

---

# 词法、语法分析

## 一、实验内容与要求

### (1)实验目的

给出 PL/0 文法规范，要求结合编写 PL/0 语言的词法、语法分析程序。

### (2)实验内容

已给 PL/0 语言文法，构造表达式部分的词法和语法分析器。

### (3)实验要求

将实验一“词法分析器”与实验二“语法分析器”之间的衔接方式由独立一遍改为独立子程序。

语法分析器的编写方法采用递归子程序法。

输入：

PL/0 表达式源语言，例如：(a+15)\*b 作为输入。

输出：

对于语法正确的表达式，报告“语法正确”；

对于语法错误的表达式，报告“语法错误”，指出错误原因。

---

## 二、实验分析与设计

由于在做第一个实验的时候就考虑到了要做第三个实验，于是我当时就把词法分析器做的非常细致，对其中的每一个函数和变量都充分考虑到了他们的可充用性，因此，将一遍的词法分析语法分析改为词法语法分析并不是很困难。

对于把词法分析器作为一个子过程，用语法分析器来调用是比较常用的编译方式，不必要再编译错误的时候一次性将所有的源程序都进行词法分析，可以一边对语法分析，一边对词法分析，大大节省了时间和空间的开销。

二者的区别主要在于，词法分析器不能作为独立的一遍，因此不能一直对词法分析自动机进行循环操作，在此主要是让语法分析调用词法分析自动机，一次识别一个单词。

因此对于语法分析来说，**Advance** 函数就不是从词汇表中读取一个单词了，应该改成调用一次词法分析自动机识别出一个单词，之后的操作完全相同，但是需要注意的是，由于词法分析器中指针的影响，需要将上一次分析结束为止的索引 **index** 传回词法分析自动机，以保证词法分析的一致性：

```
//读取下一个终结符号
void Advance()
{
    if(index >= lexString.length())
    {
        ch = "输入串不符合规范";
        chType = "";
        return;
    }

    index = FA.onFA(lexString, index);

    try {
        BufferedReader bufferedReader = new BufferedReader(new FileReader("Lex2.txt"));
        String result = bufferedReader.readLine();
        ch = result.split(",")[1];
        chType = result.split(",")[0];
    } catch (Exception e) {
        System.err.println("Lex2.txt Not Found");
        System.exit(0);
    }
}
```



---

```
public int onFA(String str,int tag)
{
    index = tag;
    getBC(str);

    if(index>=str.length())
    { //避免读入的序列最后是空格而导致异常
        return -1;
    }

    PrintStream ps = null;
    try {
        ps = new PrintStream(new FileOutputStream("Lex2.txt",false));
    } catch (FileNotFoundException e) {
        System.err.println("Lex2.txt Not Found");
        e.printStackTrace();
    }

    if(isLetter())
```

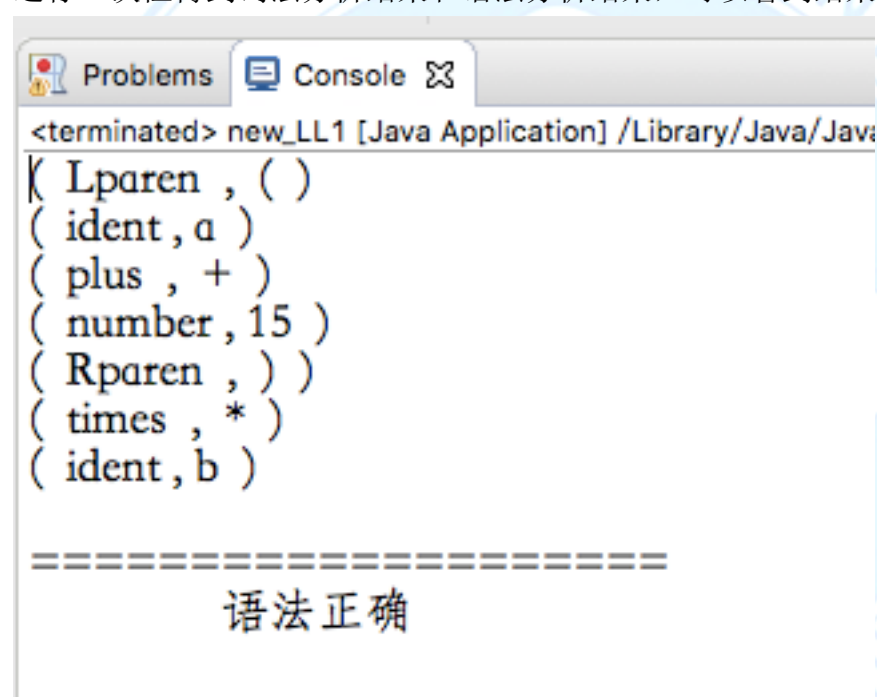
#### 四，调试结果

输入一段程序：



`(a+15)*b`

运行一次性得到词法分析结果和语法分析结果，可以看到结果正确无误：



```
<terminated> new_LL1 [Java Application] /Library/Java/Java
( Lparen , ( )
( ident , a )
( plus , + )
( number , 15 )
( Rparen , ) )
( times , * )
( ident , b )

=====
语法正确
```

异常错误演示：输入错误的表达式：



`(a+15)**b`

词法分析器不会报错，但语法分析会报错：

```
Problems Console
<terminated> new_LL1 [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_121
( Lparen , ( )
( ident , a )
( plus , + )
( number , 15 )
( Rparen , ) )
( times , * )
( times , * )


=====

Error :      -----语法错误-----

*: times  无法识别
```

异常错误演示：输入不完整的表达时：

```
input — 已编辑
(a+15)*
```



词法分析器不会报错：

但语法分析会报错：

```
Problems Console
<terminated> new_LL1 [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_121.jc
( Lparen , ( )
( ident , a )
( plus , + )
( number , 15 )
( Rparen , ) )
( times , * )

=====

Error :      -----语法错误-----

*: times  无法识别
```

当输入错误的表达式，词法分析直接报错，不进行语法分析：

