yields $S \to C_a A C_b B$, $A \to C_a A$, $B \to C_b B$, $C_a \to a$, $C_b \to b$.

$$V_N' = \{S, A, B, C_a, C_b\}$$

Step 3 P_1 consists of $S \to C_a A C_b B$, $A \to C_a A$, $B \to C_b B$, $C_a \to a$, $C_b \to b$, $A \to a$, $B \to b$

$S \to C_a A C_b B$ is replaced by $S \to C_a C_1$, $C_1 \to A C_2$, $C_2 \to C_b B$

The remaining productions in P_1 are added to P_2. Let

$$G_2 = (\{S, A, B, C_a, C_b, C_1, C_2\}, \{a, b\}, P_2, S),$$

where P_2 consists of

$S \to C_a C_1$ $C_a \to a$

$C_1 \to A C_2$ $C_b \to b$

$C_2 \to C_b B$ $A \to a$

$A \to C_a A$ and $B \to b$

$B \to C_b B$

G_2 is in CNF and equivalent to the given grammar

Example:–

Reduce the following grammar G to CNF.

G is
$S \rightarrow aAD$
$A \rightarrow aB \mid bAB$
$B \rightarrow b$
$D \rightarrow d$

Soln:– As there are no null productions or unit productions, we can proceed to step 2.

Step 2:– Let $G_1 = (V_N', \{a, b, d\}, P_1, S)$, where P_1 and V_N' are constructed as follows

(i) $B \rightarrow b$, $D \rightarrow d$ are included in P_1

(ii) $S \rightarrow aAD$ gives rise to $S \rightarrow C_aAD$ and $C_a \rightarrow a$.

$A \rightarrow aB$ gives rise to $A \rightarrow C_aB$

$A \rightarrow bAB$ gives rise to $A \rightarrow C_bAB$ and $C_b \rightarrow b$

$V_N' = \{S, A, B, D, C_a, C_b\}$

Step 3:– P_1 consists of $S \rightarrow C_aAD$, $A \rightarrow C_aB \mid C_bAB$, $B \rightarrow b$, $D \rightarrow d$, $C_a \rightarrow a$, $C_b \rightarrow b$.

$A \rightarrow C_aB$, $B \rightarrow b$, $D \rightarrow d$, $C_a \rightarrow a$, $C_b \rightarrow b$ are added to P_2

$S \rightarrow C_aAD$ is replaced by $S \rightarrow C_aC_1$ and $C_1 \rightarrow AD$.

$A \rightarrow C_bAB$ is replaced by $A \rightarrow C_bC_2$ and $C_2 \rightarrow AB$.

Let $G_2 = (\{S, A, B, D, C_a, C_b, C_1, C_2\}, \{a, b, d\}, P_2, S)$

where P_2 consists of $S \rightarrow C_aC_1$, $A \rightarrow C_aB \mid C_bC_2$, $C_1 \rightarrow AD$, $C_2 \rightarrow AB$, $B \rightarrow b$, $D \rightarrow d$, $C_a \rightarrow a$, $C_b \rightarrow b$. G_2 is in CNF and equivalent to G.

Greebach Normal form :→

GNF is another Normal form quite useful in some Proof and constructions. A CFG generating the set accepted by PDA is in GNF.

Definition :-

A Context free grammar is in GNF if every Production is of the form $A \to a\alpha$ where $\alpha \in V_N^*$ and $a \in \Sigma$ (α may be Λ) and $S \to \Lambda$ is in G if $\Lambda \in L(G)$. When $\Lambda \in L(G)$, we assume that S does not appear on the RHS of any Production.

for Example, G given by $S \to aAB/\Lambda$, $A \to bC$, $B \to b$, $C \to c$ is in GNF.

~~Ex:~~ **Lemma 1 :-** Let $G = (V_N, \Sigma, P, S)$ be a CFG. Let $A \to B\gamma$ be an A-Production in P. Let the B-Productions be $B \to \beta_1 | \beta_2 | \cdots | \beta_s$. Define

$$P_1 = (P - \{A \to B\gamma\}) \cup \{A \to \beta_i \gamma \mid 1 \le i \le s\}$$

then $G_1 = (V_N, \Sigma, P, S)$ is a Context-free grammar equivalent to G.

