编译原理上机实习三：

题目：LR1 文法匹配字符串的设计与实现

一、 思路：

在做本题时，我先是自学了 LR0 文法和 SLR1 文法，才弄清楚整个 LR1 文法的由来。LR1 文法是一种规范性很强的文法，它解决了 LR0 文法的移进/规约冲突和 SLR1 文法的规约/规约冲突。因此它的规范性就更强，需要的条件也更加苛刻。在做本题时，首先要清楚一下一些概念：

（1） 什么是 LR1 的增广文法：增广文法是为了解决最终规约时不一致的问题而引进的，在我们平时做题时，在文法的第一条上面加入一条 S’—>S 就是使这个文法变成它的增广文法。

（2） 什么是展望字符，为什么会出现展望字符的概念：展望字符是为了解决 LSR1 文法带来的规约/规约冲突的问题才引入的，严格来说，它是 follow 集的子集，它能够让我们唯一识别出要规约的动作。展望字符只有在项目需要规约的时候才有用，对于其他非待规约的项目，展望字符是没有作用的。

（3） 展望字符的求法：

规则一：如果这条产生式是开始的项目，则这条项目的展望字符为 $

规则二：对于这样一个项目 [A→aBr x](其中 x 是该项目的展望字符)，则他的展望字符为 First(rx), 当 r 为空时，此时的展望字符集就是 x，称为继承的展望字符，否则称为自生的展望字符。

（4） 什么是 Closure 函数，它有什么作用：Closure 函数是为了计算项目集闭包而引出的概念，它能够有一个项目，引出他所有的闭包。

（5） 什么是 goto 函数：goto 函数还回的是项目集 I 对应于文法符号 X 的后继项目集闭包。他在计算状态改变的时候会用到。

有了上面的概念，我们在把题目分为 3 个阶段

阶段一：设计详细的数据结构，存储产生式，这一部分如果不想好，后面处理会比较麻烦

阶段二：由初始状态构造项目集闭包，计算出所有的状态

阶段三：由状态之间的转换关系计算 LR1 分析表

阶段四：由分析表匹配字符串

在上面的四个阶段，其实阶段二和阶段三可以同时进行。

二、 详细设计与编码：

数据结构：

（1） 文法类：具体形式是 ArrayList<Production> grammar,一个文法是由多个产生式组成，其中 Production 的数据结构见下：它是由左部和多个右部组成的。

```
public class Production {
    public char leftHand;
    public ArrayList<String> rightHand;
```

图 1 产生式数据结构图

（2） 状态类：具体形式是 ArrayList<item> Items，它是由多个项目组成的。item 类是单个项目类，它记录了点的具体位置，如 W→[.B，它的结构如下：

```
public class item {
    public char leftHand;
    public HashSet lookAheadSet;//展望字符
    public String leftOfDot;//点的左部
    public String rightCharOfDot;//点的右部
    public String rightOfDot;//这里是除了下一个字符的右部
```

图 2 项目数据结构图

（3） 存储 LR1 分析表的表格类 Table：它的结构比较简单，只有三个字段，从哪个转台来，经过哪个字符，对应的动作是什么。其中动作中 r2 表示用第条产生式来规约，s2 表示移进到状态 2，a 表示 accept 即接收状态。

```
public class Table {
    public ArrayList<Node> nodes;
    public void addNode(Node node){
        if(nodes==null){
            nodes=new ArrayList<>();
            nodes.add(node);
        }else{
            nodes.add(node);
        }
    }
```

图 3 LR1 分析表数据结构图

```
public class Node{
    public int state;
    public String symbol;
    public String action;
```

图 4 具体节点数据结构图

算法：

本题与上课讲的算法基本没有多大的区别，都是按照老师给的套路做的，只有一点需要注意，就是求展望字符的算法。

根据上面求展望字符的两条规则的第二条规则，(规则二：对于这样一个项目 [A　aBr x](其中 x 是该项目的展望字符)，则他的展望字符为 First(rx), 当 r 为空时，此时的展望字符集就是 x，称为继承的展望字符，否则称为自生的展望字符。)需要特别注意的是这里的 First(rx) 是求多个字符的 First 集，这一点与上课所讲不同，其算法是用递归实现的，简单来说就是一个字符一个字符求解，逐层拨开，如果您对于这一点感兴趣，可以看看本文附录里的附图一，这张图是代码的截图。

本题其他的算法基本与老师上课时所讲没有差别，这里不一一叙述。

三、 运行结果：

```
"C:\Program Files\Java\jdk1.8.0_181\bin\java.exe" ...
你输入的语法是：
W->[S]
S->[[B]
A->[i, [B]
B->[], C]
C->[A], A,C]
********************************************************
自动机构造完成：
状态0
W-->.S    *****展望字符：[$]
S-->.[B    *****展望字符：[$]
状态1
S-->[.B    *****展望字符：[$]
B-->.]    *****展望字符：[$]
B-->.C    *****展望字符：[$]
C-->.A]    *****展望字符：[$]
C-->.A,C    *****展望字符：[$]
A-->.i    *****展望字符：[]]
```

