LANGUAGE GENERATED BY A GRAMMAR

The set of all strings that can be derived from a grammar is said to be the language generated from that grammar. A language generated by a grammar **G** is a subset formally defined by

$$L(G)=\{W|W \in \sum^*, S \Rightarrow_G \mathbf{W}\}$$

If $\mathbf{L}(G1) = \mathbf{L}(G2)$, the Grammar **G1** is equivalent to the Grammar **G2**.

Example

If there is a grammar

G: N = {S, A, B} T = {a, b} P = {S → AB, A → a, B → b}

Here **S** produces **AB**, and we can replace **A** by **a**, and **B** by **b**. Here, the only accepted string is **ab**, i.e.,

$L(G) = \{ab\}$

Example

Suppose we have the following grammar –

G: N = {S, A, B} T = {a, b} P = {S → AB, A → aA|a, B → bB|b}

The language generated by this grammar –

$L(G) = \{ab, a^2b, ab^2, a^2b^2, \ldots\ldots\ldots\}$

$= \{a^m b^n \mid m \geq 1 \text{ and } n \geq 1\}$

Construction of a Grammar Generating a Language

We'll consider some languages and convert it into a grammar G which produces those languages.

Example

Problem – Suppose, L G = {a^m b^n | m ≥ 0 and n > 0}. We have to find out the grammar **G** which produces **L**G.

Solution

Since LG = {a^m b^n | m ≥ 0 and n > 0}

the set of strings accepted can be rewritten as –

LG = {b, ab,bb, aab, abb, …….}

Here, the start symbol has to take at least one 'b' preceded by any number of 'a' including null.

To accept the string set {b, ab, bb, aab, abb, …….}, we have taken the productions –

S → aS , S → B, B → b and B → bB

S → B → b $Accepted$

S → B → bB → bb $Accepted$

S → aS → aB → ab $Accepted$

S → aS → aaS → aaB → aab$Accepted$

S → aS → aB → abB → abb $Accepted$

Thus, we can prove every single string in LG is accepted by the language generated by the production set.

Hence the grammar –

G: $S, A, B, a, b, S, S \to aS|B, B \to b|bB$

Example

Problem – Suppose, L G = {a^m b^n | m > 0 and n ≥ 0}. We have to find out the grammar G which produces LG.

Solution –

Since LG = {a^m b^n | m > 0 and n ≥ 0}, the set of strings accepted can be rewritten as –

LG = {a, aa, ab, aaa, aab ,abb, …….}

Here, the start symbol has to take at least one 'a' followed by any number of 'b' including null.

To accept the string set {a, aa, ab, aaa, aab, abb, …….}, we have taken the productions –

S → aA, A → aA , A → B, B → bB ,B → λ

S → aA → aB → aλ → a $Accepted$

S → aA → aaA → aaB → aaλ → aa $Accepted$

S → aA → aB → abB → abλ → ab $Accepted$

S → aA → aaA → aaaA → aaaB → aaaλ → aaa $Accepted$

S → aA → aaA → aaB → aabB → aabλ → aab $Accepted$

S → aA → aB → abB → abbB → abbλ → abb $Accepted$

Thus, we can prove every single string in LG is accepted by the language generated by the production set.

Hence the grammar –

G: $S, A, B, a, b, S, S \rightarrow aA, A \rightarrow aA|B, B \rightarrow \lambda|bB$