

# 实验 1 词法分析程序设计

## 【开发语言及实现平台或实验环境】

C/C++/C#, JAVA 等

Microsoft Visual Studio 6.0/ Microsoft Visual Studio .NET, Eclipse 等

## 【实验目的】

- (1) 理解词法分析在编译程序中的作用
- (2) 加深对有穷自动机模型的理解
- (3) 掌握词法分析程序的实现方法和技术

## 【实验内容】

对一个简单语言的子集编制一个一遍扫描的词法分析程序。

## 【实验要求】

- (1) 待分析的简单语言的词法

- 1) 关键字

begin if then while do end

- 2) 运算符和界符

:= + - \* / < <= > >= <> = ; ( ) #

- 3) 其他单词是标识符(ID)和整形常数(NUM)，通过以下正规式定义：

ID=letter(letter|digit)\*

NUM=digitdigit\*

- 4) 空格由空白、制表符和换行符组成。空格一般用来分隔 ID、NUM、运算符、界符和关键字，词法分析阶段通常被忽略。

- (2) 各种单词符号对应的种别编码

单词符号	种别码	单词符号	种别码
begin	1	:	17
if	2	:=	18
then	3	<	20
while	4	<>	21
do	5	<=	22
end	6	>	23
letter(letter digit)*	10	>=	24
digitdigit*	11	=	25
+	13	;	26
-	14	(	27
*	15	)	28
/	16	#	0

- (3) 词法分析程序的功能

输入：所给文法的源程序字符串

输出：二元组（syn, token 或 sum）构成的序列。

syn 为单词种别码；

token 为存放的单词自身字符串；

sum 为整形常数。

例如：对源程序 `begin x:=9;if x>0 then x:=2*x+1/3;end#` 经词法分析后输出如下序列：(1, begin) (10, 'x') (18, :=) (11, 9) (26, ;) (2, if).....

### 【实验步骤】

(1) 根据图 1.1 构建主程序框架

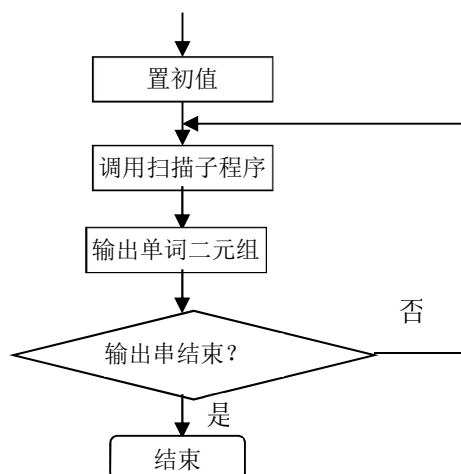


图 1.1 词法分析主程序示意图

代码提示：

```
main()
{
    p=0;
    printf("\n please input string:\n");
    do{
        输入源程序字符串，送到缓冲区 prog[p++] 中
    }
    while(ch!='#');
    p=0;
    do
    {
        scanner();//调用扫描子程序
        switch(syn)
        {
            case 11: 输出（数的二元组）; break;
            case -1: 输出（错误）; break;
            default: 输出（其他单词二元组）;
        }
    } while(syn!=0);
}
```

```
}
```

## (2) 关键字表置初值

关键字作为特殊标识符处理，把它们预先安排在一张表格中（关键字表），当扫描程序识别标识符时，查关键字表。如能查到匹配的单词，则为关键字，否则为一般标识符。

## (3) 编写扫描子程序

代码提示：

```
scanner()
{
    .....
    读下一个字符送入 ch;
    while(ch == ' ')    读下一个字符;
    if(ch 是字母或数字)
    {
        while((ch 是字母或数字))
        {
            ch=>token;
            读下一个字符;
        }
        token 与关键字表进行比较，确定 syn 的值;
    }
    else
        if(ch 是数字)
        {
            .....
            syn=11;
        }
    else
        swith(ch)//其他字符情况
        {
            case '<':
                .....
            case '>':
                .....
            .....
            Default:syn=-1;
        }
}
```

## (4) 调试程序，验证输出结果。

### 【思考题】

- (1) 在编程过程中遇到了哪些问题，你是如何解决的。
- (2) 源程序若存在注释，如何实现词法分析，在现有程序基础上进行扩充。

### 【参考文献】

1. 张素琴等，编译原理（第 3 版），清华大学出版社，2012.03
2. 胡伦骏、徐兰芳等，编译原理（第 3 版），电子工业出版社， 2010.7