编译原理

实

验

报

告

**实验名称：****\_\_\_\_\_实验一：词法分析\_\_\_\_\_\_\_\_**

**姓名：\_\_\_\_\_\_\_\_\_\_\_\_李秋阳\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**学号：\_\_\_\_\_\_\_\_ \_180110527\_\_ \_\_\_\_\_\_\_\_\_\_\_**

**学院：\_\_\_\_\_\_计算机科学与技术学院\_\_\_\_\_\_\_\_\_**

**专业：\_\_\_\_\_\_ \_\_\_计算机类\_\_\_\_\_\_\_\_\_\_\_****\_\_\_\_**

1. **实验目的与方法**

**实验目的：**

（1）通过本实验加深对词法分析程序的功能及实现方法的理解

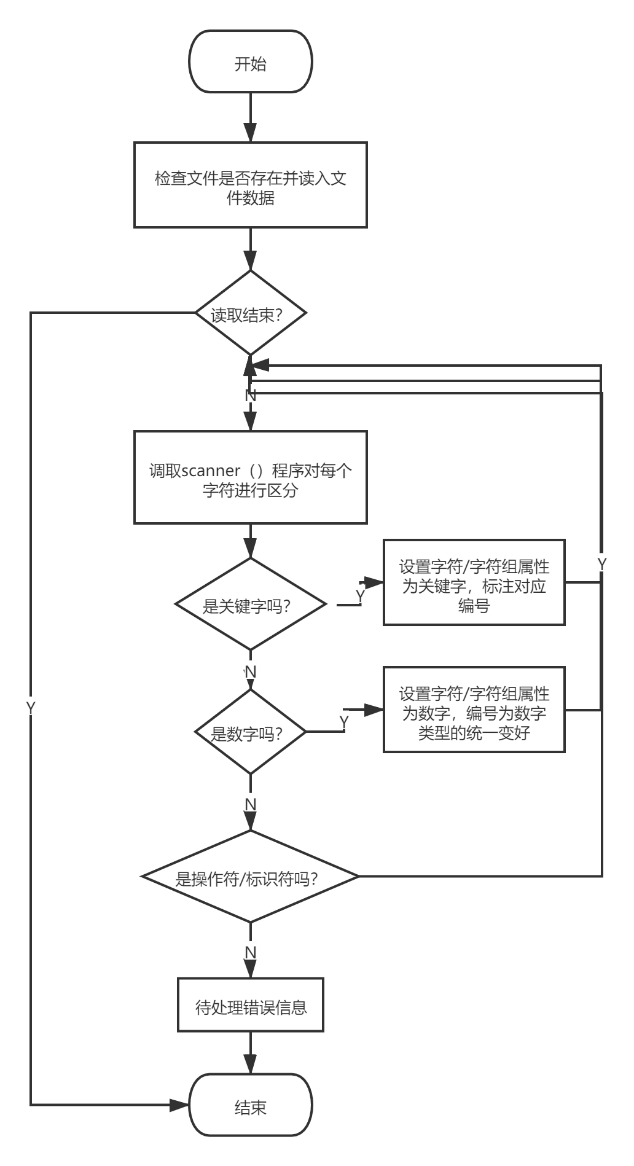
（2） 设计并编程实现一个词法分析程序，对C语言源程序段进行词法分析，加深对高级语言的认识

（3）对C语言的文法描述有更深的认识，体会有穷自动机，编码表和符号表在编译的整个过程中的应用

**实验方法：**

C++14 + Clion 2019.2.5

1. **实验总体流程与函数功能描述**

**程序流程图：**

1. **实验内容**

**函数实现：**

**关键字/变量字符判断：**

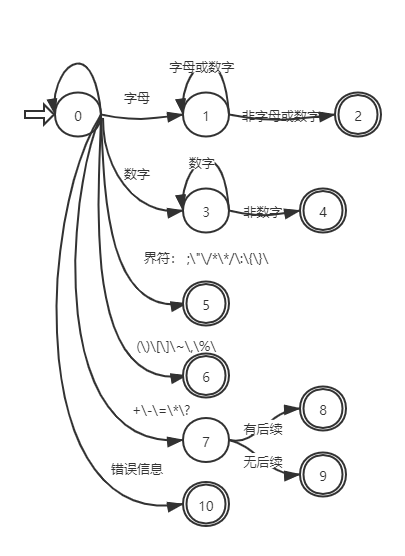
**int** reserve(){  
 **int** i = 0;  
 **while** (strcmp(keywords[i],**\_KEY\_WORDEND**) != 0){  
 **if**(!strcmp(keywords[i],token)){  
 **return** i + 1;  
 }  
 i++;  
 }  
 **return** 100;  
}

