

# 对完整程序(多条执行语句)的递归下降分析

17计二许红凯 320170941570

## 任务介绍

递归下降的语法分析

## 输入

一个完整的源程序

## 输出

正确或错误

## 题目

设计一个程序，输入字符串形式的源程序，输出该程序是否符合相关语法，有错误时报告错误

要求：

- 1.源语言及其语法规则：可以参照附录A，也可以自定义
- 2.输入为字符串形式的源程序，因此，需要调用前面实验做过的词法分析器，为语法分析器提供单词符号
- 3.应该指出错误的具体位置，如：在xx单词之后/之前发现错误，分析中止

## 代码思路

1. 首先进行词法分析，在识别去掉空格、换行和tab后，识别不为关键字的变量，数字，算数运算符，关系运算符，各种界符等
2. 对于不同模块，定义不同错误类型。例如分块程序缺少花括号 {}，语句缺少分号；，定义变量时缺少类型名，以及在选择语句、循环语句、判断语句中对应可能出现的错误类型
3. 在对具体的语句比如循环语句进行正误判断时，调用已定义过错误类型的基础语句，提高效率

## 程序功能说明

根据提示输入待检测文件名（xx.txt），根据定义的语义规则判断文件内容是否符合规则。若符合则输出“语法检查完成，共发现0个错误”；若不符合，则定位到每个错误所在的行号，并输出行号与错误类型，最后输出总共错误数。

## 重点说明

1. 定义了一个单词类Word，其中对于单词类别type进行设定，以便于词法分析时对当前符号类型进行判断，多个当前符号的类型可进一步判断出当前语句是什么类型，对应会出现什么错误，是否有错误匹配等

```
class Word//单词表类
{
    public:
    char value[20];//单词自身的值

    //单词的种别（10标识符，20常数，30关键字，4-运算符(加法运算符为41，乘法为42，关系运算符为43)
    int type;
    int line;//行号
}
```

2. 函数P（）作为递归入口，P->A->B->...

```
void P()//程序
{
    getWord();//(
    A();
}

int A() //{分程序}
{
    if(strcmp(new_w.value,"{"))
    { error("缺少'{'");}
    getWord();//int
    B();
    if(strcmp(new_w.value,"} "))
    { error("缺少'}'");}
    cout<<"语法检查完成,共发现"<<e<<"个错误."<<endl;
    return 0;
}
```

3. 函数H（）中，已识别当前符号为标识符，进一步识别该标识符可判断是赋值语句Or条件语句Or循环语句，从而进入不同类型语句对应的错误判断函数中去

```
int H()//语句
{
```

```

if(new_w.type==10)//如果当前符号为标识符，则为赋值语句
{
    f=0; //运算符的个数
    flag2=0;
    ise(new_w.value);//判断标识符是否定义
    getWord();
    a[a_num++]=new_w.value;//=号
    I();//赋值语句
    return 0;
}
else if(strcmp(new_w.value,"if")==0)//判断是否为条件语句
{
    flag2=0;
    getWord();
    J();//条件语句
    return 0;
}
else if(strcmp(new_w.value,"while")==0)//判断为循环语句
{
    flag2=1;
    getWord();
    K();//循环语句
    return 0;
}
else if(strcmp(new_w.value,";")==0)
{
    return 0;
}
return 0;
}

```

## 代码测试

### 1. 错误示例

```

{
int i,j,k;
i=1;j=9;
while(j>0)
{
k=i*j;
i=i+1;
j=j-1;
}
s=k;      //未声明s

if(s=25)  //未声明s

{
s=0;      //未声明s

```

```

输入要测试的源程序名字 (xx.txt) :error.txt
第 10 行出错!   s 没有被声明 !
第 11 行出错!   s 没有被声明 !
第 13 行出错!   s 没有被声明 !
语法检查完成,共发现3个错误。

```

```
}  
}
```

## 2. 正确示例

```
输入要测试的源程序名字 (xx.txt) :right.txt  
语法检查完成,共发现0个错误。
```

```
`{  
  int i,j,k,s;  
  i=1;j=9;  
  while(j>0)  
  {  
    k=i*j;  
    i=i+1;  
    j=j-1;  
  }  
  s=k;  
  if(s=25)  
  {  
    s=0;  
  }  
}`
```