$$B \rightarrow aB/a \qquad (P_2, \text{ say})$$

$$aaaB \rightarrow ba \qquad (P_3, \text{ say})$$

Solution. The first production P_1 says that we can start with A and derive number of symbols of the type B, for example

$$A \Rightarrow BA$$
$$\Rightarrow BBA$$
$$\Rightarrow BBBA$$
$$\Rightarrow BBBBA$$
$$\Rightarrow BBBB$$

The second production shows us that each B can be any string of a's (with at least one a) :

$$B \Rightarrow aB$$
$$\Rightarrow aaB$$
$$\Rightarrow aaaB$$
$$\Rightarrow aaaaB$$
$$\Rightarrow aaaaa$$

The third production (P_3) says that any time we find three a's and B, we can replace these four symbols with the two-terminal string ba.

The following is a summary of one possible derivation in this grammar :

$$A \overset{*}{\Rightarrow} BBBBBB$$
$$\overset{*}{\Rightarrow} aaaaBBBBB$$
$$\Rightarrow aabaBBBB$$
$$\Rightarrow aabaaaBB$$
$$\Rightarrow aabbaBB$$
$$\Rightarrow aabbaaaB$$
$$\Rightarrow aabbba$$

All CFG's are phrase-structure grammars in which we restrict ourselves as to what we put on the left side of productions. So, all CFLs can be generated by phrase-structure grammars.

10.2. CHOMSKY HIERARCHY

We can exhibit the relationship between grammars by the Chomsky Hierarchy. Noum Chomsky, a founder of formal language theory, provided an initial classification in to four languages types :

Type – 0	(*Unrestricted grammar*)
Type – 1	(*Context sensitive grammar*)
Type – 2	(*Context free grammar*)
Type – 3	(*Regular grammar*)

Type 0 languages are those generated by unrestricted grammars, that is, the recursively enumerable languages. Type 1 consist of the context-sentive languages, Type 2 consists of the context-free languages and Type 3 consists of the regular languages. Each language family of type k is a proper subset of the family of type $k - 1$. Following diagram shows the original Chomsky Hierarchy.

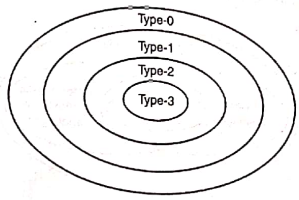

Fig. 10.1. The original Chomsky Hierarchy.

We have also met several other language families that can be fitted in to this picture. Including the families of deterministic context-free languages (L_{DCF}), and recursive languages (L_{REC}). The modified Chomsky Hierarchy can be seen in Fig. 10.2.

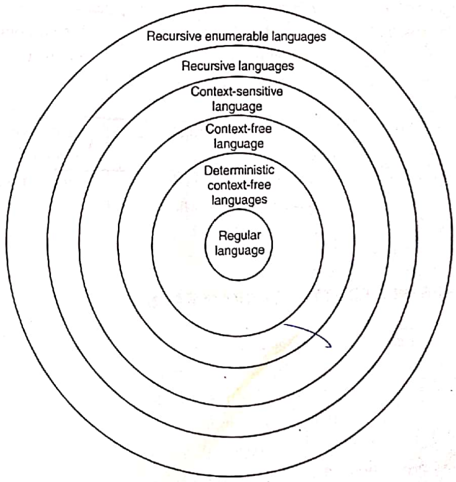

Fig. 10.2.

We know that $L = \{w : n_a(w) = n_b(w)\}$ is deterministic, but not linear. On the other hand, the language

$$L = \{a^n b^n\} \cup \{a^n b^{2n}\}$$

is linear, but not determinstic. The relationship between, linear, determinsitic context-free and non-determinstic context-free language is shown in Fig. 10.3.

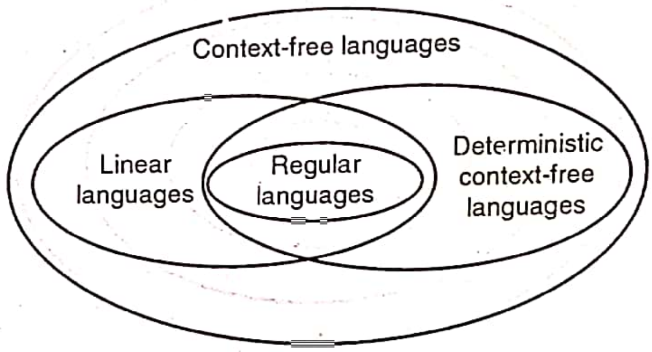

Fig. 10.3.

In the last we can summarise our discussion as follows :

Type	*Name of languages generated*	*Production Restrictions* $A \rightarrow B$	*Acceptor*
0	Unrestricted (reversively-enumerable)	A = any string with non-terminal B = any string	Turing machine
1	Context-sentive	A = any string with non-terminals B = any string as long as or longer than A	Linear bound automata
2	Context-free	A = one non-terminal B = any string	Pushdown automata
3	Regular	A = one non-terminal $B = aX$ or $B = a$, where a is a terminal and X is a non-terminal.	Finite automata