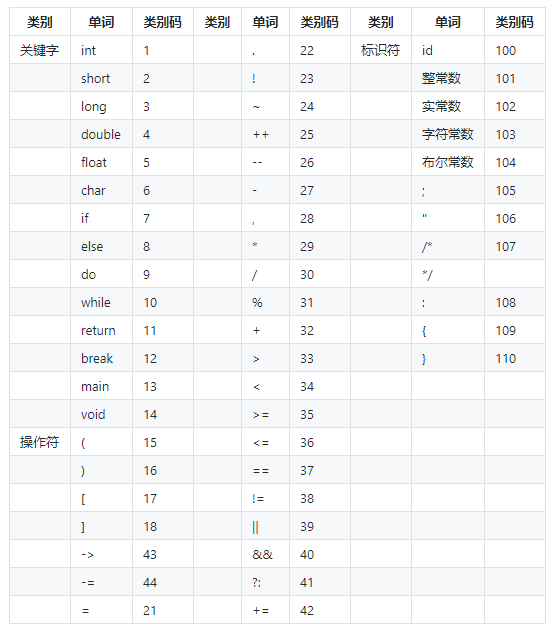
**取字符判断与空值处理：**

**char** m\_getch(){  
 ch = input[p\_input];  
 p\_input++;  
 **return** ch;  
}  
  
**void** get\_blank(){  
 **while** (ch == **' '** || ch == 10)  
 {  
 ch = input[p\_input];  
 p\_input++;  
 }  
}

**字符扫描处理节选（’+’相关）：**

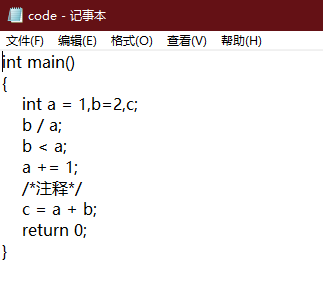
**case '+'**:  
 m\_getch();  
 **if**(ch == **'+'**){  
 inner\_word->type = 25;  
 inner\_word->word = **"++"**;  
 **return**(inner\_word);  
 } **else if** (ch == **'='**){  
 inner\_word->type = 42;  
 inner\_word->word = **"+="**;  
 **return**(inner\_word);  
 } **else**{  
 retract();  
 inner\_word->type = 32;  
 inner\_word->word = **"+"**;  
 **return**(inner\_word);  
 }

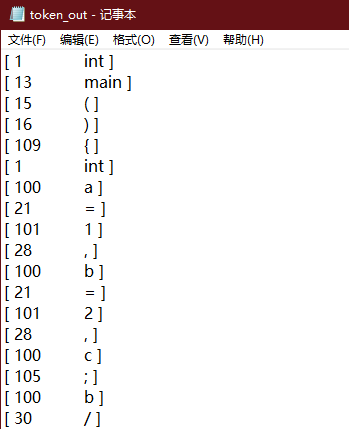
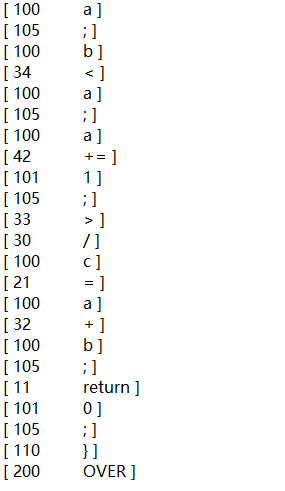
**有穷自动机状态转换图：**

**词类编码表：**

1. **实验结果与分析**

**输入：**



**输出：**

1. **实验中遇到的困难与解决办法**

实验在定义词类编码表的时候没有比较系统的思路，因此基本采用的c语言的词法并做了一定的简化，在后续的实验中可能会对词类定义进行进一步完善。