Chapter 13: Grammatical Format

I. Theory of Automata
→ II. Theory of Formal Languages
III. Theory of Turing Machines …

Theorem. All regular languages are context-free languages.

Proof. We show that for any FA, there is a CFG such that the language generated by the grammar is the same as the language accepted by the FA.

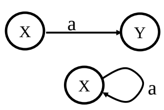
By constructive algorithm.

Input: a finite automaton.

Output: a CF grammar.

- The alphabet Σ of terminals is the alphabet of the FA.
- Nonterminals are the state names. (The start state is renamed S.)

- For every edge, create a production:



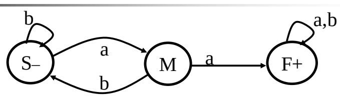
$X \to aY$

$X \to aX$

- For every final state, create a production:

$X \to \Lambda$

Example:



S → aM / bS
M → aF/ bS
F → aF/ bF/ Λ

babbaaba

$$S \xrightarrow{b} S \xrightarrow{a} M \xrightarrow{b} S \xrightarrow{b} S \xrightarrow{a} M \xrightarrow{a} F \xrightarrow{b} F \xrightarrow{a} F$$

$$S \Rightarrow bS \Rightarrow baM \Rightarrow babS \Rightarrow babbS \Rightarrow babbaM \Rightarrow babbaaF \Rightarrow babbaabF \Rightarrow babbaabaF \Rightarrow babbaaba$$

Definition: A semiword is a sequence of terminals (possibly none) followed by exactly one nonterminal.

(terminal)(terminal)…(terminal)(Nonterminal)

Definition: A CFG is a regular grammar if all of its productions have the form:

Nonterminal → semiword

or

Nonterminal → word (a sequence of terminals or Λ)

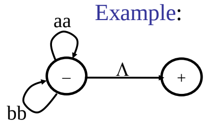
Theorem. All languages generated by regular grammars are regular.

Proof. By constructive algorithm. We build a transition graph.

- The alphabet Σ of the transition graph is the set of terminals.
- One state for each nonterminal. The state named S is the start state. We add one final state +.
- Transitions:

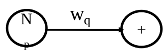
$N_x \to w_y N_z$ $N_p \to w_q$

Example: S → aaS   S → bbS   S → Λ



(aa+bb)*

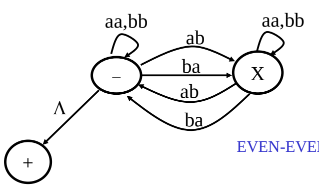
Alternative Algorithm:



1. Np→ wq whenever wq is not Λ
2. For each transition of the form N→ Λ, we mark the state for N with +..

Example

S → aaS / bbS / abX / baX / Λ
X → aaX / bbX / abS / baS



EVEN-EVEN

- If the empty word is in the language, a production of the form N → Λ (called a Λ-production) is necessary.
- The existence of a production of the form N → Λ des not necessarily mean that Λ is a part of the language.

  S → aX   S → a   X → Λ

Theorem. Let L be a language generated by a CFG. There exists a CFG without productions of the form X → Λ such that:

1. If Λ∉L, L is generated by the new grammar.
2. If Λ∈L, all words of L except for Λ are generated by the new grammar.

Definition. A production of the form:
Nonterminal → one Nonterminal is called a unit production.

Theorem. Let L be a language generated by a CFG that has no Λ-productions. Then there is another CFG without Λ-productions and without unit productions that generates L.

Theorem. Let L be a language generated by a CFG. The there exists another CFG that generates all words of L (except Λ) such that all the productions are of the form:

Nonterminal → sequence of Nonterminals
Nonterminal → one terminal

Definition. A CFG in is said to be in Chomsky Normal Form (CNF) if all the productions have the form:

Nonterminal → (Nonterminal)(Nonterminal)
Nonterminal → terminal

Theorem. Let L be a language generated by a CFG. There there is another grammar which is in CNF that generates all the words of L (except Λ).

Definition. In a sequence of terminals and nonterminals in a derivation (called a working string), if there is at least one nonterminal, then the first one is called the leftmost nonterminal.

Definition. A leftmost derivation is a derivation where at each step, a production is applied to the leftmost nonterminal in the working string.

Example. S → aSX | b
X → Xb | a

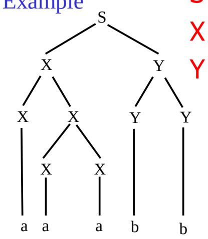
S ⇒ aSX ⇒ aaSXX ⇒ aabXX ⇒ aabXbX ⇒ aababX ⇒ aababa

Example

$S \rightarrow XY$

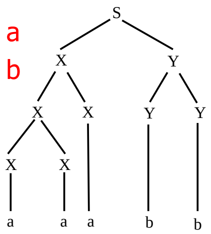
$X \rightarrow XX \mid a$

$Y \rightarrow YY \mid b$



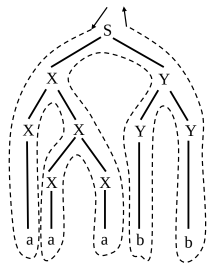
Derivation I

aaabb

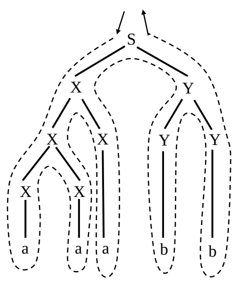


Derivation II

S $\Rightarrow$ XY
$\Rightarrow$ XXY
$\Rightarrow$ aXY
$\Rightarrow$ aXXY
$\Rightarrow$ aaXY
$\Rightarrow$ aaaY
$\Rightarrow$ aaaYY
$\Rightarrow$ aaabY
$\Rightarrow$ aaabb



S $\Rightarrow$ XY
$\Rightarrow$ XXY
$\Rightarrow$ XXXY
$\Rightarrow$ aXXY
$\Rightarrow$ aaXY
$\Rightarrow$ aaaY
$\Rightarrow$ aaaYY
$\Rightarrow$ aaabY
$\Rightarrow$ aaabb



Theorem. Any word that is in the language generated by a CFG has a leftmost derivation.

Example. $S \rightarrow (S) \mid S \supset S \mid \sim S \mid p \mid q$

$S \Rightarrow (S) \Rightarrow (S \supset S) \Rightarrow (p \supset S)$
$\Rightarrow (p \supset (S)) \Rightarrow (p \supset (S \supset S))$
$\Rightarrow (p \supset (\sim S \supset S))$
$\Rightarrow (p \supset (\sim p \supset S))$
$\Rightarrow (p \supset (\sim p \supset q))$