Lemma 2 :- Let $G = (V_N, \Sigma, P, S)$ be a Context-free-grammar. Let the set of A-Productions be $A \to A\alpha_1 | \cdots A\alpha_r | \beta_1 \cdots | \beta_s$ (β_i's do not start with A).

Let Z be a new variable. Let $G_1 = (V_N \cup \{Z\}, \Sigma, P_1, S)$, where P_1 is defined as follows:

(i) The set of A-Productions in P_1 are $A \to \beta_1 | \beta_2 | \cdots | \beta_s$.

$$A \to \beta_1 Z | \beta_2 Z | \cdots | \beta_s Z$$

(ii) The set of Z-productions in P_1 are ..

$$\cancel{A \rightarrow \beta_1 | \beta_2 | \cdots | \beta_s}$$

$$Z \rightarrow \alpha_1 | \alpha_2 | \cdots | \alpha_r$$

$$Z \rightarrow \alpha_1 Z | \alpha_2 Z \cdots | \alpha_r Z$$

(iii) The production for the other variables are as in P. Then G_1 is a CFG and equivalent to G.

Ex:- Apply lemma 2 to the following A-productions in a CFG G.

$$A \rightarrow aBD | bDB | c$$
$$A \rightarrow AB | AD$$

Solⁿ:→ In this example, $\alpha_1 = B$, $\alpha_2 = D$, $\beta_1 = aBD$, $\beta_2 = bDB$, $\beta_3 = c$. So the new productions are:

(i) $A \rightarrow aBD | bDB | c$

(ii) $A \rightarrow aBDZ | bDBZ | cZ$

(iii) $Z \rightarrow B$, $Z \rightarrow D$

(iv) $Z \rightarrow BZ | DZ$

Ex:— Construct a grammar in GNF equivalent to the grammar $S \to AA \mid a$, $A \to SS \mid b$

Soln:→ The given grammar is in CNF.

S and A are renamed as A_1 and A_2, respectively. So the productions are $A_1 \to A_2A_2 \mid a$ and
$A_2 \to A_1A_1 \mid b$, As the grammar has no null productions and is in CNF we need not carry out step 1. So we proceed to step 2

Step 2:— (i) A_1-production are in the required form. They are $A_1 \to A_2A_2 \mid a$.

(ii) $A_2 \to b$ is in the required form. Apply lemma 1 to $A_2 \to A_1A_1$.

The resulting productions are $A_2 \to A_2A_2A_1$, $A_2 \to aA_1$. Thus the A_2-productions are

$A_2 \to A_2A_2A_1$ $A_2 \to aA_1$, $A_2 \to b$

Step 3:— we have to apply lemma 2 to A_2-productions as we have $A_2 \to A_2A_2A_1$. Let Z_2 be the new variable. The resulting productions are

$A_2 \to aA_1$, $A_2 \to b$

$A_2 \to aA_1Z_2$, $A_2 \to bZ_2$

$Z_2 \to A_2A_1$, $Z_2 \to A_2A_1Z_2$

Step 4:— (i) The A_2-productions are $A_2 \to aA_1 \mid b \mid aA_1Z_2 \mid bZ_2$

(ii) Among the A_1-productions we retain $A_1 \to a$ and eliminate $A_1 \to A_2A_2$ using lemma 1. The resulting productions are
$A_1 \to aA_1A_2 \mid bA_2$, $A_1 \to aA_1Z_2A_2 \mid bZ_2A_2$.

The set of all (modified) A_1 Productions is

$$A_1 \rightarrow a \mid aA_1A_2 \mid bA_2 \mid aA_1Z_2A_2 \mid bZ_2A_2$$

Step 5:— the Z_2-Productions to be modified are

$Z_2 \rightarrow A_2A_1$, $Z_2 \rightarrow A_2A_1Z_2$,

We apply lemma 1 and get

$$Z_2 \rightarrow aA_1A_1 \mid bA_1 \mid aA_1Z_2A_1 \mid bZ_2A_1$$

$$Z_2 \rightarrow aA_1A_1Z_2 \mid bA_1Z_2 \mid aA_1Z_2A_1Z_2 \mid bZ_2A_1Z_2$$

Hence the equivalent grammar is

$$G' = (\{A_1, A_2, Z_2\}, \{a, b\}, P_1, A_1)$$

where P_1 consists of

$$A_1 \rightarrow a \mid aA_1A_2 \mid bA_2 \mid aA_1Z_2A_1 \mid bZ_2A_2$$

$$A_2 \rightarrow aA_1 \mid b \mid aA_1Z_2 \mid bZ_2$$

$$Z_2 \rightarrow aA_1A_1 \mid bA_1 \mid aA_1Z_2A_1 \mid bZ_2A_1$$

$$Z_2 \rightarrow aA_1A_1Z_2 \mid bA_1Z_2 \mid aA_1Z_2A_1Z_2 \mid bZ_2A_1Z_2$$

Example:— Convert the grammar $S \rightarrow AB$, $A \rightarrow BS \mid b$, $B \rightarrow SA \mid a$ into GNF.

