**1.**给出下列术语的严格定义：

1) 上下文无关文法　　　　2) LL(1)文法

(浙江大学 1998 年硕士生入学考试试题)

解答：

上下文无关文法的定义：

形式上说，一个上下文无关文法 G 是一个四元式（$V_T$，$V_N$，S，*P*），其中

$V_T$是一个非空有限集，它的每个元素称为终结符号；

$V_N$是一个非空有限集，它的每个元素称为非终结符号，$V_T \cap V_N = \phi$；

S 是一个非终结符号，称为开始符号；

*P* 是一个产生式集合（有限），每个产生式的形式是 $P \to \alpha$，其中，$P \in V_N$，$\alpha \in$（$V_T \cup V_N$）$^*$。开始符号 S 至少必须在某个产生式的左部出现一次。

LL（1）文法的定义：

如果一个文法 G 满足下面的条件，则称该文法 G 为 LL（1）文法：

1. 文法不含左递归，

2. 对于文法中每一个非终结符 A 的各个产生式的候选首符集两两不相交。即，若 $A \to \alpha_1 | \alpha_2 | \ldots | \alpha_n$

则　　FIRST（$\alpha_i$）∩FIRST（$\alpha_j$）$= \phi$　　　（$i \neq j$）

3. 对文法中的每个非终结符 A，若它存在某个候选首符集包含，则

FIRST（A）∩FOLLOW（A）$=\phi$

**2.**给出文法 G1(S)

$S \to aSb \mid P$

$P \to bPc \mid bQc$

$Q \to Qa \mid a$

消除文法的左递归、提取左公共因子后是不是 LL(1)文法？请证实。

(上海交大 1999 年硕士生入学考试试题)

解题思路：这类题目主要考查理解和运用消除文法的左递归、提取左公共因子等算法的能力，为判断文法是否是 LL(1)文法，还要计算文法的 FIRST 和 FOLLOW 集合。

消除文法的左递归的基本思想是，将文法规则中的的左递归结构变换成等价的右递归结构。

消除直接左递归。一般而言，假定 P 关于的全部产生式是

$$P \to P\alpha 1 \mid P\alpha 2 \mid \ldots \mid P\alpha m \mid \beta 1 \mid \beta 2 \mid \ldots \mid \beta n$$

其中，每个$\alpha$都不等于$\varepsilon$，而每个$\beta$都不以 P 开头，那么，消除 P 的直接左递归性就是把这些规则改写成：

$$P \to \beta 1P' \mid \beta 2P' \mid \ldots \mid \beta nP'$$

$$P' \to \alpha 1P' \mid \alpha 2P' \mid \ldots \mid \alpha mP' \mid \varepsilon$$

$$\overset{+}{\Rightarrow}$$

对于间接左递归的情况，如果一个文法不含回路（形如 PP 的推导），也不含以$\varepsilon$为右部的产生式，那么我们可以按照下面的算法消除文法中所有的左递归：

1. 把文法 G 的所有非终结符按任一种顺序排列成 $P_1$，$P_2$，…，$P_n$；按此顺序执行：

2. FOR i:=1 TO n DO

BEGIN

FOR j:=1 TO i-1 DO

把形如 $P_i \to P_j\gamma$的规则改写成

$P_i \to \delta_1\gamma|\delta_2\gamma|\ldots|\delta_k\gamma$。其中 $P_j \to \delta_1|\delta_2|\ldots|\delta_k$ 是关于 $P_j$ 的所有规则；

消除关于 $P_i$ 规则的直接左递归性

END

3. 化简由 2 所得的文法。即去除那些从开始符号出发永远无法到达的非终结符的产生规则。

提取左公因子的算法：

假定关于 A 的规则是

$A \to \delta\beta_1 | \delta\beta_2 | \ldots | \delta\beta_n | \gamma_1 | \gamma_2 | \ldots \gamma_m$

（其中，每个γ不以δ开头）

那么，可以把这些规则改写成

$$A \to \delta A' \mid \gamma_1 \mid \gamma_2 \mid \ldots \mid \gamma_m$$
$$A' \to \beta_1 \mid \beta_2 \mid \ldots \mid \beta_n$$

经过反复提取左因子，就能够把每个非终结符（包括新引进者）的所有候选首符集变成为两两不相交，但是我们大量引进了新的非终结符和ε-产生式。

消除文法的左递归、提取左公共因子后，再计算文法的各非终结符(X)的首符集 FIRST(X)和随符集 FOLLOW(X)，首先要理解这两个集合的计算方法，特别要注意，算法终止的条件：直到集合不再变化为止，也就是说，反复检查每一个产生式，直到从头到尾检查了所有产生式，而 FIRST 和 FOLLOW 集合都没有变化了，这时候计算才能结束。

判断一个文法是不是 LL(1)的，首先必须计算文法的各非终结符(X)的首符集 FIRST(X)和随符集 FOLLOW(X)，然后根据 LL(1)文法的充分必要条件(即 LL(1)文法的定义)来判断。即可根据是 LL(1)文法的充要条件，判断文法是否是 LL(1)的。

解答：消除文法的左递归，得到

$S \to aSb \mid P$

$P \to bPc \mid bQc$

$Q \to a\,Q'$

$Q' \to a\,Q' \mid \varepsilon$

提取左公共因子，后得到文法 G'(S)：

$S \to aSb \mid P$

$P \to bP'$

$P' \to Pc \mid Qc$

$Q \to a\,Q'$

$Q' \to a\,Q' \mid \varepsilon$

计算变换后的文法的 FIRST 和 FOLLOW 集合：

FIRST(S) = {a，b}　　FIRST(P) = {b}　　FIRST(P') = {a，b}

FIRST(Q) = {a }　　FIRST(Q') = {a，ε}

FOLLOW (S) = {b，#}　　FOLLOW (P) = { b，c，#}

FOLLOW (P') = { b，c，#}

FOLLOW (Q) = {c}　　FOLLOW (Q') = {c}

检查变换后的文法，我们可以得到：

1. 变换后的文法不含左递归，

2. 对于变换后的文法中每一个非终结符 S，P，P'，Q，Q'的各个产生式的候选首符集两两不相交。

3. 对变换后的文法中的非终结符 Q'，它的一个候选式是，而且

FIRST（Q'）∩FOLLOW（Q'）=$\phi$

所以变换后的文法是 LL(1)文法。