

编号: CS65

实习 成绩	一	二	三	四	五	六	七	八	九	十	总评	教师签名

武汉大学计算机学院
《编译原理》课程
词法分析
实习报告

班 序 号: 20202021605

实习题目: mini 语言词法分析

专业 (班): 计算机科学与技术三班

学生学号: 2018302110169

学生姓名: 孙嘉曦

任课教师: 杜 卓 敏

2021 年 6 月 22 日

目 录

第一部分	语言形式化描述	1
第二部分	单词编码表	2
第三部分	状态转换图	3
第四部分	词法分析算法	4
第五部分	测试计划	5

Mini 语言的要求如下:

Mini 语言包含简单算术表达式、赋值语句以及 IF 语句;
程序以#结尾,当有多个#时在预处理阶段对程序进行分段;
词法分析后输出单词表和单词串;
以文件中行为标记信息返回错误位置;
允许的字母为 a-z,数字为正整数。

规定的各符号类型有:关键字 (if else int)、标识符、常数、运算符 (+ - * /
== != <= >= < >), 界限符: (() { } ;)

第一部分 形式化语言描述

<关键字>→int|if|else

<标识符>→<字母>|<标识符><字母>|<标识符><数字>

<常量>→<无符号整数>

<无符号整数>→<数字序列>

<数字序列>→<数字序列><数字>|<数字>

<字母>→a|b|c|.....|x|y|z

<数字>→0|1|2|3|4|5|6|7|8|9

<加法运算符>→+|-

<乘法运算符>→*|/

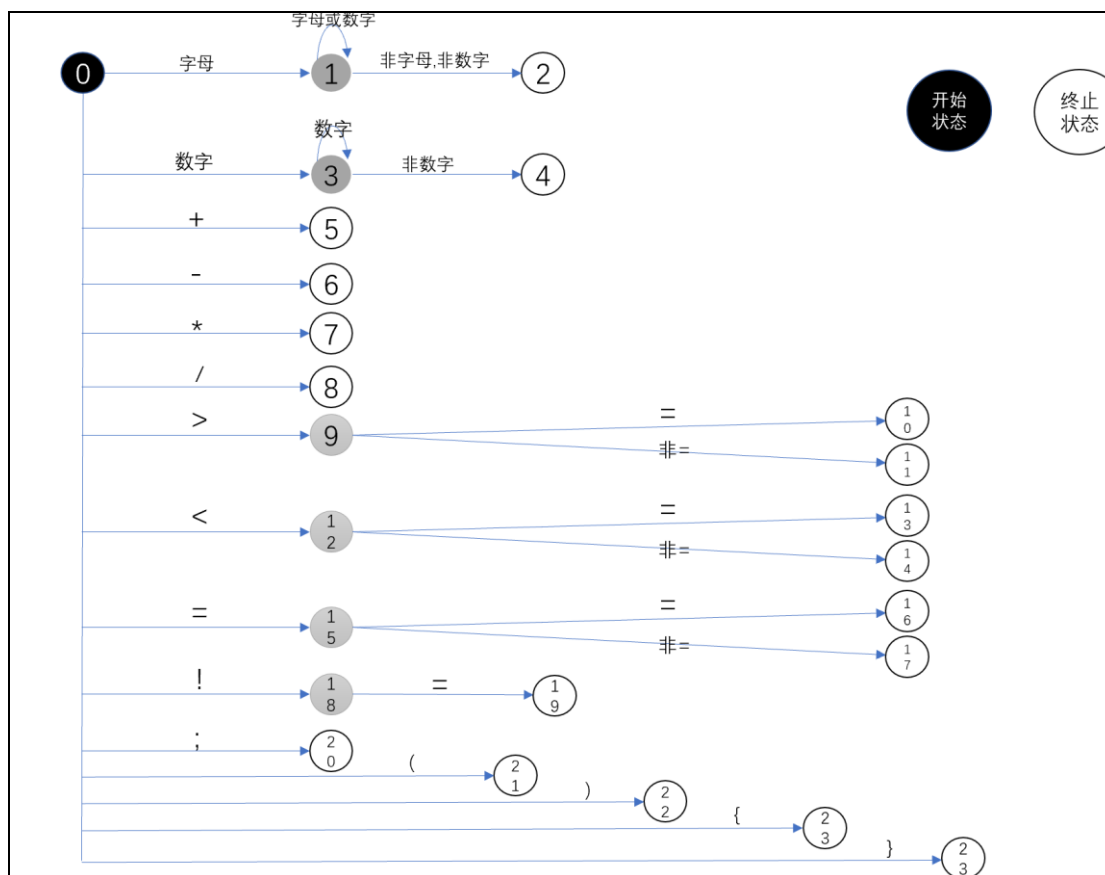
<关系运算符>→|=|<|>|=|<|=|=|!=

<分界符>→;|()|{ }

第二部分 单词编码表

标识符	1	标识符表地址
常量	2	常量表地址
If	3	
Else	4	
Int	5	
+	6	
-	7	
*	8	
/	9	
=	10	
>	11	
<	12	
>=	13	
<=	14	
==	15	
!=	16	
;	17	
(18	
)	19	
{	20	
}	21	

第三部分 状态转换图



第四部分 词法分析算法

1、预处理程序

首先以#为结尾把源程序分段，并依此产生相应的符号表等信息。

删除不必要的空格,跳格,注解,回车等部分。

发现不允许使用的符号即报错。

以行为单位处理程序，每次未能正确进行状态转换、错误符号、未声明等错误都返回错误行号。

2、数据结构

