# LL(1) 文法的预测分析

## 一、实验目的

1. 了解 LL(1)语法分析是如何根据语法规则逐一分析词法分析所得到的单词，检查语法错误，即掌握语法分析过程。

2. 掌握 LL(1)语法分析器的设计与调试。

## 二、实验内容

文法G[E]：E→TE’，E’→+TE’|ε，T→FT’，T’→\*FT’|ε，F→(E) | i

针对上述文法，编写一个 LL(1)语法分析程序：

1. 输入：诸如 i+i\*i 的字符串，以$结束。

2. 处理：基于分析表进行 LL(1)语法分析，判断其是否符合文法。

3. 输出：串是否合法。

## 三、实验要求

1. 在编程前，根据上述文法建立对应的、正确的预测分析表。

2. 设计恰当的数据结构存储预测分析表。（ε可用#代替）

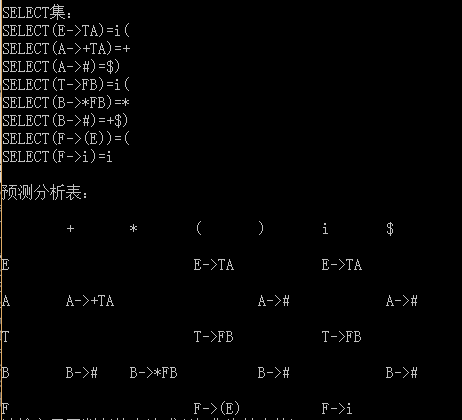
3. 任选 熟悉的编程语言，要求所编程序结构清晰。

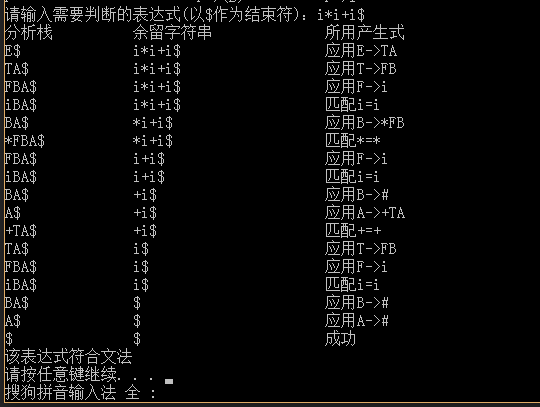
4. 完成实验报告及总结。

## 四、实验设计方案

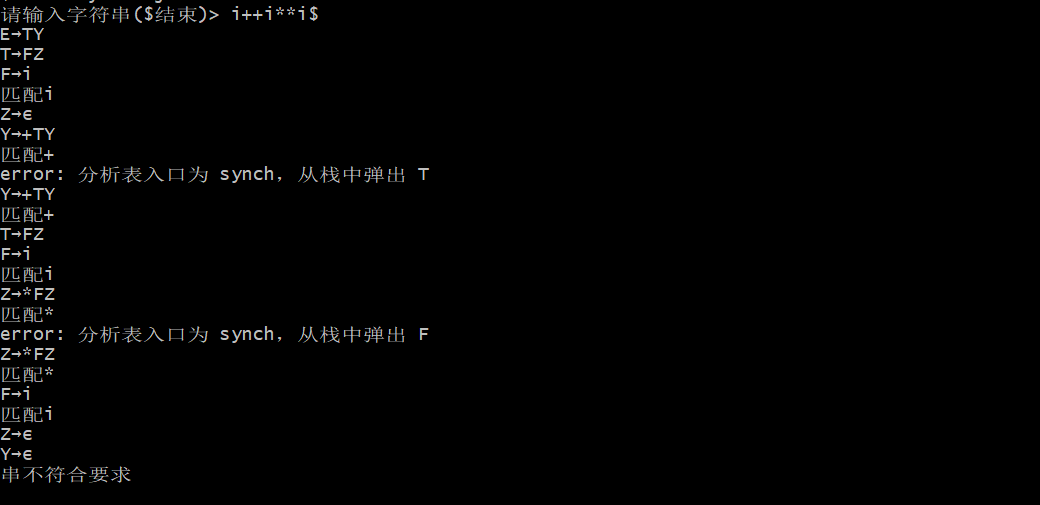
本实验设计思路：先判断输入的文法是否存在左递归，若有则消除之，并将文法放入数组GG[][]中；若不存在左递归，则将原文法放入数组GG[][]中。对GG[][]中的文法依次求解FIRST集、FOLLOW集、SELECT集，再根据SELECT集构造对应的预测分析表。对于输入的串，根据预测分析表来检查输入的文法是否正确。求解各个集合的结果以及预测分析表及匹配过程在界面中给出。

## 五、实验结果及分析

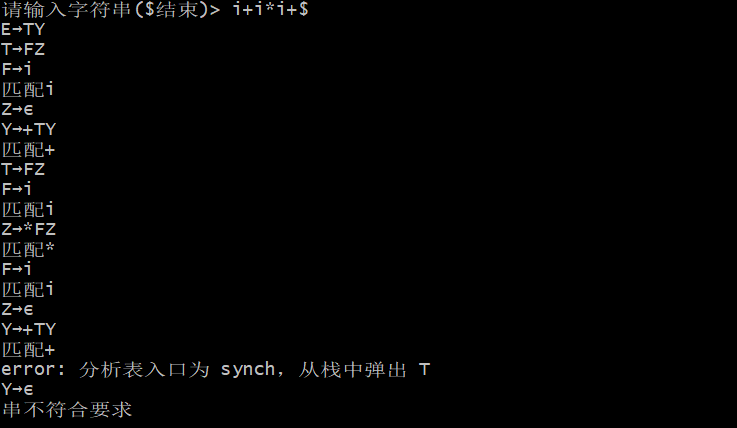




## 六、测试结果用例







## 七、问题解决

求解FIRST集时，刚开始发现无法设置参数检验一次文法执行完后是否每个FIRST集有变化，于是索性直接循环文法执行10次(假设10次以内FIRST集不再有变化)。在验收时询问老师发现可以在某个FIRST集添加元素时将flag置为1表示有变化，一次文法循环完后，若flag值为1则循环该文法直至某次flag值为0。FOLLOW集也可以采用这种方法，在试验验收后重新改修改程序完成了用flag记录的方法。

平时在作业中FIRST集和FOLLOW集很好求，但是在编程时自己还是想的不够有条理，借助了书上和老师所给pdf中的算法思想完成。

在最后根据预测分析表检验输入串的过程中，因为用了STL中的stack，但是在每次匹配之后输出分析栈剩下来的内容时，竟然，没有直接输出stack中全部元素的函数!(在Java中可以直接输出stack中的元素而且不需要调用函数)…还得写一个没有难度的输出栈所有函数也是够了…

## 八、实验感谢及体会

虽然实验要求针对该文法，但是在询问老师后发现其实是针对所有文法都要成立，因此难度有所增大，因为自己的编程能力有限写了好久才写出来，而且没有分成模块(完成后有做过分开成模块的工作，但是失败了)，代码的可读性不强，以后会注意提升自己。通过这次实验，对于FIRST集、FOLLOW集、SELECT集和LL(1)文法有了更深刻的理解和系统的求解过程了解。