

北京交通大学
BEIJING JIAOTONG UNIVERSITY

《编译原理》 实验报告

实验名称:	专题 2_递归下降语法分析设计原理与实现
学 号:	
姓 名:	
学 院:	计算机与信息技术学院
日 期:	2022 年 10 月 17 日

目录

1. 实验目的	3
2. 实验要求	3
3. 程序实现	3
3.1. 相关环境介绍	3
3.2. 主要数据结构	3
3.3. 程序结构描述	4
3.3.1. 设计方法	4
3.3.2. 函数定义	4
4. 程序测试	4
5. 实验汇总	6
5.1. 技术难点及解决方案	6
5.2. 实验感想和经验总结	6

1. 实验目的

通过实验，掌握 LL (1) 文法及其判定；无回朔的递归下降分析的设计与实现实验内容。

2. 实验要求

完成四则运算描述赋值语句的 LL(1)文法的递归下降分析程序。

3. 程序实现

3.1. 相关环境介绍

操作系统：window 10 21H2

开发环境：Clion-2022.2.1-Windows

编译器：mwing-10.0

3.2. 主要数据结构

主要是单词的信息保存，建立了一个 struct

```
01:struct Keyword{
02:    string notation;
03:    int class_num;
04:    int line;
05:    Keyword(string str, int num, int line_){
06:        notation = str;
07:        class_num = num;
08:        line = line_;
09:    }
10:    Keyword(char* str, int num, int line_){
11:        notation = string(str);
12:        class_num = num;
13:        line = line_;
14:    }
15:    Keyword(char str, int num, int line_){
16:        notation = str;
17:        class_num = num;
```

```
18:         line = line_;
19:     }
20:};
```

其中 notation 为单词的值，class_num 为单词所属的类别，line 是单词在源程序中的行号。

3.3. 程序结构描述

3.3.1. 设计方法

根据专题 1 的输出结果，读入进行分析。也可以直接在程序中调用专题 1 的程序，进行词法分析。

读入之后按照递归调用函数流程图进行编写。

程序要保证一定的提示信息，可以提示本句是否满足语法结果即可。

3.3.2. 函数定义

int init()

实验初始化函数。读入，专题 1 输出的标准终结符号集的分类号。

int alu_analysis(string file_path)

语法分析入口程序，完成读入专题 1 输出的分词结果，调用推导的函数判断是否符合语法和错误提示等功能。

Int statement()、int expression()、int M() int factor()、int expression_1()、int term()、int term_1()、int arithmetic()，为按照课程中的递归调用流程图给出的推导式对应的函数。int goon () 为读入函数。

