****语法分析器实验****

LL语法分析器的设计与实现

**一、实验目的**

熟悉和掌握LL语法分析器的构造，加深对自上而下语法分析方法的理解。

**二、实验内容**

需要实现的功能：

1）构造文法的LL预测分析表；

2）构造LL语法分析器的总控程序；

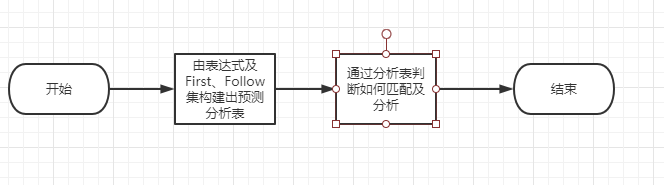
3）输入文法（语言语法结构的文法描述存储在文本文件中）；

4）输出文法的LL预测分析表（标准输出设备）；

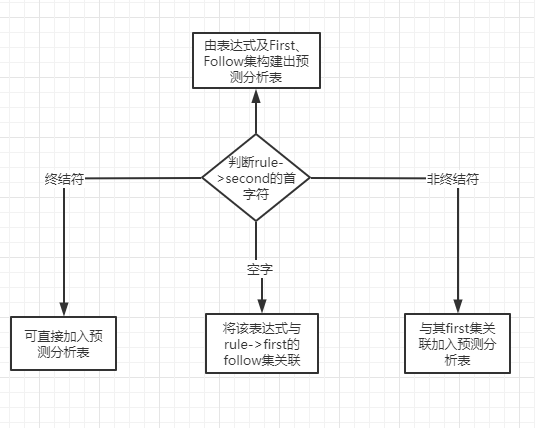
5）对给定的符号串（存储在文本文件中），输出其是否为该文法正确句子的判断，并输出文本形式的分析过程。

**三、分析模型**

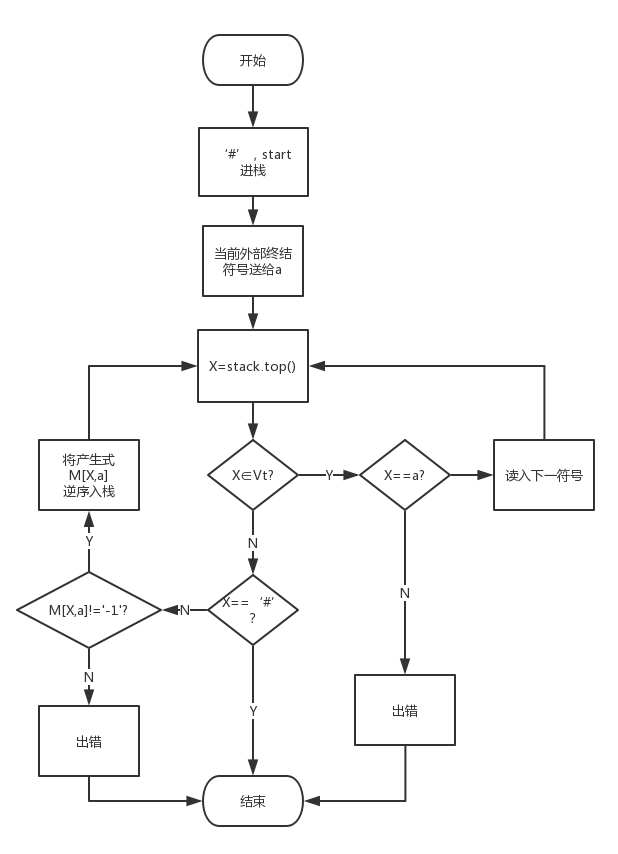
该模型是在文法构造一和二的基础上实现的，在First集、Follow集的基础上实现了预测分析表和对一个具体语句的分析过程。总流程如下：



其中第一步又可划分为：



第二步的流程图如下：



在以上思路确立之后，我们便可以开始模型的构建了。

**四、构建模型**

1、要说构建模型，通过这几次实验我算是明白了，关键就在于设计一个好的数据结构，实现起来就会简单很多。构建预测分析表，其实数据结构挺容易设计的，无非就是map<pair<char,char>,string>,其中pair中第一个char代表Vn，第二个char代表Vt，String自然就是->后面的那部分内容了。

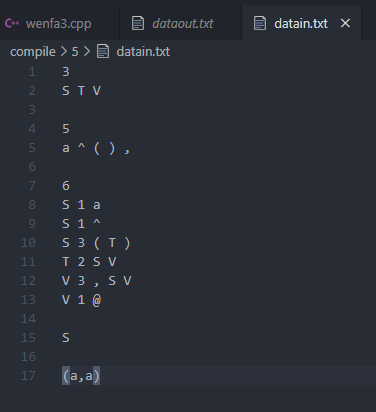
2、通过这个数据模型，遍历这些表达式，能很好地找到并存储满足预测分析表条件的表达式，然后通过一定的规则使他们输出就可以了

3、得到了预测分析表，给出分析过程也不是一件难事了。首先，栈这个数据结构是毋庸置疑需要用到的。然后就通过栈顶元素与str串的内容进行匹配，匹配成功就前进，不成功就在预测分析表找，找到了就逆序压进栈中，找不到那就出错。最后两个‘#’号相遇，就算匹配成功了。

4、分析完模型就该编码实现了，就一个函数make\_table完成就好了，代码量不是很多。

**五、实验数据**

Datain.txt



Dataout.txt



**六、实验总结**

1、通过这次实验，我认识到了这种循序渐进做实验的好处。一方面，不是一下子面对很大的工作量，不会让自己丧失信心；另一方面，在已有代码上的增添功能能让自己更加注意代码的重用的功能，这样子会让自己写出来的代码有了些条理，稍微能看些吧。

2。由于这次的实验不是很难，只需要设计个函数注意下输出格式就好了。但是还是要注意数据结构的构建的脑海里思路的构建。分析问题、问题建模、实现问题都是缺一不可的。

3、多动手、多实践、多思考。希望在以后的学习中能做到这些吧。