Soln:— As the given grammar is in CNF, we can omit step 1 and proceed to step 2 after renaming S, A, B as A_1, A_2, A_3, respectively. The productions are

$$A_1 \rightarrow A_2A_3$$

$$A_2 \rightarrow A_3A_1 \mid b$$

$$A_3 \rightarrow A_1A_2 \mid a$$

step 2:-

(i) The A_1-Production $A_1 \to A_2A_3$ is in required form

(ii) The A_2-Productions $A_2 \to A_3A_1 | b$ are in the required form

(iii) $A_3 \to a$ is in the required form

Apply lemma 1 to $A_3 \to A_1A_2$. The resulting production are

$A_3 \to A_2A_3A_2$ applying the lemma once again to $A_3 \to A_2A_3A_2$.

we get $A_3 \to A_3A_1A_3A_2 | bA_3A_2$

Step 3:- The A_3-Productions are $A_3 \to a / bA_3A_2$ and

$A_3 \to A_3A_1A_3A_2$. As we have $A_3 \to A_3A_1A_3A_2$. we have to apply lemma 2 to A_3-Productions let Z_3 be the new variable. The resulting productions are

$A_3 \to a | bA_3A_2$

$A_3 \to aZ_3 | bA_3A_2Z_3$

$Z_3 \to A_1A_3A_2$

$Z_3 \to A_1A_3A_2Z_3$

step 4:- (i) The A_3-Productions are

$A_3 \to a | bA_3A_2 | aZ_3 | bA_3A_2Z_3$

(ii) Among the A_2-Productions, we retain $A_2 \to b$ and eliminate $A_2 \to A_3A_1$. using lemma 1. The resulting Productions are

$A_2 \to aA_1 | bA_3A_2A_1 | aZ_3A_1 | bA_3A_2Z_3A_1$

The modified A_2-Productions are

$A_2 \to b | aA_1 | bA_3A_2A_1 | aZ_3A_1 | bA_3A_2Z_3A_1$

(iii) We apply lemma 1 to $A_1 \to A_2A_3$ to get

$$A_1 \to bA_3 \mid aA_1A_3 \mid bA_3A_2A_1A_3 \mid aZ_3A_1A_3 \mid bA_3A_2Z_3A_1A_3$$

Step 5:- The Z_3 - Productions to be modified are

$$Z_3 \to A_1A_3A_2 \mid A_1A_3A_2Z_3$$

We apply lemma 1 and get

$$Z_3 \to bA_3A_3A_2 \mid bA_3A_2Z_3$$

$$Z_3 \to aA_1A_3A_3A_2 \mid aA_1A_3A_3A_2Z_3$$

$$Z_3 \to bA_3A_2A_1A_3A_3A_2 \mid bA_3A_2A_1A_3A_3A_2Z_3$$

$$Z_3 \to aZ_3A_1A_3A_3A_2 \mid aZ_3A_1A_3A_3A_2Z_3$$

$$Z_3 \to bA_3A_2Z_3A_1A_3A_3A_2 \mid bA_3A_2Z_3A_1A_3A_3A_2Z_3$$

The required grammar is in GNF.

Pumping lemma for Context-free languages: →

The Pumping lemma for CFG gives a method of generating an infinite number of strings from a given sufficiently long string in a context free language L. It is used to prove that certain languages are not context free.

The construction we make use of in proving pumping lemma yield some decision algorithms regarding CFG.

Theorem :- Pumping lemma for context free ~~grammar~~ languages — Let L be a context free language. Then we can find a natural number n such that:

(i) Every $z \in L$ with $|z| \geq n$ can be written as $uvwxy$ for some strings u, v, w, x, y.

(ii) $|vx| \geq 1$

(iii) $|vwx| \leq n$.

(iv) $uv^k w x^k y \in L$ for all $k \geq 0$