专题 1\_ 词法分析程序构造原理与实现

李若森 13281132 计科1301

# 程序功能描述

**[功能]：**

完成下述正则文法所描述的C语言子集单词符号的词法分析程序。

**[要求]：**

1. 给出各单词符号的类别编码。
2. 能发现输入串的错误。
3. 将分析所得二元序列输出到中间文件中。

**[文法]：**

<标识符>→c|c<余留标识符>

<余留标识符>→d|c

<实数>→\((+|-)<无符号数>|d<余留无符号数>|.<小数部分>|d\)

<无符号数>→d<余留无符号数>|.<小数部分>|d

<余留无符号数>→d<余留无符号数>|.<十进小数>|(E|e)<指数部分>|.|d

<十进小数>→(E|e)<指数部分>|d<十进小数>|d

<小数部分>→d<十进小数>|d

<指数部分>→d<余留指数>|(+|-)<整指数>|d

<整指数>→d<余留整指数>|d

<余留整指数>→d<余留整指数>|d

<算数运算符>→+|-|\*|/|++|--

<关系运算符>→>|<|==|>=|<=|!=

<逻辑运算符>→!|&&|\|\|

<位操作运算符>→>>|<<

<赋值运算符>→=|+=|-=|\*=|/=|%\

<特殊运算符>→,|(|)|{|}

<分隔符>→;

保留字: void int float double if else for do while

**[说明]：**

1. 该语言对大小写不敏感
2. c代表字母a-z&&A-Z，d代表数字0-9。
3. “/\*..\*/”以及“//”为程序注释部分。

# 主要数据结构描述

**pair<int, string>:**

用pair<int, string>来存储单个二元组。其中第一个元素为类型号，第二个为元素的值。当类型号小于40时代表程序分界符，第二个元素不存储有效信息，用“-”代替；类型号为40时是标识符，第二个元素存储标识符字符串；类型号为41时代表实数，第二个元素存储的是该实数的二进制值。

**vector<>:**

vector是C++中的动态数组，用来存储每一行的二元组。

# 程序结构描述

## 函数定义：

|  |
| --- |
| **lindAnalyse:** |
| bool lineAnalyse( string &line, int &colNo ); |
| **功能：** |
| 行词法分析 |
| **传入参数：** |
| line:包含该行的字符串 |
| colNo:起始处理位置 |
| **传出参数：** |
| colNo:处理完成位置 |
| **返回值：** |
| 该行是否成功解析。成功返回true，否则返回false。 |

|  |
| --- |
| **getOutputName:** |
| void getOutputName( char \*inputName, char \*outputName ); |
| **功能：** |
| 得到输出文件名 |
| **传入参数：** |
| inputname:需处理的文件的文件名 |
| outputname:输出文件的文件名指针 |
| **传出参数：** |
| Outputname:输出文件的文件名 |
| **返回值：** |
| 无 |

|  |
| --- |
| **errMsg:** |
| void errMsg( string filename, int rowNo, int colNo ); |
| **功能：** |
| 向屏幕输出错误信息 |
| **传入参数：** |
| filename:正在处理的文件的文件名称 |
| rowNo:出错行 |
| colNo:出错列 |
| **传出参数：** |
| 无 |
| **返回值：** |
| 无 |

|  |
| --- |
| **spaceProcess:** |
| void spaceProcess( char \*line ); |
| **功能：** |
| 预处理，删除字符串内空格 |
| **传入参数：** |
| line:需处理的字符串 |
| **传出参数：** |
| line:处理过的字符串 |
| **返回值：** |
| 无 |

|  |
| --- |
| **noteProcess:** |
| bool noteProcess( string &str, bool &noteflag ); |
| **功能：** |
| 预处理，删除字符串内注释 |
| **传入参数：** |
| str:需处理的字符串 |
| Noteflag:该字符串是否处于多行注释中 |
| **传出参数：** |
| str:处理过的字符串 |
| **返回值：** |
| 是否成功处理。如果注释匹配则返回true，否则返回false。 |

|  |
| --- |
| **print:** |
| void print( FILE \*fp ); |
| **功能：** |
| 将结果输出到文件中 |
| **传入参数：** |
| str:需处理的字符串 |
| Noteflag:该字符串是否处于多行注释中 |
| **传出参数：** |
| str:处理过的字符串 |
| **返回值：** |
| 是否成功处理。如果注释匹配则返回true，否则返回false。 |

# 程序测试

# 专题总结