

# C 语言程序词法分析（C++实现）测试报告

姓名：熊宇

班级：2018211302

学号：2018210074

指导教师：李文生

## 目录

输入 .....	2
运行结果 .....	3
分析说明 .....	6

## 一、输入

```
/*
    Version:1.0
    User:Septer
    Last Updated Time:2020/10/06 16:42
    Todo:Test
*/

#define relax 10

#include <stdio.h>
#include <math.h>

int main()
{
    int m,flag,i;
    scanf("%d", &m);
    flag = 1;
    for(i = 2;i <= sqrt(m);i++)
    {
        if(m%i == 0)
        {
            flag = 0;
            break;
        }
    }
    if(flag)
    {
        //Now have a output.
        printf("%d is primers.\n", m);
    }
    else
    {
        //Now have a output.
        printf("%d is not primers.\n", m);
    }
    return 0;
}
```

## 二、运行结果

```
2 <Variable, Version>
2 <Number, 1.0>
3 <Variable, User>
3 <Variable, Septer>
4 <Variable, Last>
4 <Variable, Updated>
4 <Variable, Time>
4 <Number, 2020>
4 <Number, 10>
4 <Number, 06>
4 <Number, 16>
4 <Number, 42>
5 <Variable, Todo>
5 <Variable, Test>
8 <Variable, define>
8 <Variable, relax>
8 <Number, 10>
10 <Variable, include>
10 <Variable, stdio>
10 <Variable, h>
11 <Variable, include>
11 <Variable, math>
11 <Variable, h>
13 <Keyword, int>
13 <Variable, main>
13 <(<, ->
13 <(<), ->
14 <{, ->
15 <Keyword, int>
15 <Variable, m>
15 <Variable, flag>
15 <Variable, i>
16 <Variable, scanf>
16 <(<, ->
16 <string, %d>
16 <Variable, m>
16 <(<), ->
17 <Variable, flag>
17 <=<, ->
17 <Number, 1>
18 <Keyword, for>
18 <(<, ->
18 <Variable, i>
18 <=<, ->
18 <Number, 2>
18 <Variable, i>
18 <=<, ->
```

```

18 <Variable, sqrt>
18 <(, ->
18 <Variable, m>
18 <), ->
18 <Variable, i>
18 <++, ->
18 <-, ->
18 <), ->
19 <{, ->
20 <KeyWord, if>
20 <(, ->
20 <Variable, m>
20 <Variable, i>
20 <==, ->
20 <Number, 0>
20 <), ->
21 <{, ->
22 <Variable, flag>
22 <=, ->
22 <Number, 0>
23 <Variable, break>
24 <}, ->
25 <}, ->
26 <KeyWord, if>
26 <(, ->
26 <Variable, flag>
26 <), ->
27 <{, ->
28 <Variable, Now>
28 <Variable, have>
28 <Variable, a>
28 <Variable, output>
29 <Variable, printf>
29 <(, ->
29 <string, %d is primers.\n>
29 <Variable, m>
29 <), ->
30 <}, ->
31 <KeyWord, else>
32 <{, ->
33 <Variable, Now>
33 <Variable, have>
33 <Variable, a>
33 <Variable, output>
34 <Variable, printf>
34 <(, ->
34 <string, %d is not primers.\n>
34 <Variable, m>
34 <), ->

```

```
35 <}, ->  
36 <KeyWord, return>  
36 <Number, 0>  
37 <}, ->
```

行数: 37

字符数目: 52

关键字数目: 7

变量名数目 45

数字常量数目: 12

分界符和操作符数目: 36

Press any key to continue . . .

### 三、分析说明

该 C 语言词法分析程序

- ①可以识别出用 C 语言编写的源程序中的每个单词符号，并以记号的形式输出每个单词符号；
- ②可以识别并跳过源程序中的注释；
- ③可以统计源程序中的语句行数、各类单词个数、以及字符总数，并输出统计结果；
- ④检查源程序中存在的词法错误，并报告错误所在的位置；
- ⑤对源程序中出现的错误进行适当的恢复，使词法分析可以继续进行，对源程序进行一次扫描，即可检查并报告源程序中存在的所有词法错误。