图 5 状态图部分截取

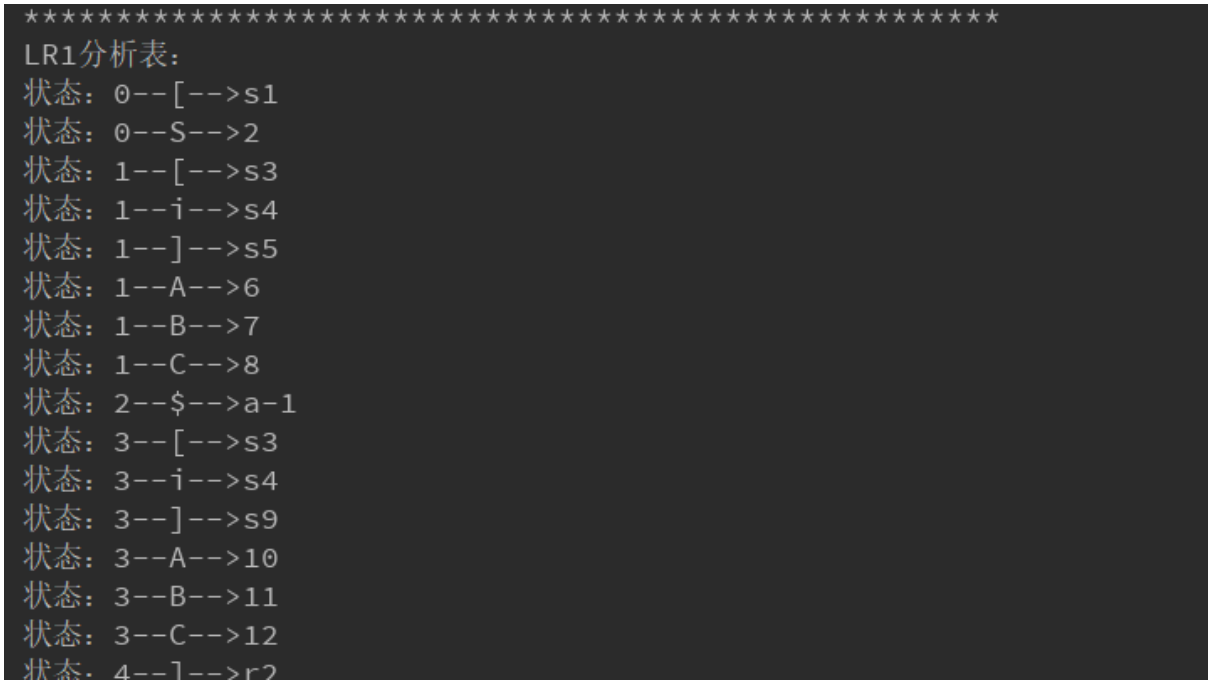

图 6　分析表部分截取

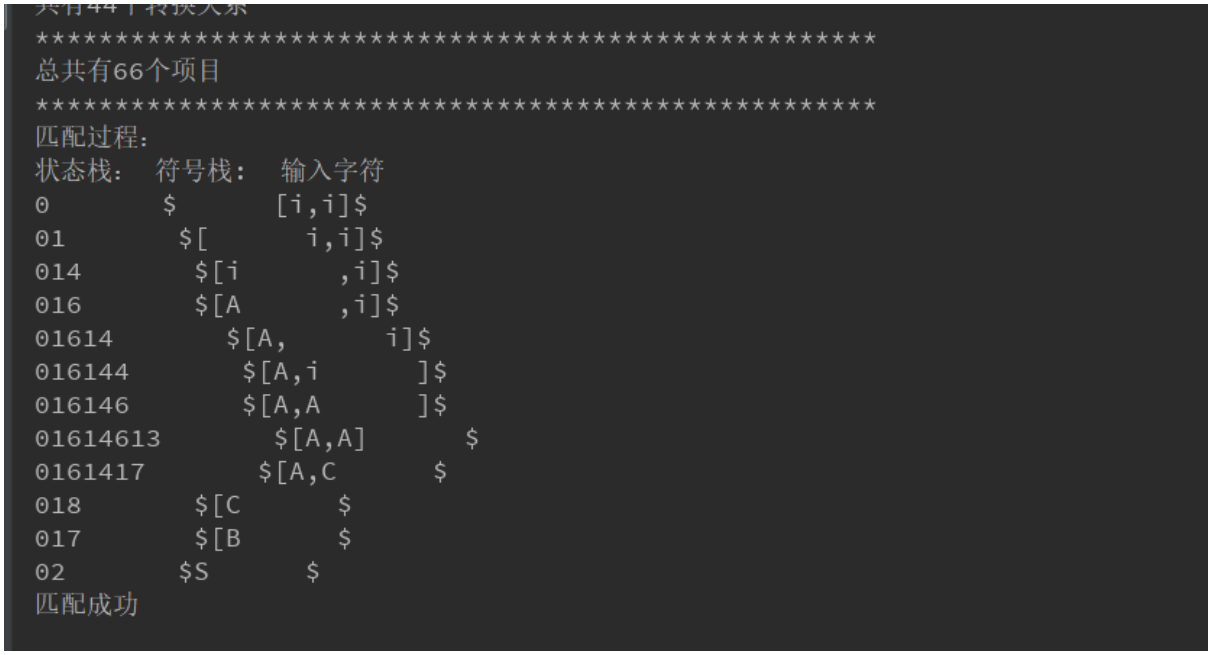

图 7 匹配情况截取

四、 出现的问题以及解决方法：

（1） 数据结构设计的不够合理

对于语法类数据结构设计的不够合理，导致后面计算太过繁琐，后来在语法类中加入新的字段，将原来数据转入到新的结构中，解决问题

（2） 代码设计用递归

一开始为了使程序更加易懂，所以我才用了递归的形式写代码，由于递归的实现并不是一件简单的事，所以在这里我浪费的大量的时间找错误，后来还是没能找到，于是把递归改成了非递归的形式

（3） 写代码效率不高

在这次实习中，由于采用错误的算法导致写了很多无用的代码，本题实际有效的代码量应该在 300 行左右，但是我的代码量曾一度上了 800 行，这里面很多事没用的代码，一方面这体现了我能力需要进一步提高，另一方面正是我之前大量的试错，我后面改代码起来才快，但是以后还是要注意写代码的效率问题，争取比较快的把代码写出来。

五、 总结与反馈：

经过一学期的编译原理的学习，使我深深的加深了对代码编译的理解，我想这一学期我的收获不仅仅是知识的收获，更是思想和自学能力上的提升。具体总结如下：

（1） 学习能力上的提升

由于课堂时间有限，老师所讲内容也很有限，为了弄清上课时没听懂的问题，我常用慕课、youtube 的工具进行学习，期间收获了大量课堂没有讲过的内容，我也曾拿着考研题研究别人的解题思路，但大多数时候，这些思路给的都更加适合人脑的思考，而不是代码的实现部分。例如在求 first 集的时候，有一本考研书上是这么写的，第一步：直接收取 A->a…第二步：反复传送 A->U…这种形式，我想这种形式是对的，但是只适合人脑的思考，不适合人在计算机上编程，所以我去 google 上搜集了很多解决方法，虽然这些资料大多都是英文的，但是我觉得他能给你讲的特别清楚。

（2） 思想上的提升

学完编译原理我觉得给我最大的收获不是我学了多少编译原理方面的知识，而是我懂得了自己去查阅资料，自己去读和看英文的资料所带来完全不一样的知识体系。比如如果现在让我去讲编译原理我觉得我讲的就会和老师的不一样，因为我们接触的资料不一样，整个知识体系的构件过程也不一样，所以讲的也一定会不一样。我觉得在这个时代，最重要的不是你做了多少东西，而是你做了多少创新，这些创新往往是小的，但是却非常与众不同。

（3） 专业知识的提升

最后就是专业知识的提升，经过一学期的学习，我更加了解了编译原理相关的知识，这为我后面的继续学习打下了坚实的基础。

之前还指望能写出编译器拿到一个好的分数，但是现在看来有点不可能了，但是我这里推荐一个好的资源可以降低写出编译器的难度，希望下届的学弟学妹们能够把这项任务完成。网易云上的教程，手写编译器，网址
