Chomsky Hierarchy of Grammars

Shashank Gupta
Assistant Professor
Department of Computer Science and Information Systems

BITS Pilani
Pilani Campus

Examples of CFG

Construct a CFG that generates all strings of a's and b's where each string starts and ends with different symbol.

- L = {ab, ba, abb, bbbaba, ------}

$S \rightarrow aAb \mid bAa$

$A \rightarrow \in \mid aA \mid bA$

Examples of CFG (Continued…..)

Construct a CFG that generates all even length palindromes over $\Sigma = \{a,b\}$

- L = {$\in$, aa, bb, abba, baab,-------}

$S \rightarrow \in \mid aSa \mid bSb$

Examples of CFG (Continued….)

Construct a CFG for the following language of balanced parenthesis

- L = {w ∈ {[,]}* | w is balanced}

S → ∈| [S] | SS

Chomsky Hierarchy of Grammars
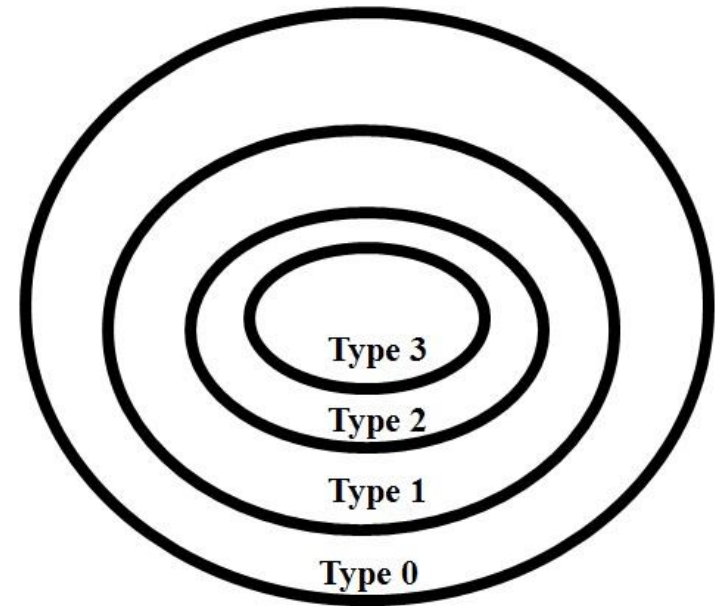
The Chomsky Hierarchy

In 1950s, *Noam Chomsky* structured a hierarchy of formal languages based on the properties of Grammars required to generate the language.

He classified the grammar into four types (Type 3, Type 2, Type 1 and Type 0) depending on the production rules.

The Chomsky Hierarchy (Continued……)

Type Name	Formal Language	Automata	Grammar
Type 3	Regular Language	Finite Automata(FA)	Regular Grammar/Restricted Grammar
Type 2	Context Free Language	Pushdown Automata(PDA)	Context-Free Grammar
Type 1	Context Sensitive Language	Linear Bounded Automata(LBA)	Context-Sensitive Grammar
Type 0	Recursive Enumerable Language	Turing Machine(TM)	Unrestricted Grammar

Type 3
Type 2
Type 1
Type 0

The Chomsky Hierarchy (Continued…..)

Recursive Enumerable Language or Type 0 Language

Context-Sensitive Language (CSL) or Type 1 Language

Context-Free Language (CFL) or Type 2 Language

Regular Language or Type 3 Language

Finite Automaton

Pushdown Automaton

Linear Bounded Automaton

Turing Machine

$$Type_{i+1} \subseteq Type_i$$
$$For\ i=0,1,2$$

The Chomsky Hierarchy of Grammars

Recursive Enumerable Language or Type 0 Language

Context-Sensitive Language (CSL) or   Type 1 Language

Context-Free Language (CFL) or Type 2 Language

$$A \rightarrow xB \mid Bx \mid x$$

$$A, B \in V$$

$$x \in \sum{}^{*}$$

Regular Grammar/ Restricted Grammar

Finite Automaton

Pushdown Automaton

Linear Bounded Automaton

Turing Machine

The Chomsky Hierarchy of Grammars (Continued…..)

Recursive Enumerable Language or Type 0 Language

Context-Sensitive Language (CSL) or   Type 1 Language

$$A \rightarrow \alpha \ where \ \alpha \in (V + T)*$$

Context-Free Grammar/ Type 2 Grammar

$$A \rightarrow xB \mid Bx \mid x$$
$$A, B \in V$$
$$x \in \sum\nolimits^{*}$$

Regular Grammar/ Restricted Grammar/ Type 3 Grammar

Finite Automaton

Pushdown Automaton

Linear Bounded Automaton

Turing Machine

The Chomsky Hierarchy of Grammars (Continued…..)

Recursive Enumerable Language or Type 0 Language

$\alpha A \beta \rightarrow \alpha \gamma \beta \ where \ A \in V, \alpha, \beta \in (V+T)^* and \ \gamma \in (V+T)^+ \ s.t. |LHS| \leq |RHS|$

Context-Sensitive Grammar/Type 1 Grammar

$A \rightarrow \alpha \ where \ \alpha \in (V+T)^*$

Context-Free Grammar (CFG) or Type 2 Grammar

$A \rightarrow xB \,|\, Bx \,|\, x$

$A, B \in V$

$x \in \sum^*$

Regular Grammar/ Unrestricted Grammar/ Type 3 Grammar

Finite Automaton

Pushdown Automaton

Linear Bounded Automaton

Turing Machine

Note: **In Type 1 Grammar, Start symbol can derive epsilon symbol unlike, other variables of Grammar.**

The Chomsky Hierarchy (Continued…..)

$$\alpha A \beta \rightarrow \gamma \ \ where \ A \in V, \alpha, \beta, \gamma \in (V + T)*$$

$$\alpha A \beta \rightarrow \alpha \gamma \beta \ \ where \ A \in V, \alpha, \beta \in (V+T)* and \ \gamma \in (V+T)^+ \ s.t. |LHS| \leq |RHS|$$

Type 0 Grammar/ Unrestricted Grammar

Context-Sensitive Grammar/ Type 1 Grammar

$$A \rightarrow \alpha \ \ where \ \alpha \in (V + T)*$$

Context-Free Grammar or Type 2 Grammar

$$A \rightarrow xB \mid Bx \mid x$$
$$A, B \in V$$
$$x \in \sum{}^{*}$$

Regular Grammar/ Restricted Grammar/ Type 3 Grammar

Finite Automaton

Pushdown Automaton

Linear Bounded Automaton

Turing Machine

Examples

Determine the highest type number satisfied by the following Grammar

S → AB	S → aSAc | abc	S → aSAc | a
A → a	cA → Bc	CAa → Bc
B → b	B → a	B → abc
Type 2 Grammar	**Type 1 Grammar**	**Type 0 Grammar**