枚举：根据状态转换图来看，状态是有限多个的，符号的类别也是有限的，故采取枚举类型来表示各属性字，可以进行数字与属性字之间的转换使得表示更加简洁。

标识符表：

定义：每新定义一个标识符，首先把标识符类型 1 以及标识符名称保存到标识符表内，如果表内存在该声明，报错。之后返回地址，与类型一同作为 token 保存。

使用：每次使用标识符，首先查表是否存在此标识符，即是否声明。之后返回地址。

常量表：每使用一个新的常量，先查常量表，如果存在此常量，就把其地址返回，与常量类型配对，作为输出符号串保存。如果不存在此常量。就把常量的类型 2，以及常量的值保存到一个常量表中。

3、依据状态转换图实现的分词

依次遍历整段源程序，根据状态转换图，到达终止状态后，返回对应单词的 token 序列。例如：

输入字母，先保存到某一空间记录，之后循环接收字母和数字，当接收到非数字非字母，比如空格时，结束状态迁移。之后对该单词进行解析，分析是否为关键字，并据此解析 token 序列并保存。

输入数字。之后若接收到字母，则产生 default，报错。另外因为会回退一个符号，保证识别的准确。因为此时预判到即将产生非字母开头的标识符。若为数字，则继续读取，直至结束。

第五部分 测试计划

1、分段测试

期望预处理算法把程序分成三段

```
int data1=1;#  
int data2=2;#  
int data3=3;#
```

2、空格、换行、注解测试

期望消除多余的空格\换行\注解部分.

```
//一段注释  
int data=1;//init  
  
data=2;///
```

3、不合法字符测试

识别出%以及”甲”不合法的字符。

```
if (%==0){}#  
int data=1;  
if (data==1)  
{  
|    int 甲=1;  
}  
#
```

4、 不合法标识符测试

会对“1a”报错，而暂时不会对未声明的 data 反应

```
int 1a=1;  
data=1;
```

5、 不合法运算符测试

对非完整运算符 !=ch 报错

```
if (data!0) {}#
```

6、 完整分析测试

预期输出完整正确 token 串和标识符、常量表。

```
int data1=1;  
int data2=2;  
int data3=data1+data2;  
if (data3<=4) data4=1;  
else data4=0;#
```