https://study.163.com/course/introduction/1004632002.htm?share=1&shareId=1151765179

给老师的建议：

（1） 希望课堂讲的内容会多一点，如果课堂讲不完，可以推荐同学们课下去慕课等学习平台学习

（2） 平时课堂练习一定不能少，就是靠练习，才能是学生的心思在课堂上，也更能理解课堂知识

（3） 一定要把学生分享这个活动做下去，老师的语言和学生的语言往往是不一样的，学生是实践者遇到实际的问题也会更过，老师往往会站在一个比较高的起点看问题，解决的思路不一样

（4） 如果您能接纳我的建议，我觉得我这门课就没有白上。

六、附录：

```
public void firstSet(String str,HashSet firstset){//传入一个字符或者字符串，求他的first集
    if(str.length()==0){return;}
    String s0=str.substring(0,1);
    if(isUpper(s0)){
        for(Production p:g){
            if(s0.equals(String.valueOf(p.leftHand))){
                for(int j=0;j<p.rightHand.size();j++){
                    if(p.rightHand.get(j).equals("@")){
                        if(str.length()!=1){
                            firstSet(str.substring(1),firstset);
                            //firstset.add(result);
                        }else{
                            firstset.add("@");

                        }
                    }else{
                        firstSet(p.rightHand.get(j),firstset);
                    }
                }
            }
        }
    }else{
        firstset.add(s0);
    }
}
```

附图 1 求 First 集的算法代码截图