补充材料：

1. 文法定义简介：

CNF：Chomsky normal form

In formal language theory, a context-free grammar G is said to be in **Chomsky normal form** (first described by Noam Chomsky)[1] if all of its production rules are of the form:[2]:92–93,106

$A \to BC$, or
$A \to a$, or
$S \to \varepsilon$,

GNF：Greibach normal form; 每一个不产生空串的上下文无关语法都能转化为格雷巴赫范式

In formal language theory, a context-free grammar is in **Greibach normal form** (GNF) if the right-hand sides of all production rules start with a terminal symbol, optionally followed by some variables. A non-strict form allows one exception to this format restriction for allowing the empty word (epsilon, ε) to be a member of the described language. The normal form was established by Sheila Greibach and it bears her name.

More precisely, a context-free grammar is in Greibach normal form, if all production rules are of the form:

$$A \to aA_1A_2\cdots A_n$$

or

$$S \to \varepsilon$$

GNF 构造步骤

1.把 2 型文法写成 CNF

2.对非终结符进行编号

3.如果存在生成式 Ai -> Aj ω 其中 Ai 的编号不小于 Aj 的编号，则把之前 Aj 的生成式代入，直到左侧非终结符 Ai 的编号不大于右侧第一个非终结符的编号

4.消除左递归，对 An -> Anω 进行变换，得到 An 的无左递归生成式

5.把 An 的无左递归生成式回代如编号小的非终结符的生成式

e.g.

```
A -> BC
B -> CA | b
C -> AB | a
```

对非终结符进行编号：A - 1, B - 2, C - 3

在生成式 C -> AB 中左侧非终结符编号大于右侧第一个非终结符编号，把 A 的生成式代入得 C -> BCB | a，仍然不满足条件。把 B 的生成式代入得 C -> CACB | bCB | a

消除左递归：（见下面）

C -> bCBC' | aC' | bCB | a

C' -> ACBC' | ACB

得到了 C 的生成式，回代入以 C 开头的生成式 B -> CA 即可获得其他字母的表示

2. 消除左递归方法

E→E +T |T

由于左递归时本身出现至少一次，则创建新符号为最左推导，该符号指向原有的产生式推导即可：

E→TE'

E'→+TE'|ε