int correct_prom() 语句符合文法提示。

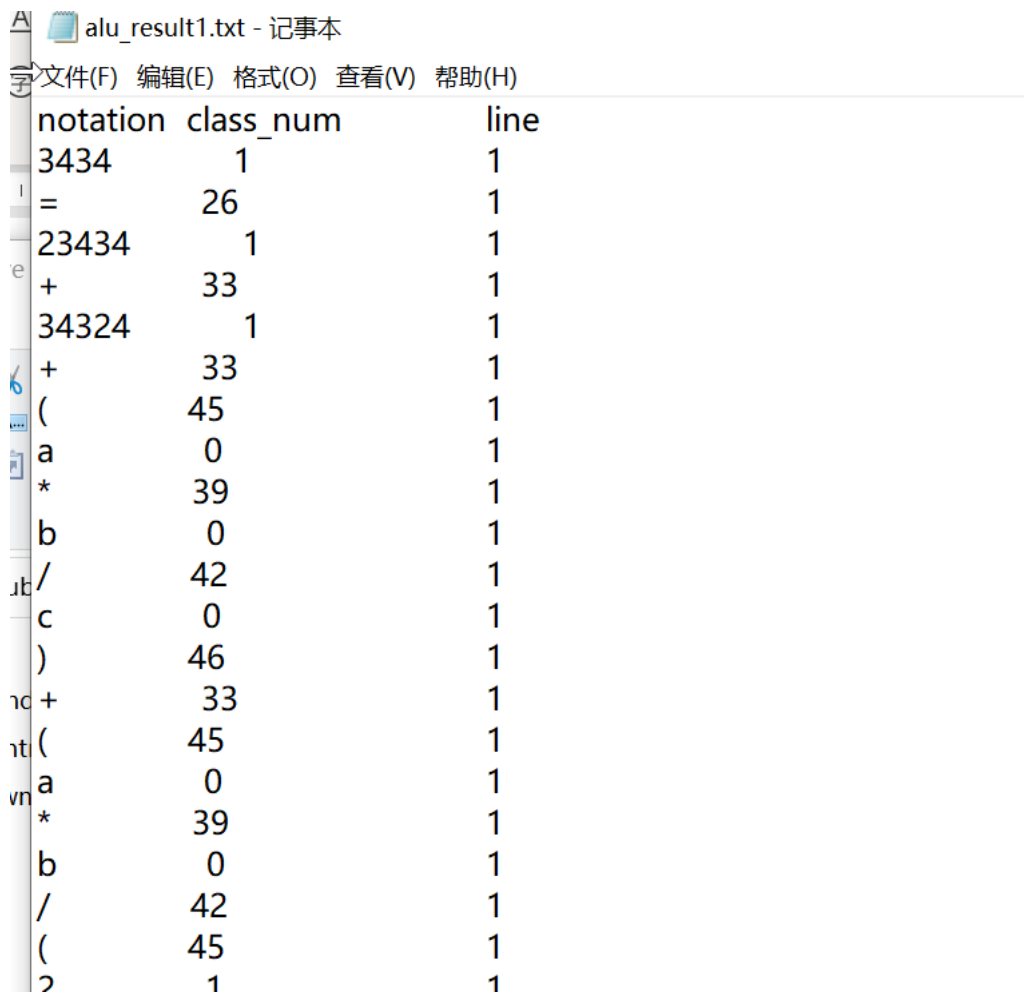
int error_prom() 语句不合文法提示。

4. 程序测试

Alu_test1.txt

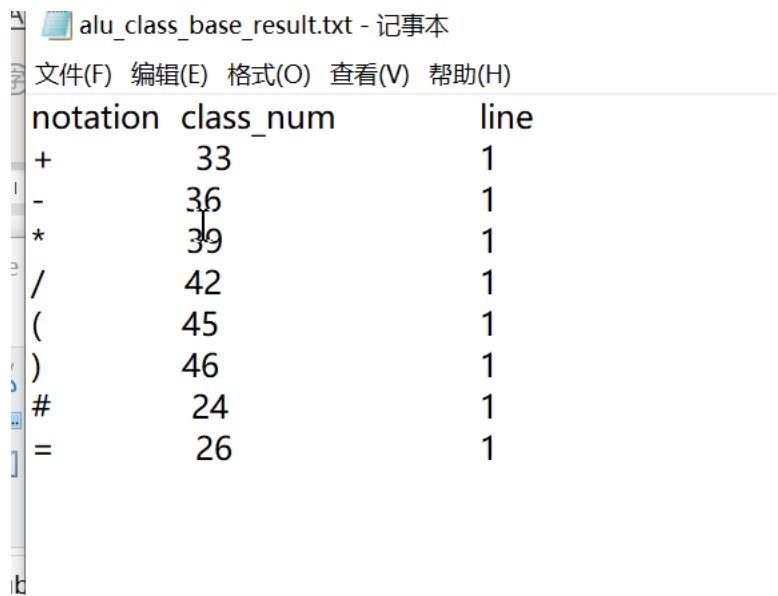
这里一个文件中包含了 4 个测试样例。

```
01:3434 = 23434+34324+(a*b / c) +(a*b/ (2-2)) #
02:a = ( a+ b -c ) * 32 / a #
03:b=(a+c * b #
04:k=a- b *c #
```



notation	class_num	line
3434	1	1
=	26	1
23434	1	1
+	33	1
34324	1	1
+	33	1
(45	1
a	0	1
*	39	1
b	0	1
/	42	1
c	0	1
)	46	1
+	33	1
(45	1
a	0	1
*	39	1
b	0	1
/	42	1
(45	1
2	1	1

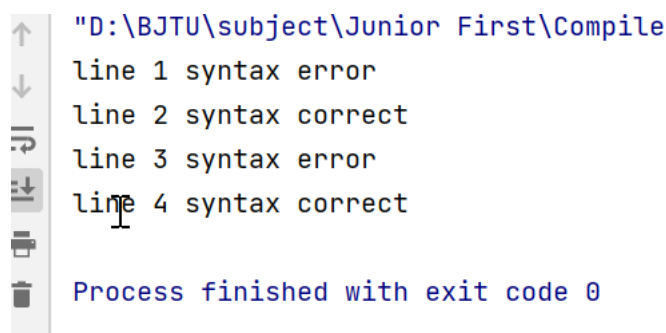
图表 4-1 test1 中词法分析结果



notation	class_num	line
+	33	1
-	36	1
*	39	1
/	42	1
(45	1
)	46	1
#	24	1
=	26	1

图表 4-2 config 文件

由专题 1 输出的四则运算终结符号集类别集，相当于本程序的 config 文件。这里仍然采用了三元组，多了一个行号，方便错误提示，更多的我们直接用了 Keyword 结构，这对结果没有影响。



```
"D:\BJTU\subject\Junior First\Compile
line 1 syntax error
line 2 syntax correct
line 3 syntax error
line 4 syntax correct
Process finished with exit code 0
```

图表 4-3 本程序结果，信息提示

通过信息提示可以看出，程序对 4 个测试样例完成的很好，结果正确。

5. 实验汇总

5.1. 技术难点及解决方案

实验本身的重要程序框图，已经在课堂上教授过了，主要是代码编写过程中的问题有些多，所幸最后都解决了。

这里的实验和课堂上讲述的就是多了一个赋值语句，但是给的也十分的简单，如果 $V \rightarrow E$ 这样就是一般的情况，实验要求的是 $V \rightarrow I$ 这样会简单一些，直接判断就可以了。

5.2. 实验感想和经验总结

实验要是递归的编写。一开始没有设置识别成功的出口，导致错误。而且因为递归深度太大，不好 debug，这里着实花了一点时间。

一开始在 `expression_1` 和 `term_1` 中的循环结构，就是直接调用自身，后来发现这样做递归深度太大，完全可以写成 `while-continue` 结构，这样程序效率更高，还方便 debug。