

原 黑马程序员——C语言笔记之预处理指令

2015年09月12日 15:38:22 meimei6\_6 阅读数：274

-----<a href="http://www.itheima.com" target="blank">Java培训、Android培训、iOS培训、.Net培训</a>、期待与您交流！

预处理指令：

预处理指令就是指在编译器进行编译的第一遍扫描（词法扫描和语法分析）之前所作的工作。当队一个源文件进行编译时，系统将自动引用预处理程序对源程序作处理，处理完毕自动进入对源程序的编译。C语言中提供了很多种与处理功能，如宏定义、文件包含、条件编译等。合理使用与处理功能编译写的程序便于阅读、修改、利于模块化程序设计。

一.宏

- 1.C语言中自定义的特殊标志符，习惯大写
- 2.宏定义：# define 宏名 宏字符串（常量、变量、表达式）
- 3.宏替换：源程序在编译之前，由于处理程序对我们写的源代码进行处理，会把源代码中所有出现的宏名的地方一律使用宏的字符串去 替换。

```
//例：
#include <stdio.h>
#define M 10 //宏定义
int main(int argc, const char * argv[]) {
    int a[M+2];
    //打印M的值
    printf("M=%d",M);

    return 0;
}
```

M=10

4.宏使用的注意事项：

- 1) 预处理指令，经常写在函数之前
- 2) 宏是有作用域的 # undef宏名，可以取消宏定义
- 3) 在字符串中出现的宏名不会被替换
- 4) 宏可以嵌套定义：

```
例：#include<stdio.h>

#define R 4

#define PI 3.14

#define AREA PI* R* R

嵌套定义
```

5.无参宏

```
例：#define M 10
```

6.有参宏

```
//例：有参宏的使用
#include <stdio.h>
#define M1(x,y)x*y+x+y //有参宏定义
int main(int argc, const char * argv[]) {
    int result= 0;
    //有参宏的使用
    result=M1(4, 5);
    printf("result=%d",result);
    return 0;
}
```

result=29

## 7. #define和typedef区别

宏定义是在预处理完成的，而typedef是在编译时处理的，它不是简单的替换，而是对类型说明符重新命名，被命名的标识符具有类型定义说明的功能。

### 二.条件编译

#### 1. #if - #else条件 编译指令

格式：#if 常量表达式

程序段1

#else

程序段2

#endif

作用：如果常量表达式的值为真，则对程序段1进行编译，否则对程序段2进行编译。

#### 2. #ifdef - #else条件 编译指令

格式：#ifdef 标识符

程序段1

#else

程序段2

#endif

作用：如果标识符已被 #define 命令定义过则对程序段1进行编译，否则对程序段2进行编译。如果没有程序段2（它为空），本格式中的 #else 可以没有，即可以写为：

#ifdef 标识符

程序段

#endif

#### 3. #ifndef - #else条件 编译指令

格式：#ifndef 标识符

程序段1

#else

程序段2

#endif

作用：如果标识符未被 # define命令定义过则对程序段1进行编译，否则对程序段2进行编译。

```
#include <stdio.h>
#define DEBUG1 1
int main(int argc, const char * argv[]) {
    int a= 0;
    //检测宏是否定义
#ifdef DEBUG1
    a=10;
#else
    a=100;
#endif
    printf("%d\n",a);
    return 0;
}
```

10

### 三.static和extern修饰局部变量

1.1 ) static定义的变量作用域会得到延长

2 ) static定义的变量的语句只会被执行一次

#### 2.extern是不能修饰局部变量的

3.static对全局变量的作用（内部变量）

static修饰的全局变量，表示该全局变量只能在当前文件中使用，而不能在其他文件中使用

在不同的文件中使用static可以声明同名的变量

4. extern对全局变量的作用

extern修饰的全局变量，表示当前变量可以在本文件中使用，也可以在其他文件中使用

注意：extern声明的全局变量在不同的文件中不能同名（两个文件需包含关系）

```
// static修饰局部变量
#include <stdio.h>
void sum(){

    static int num =0;
    printf("num=%d\n",num);
    num++;

}
int main(int argc, const char * argv[]) {

    sum();
    sum();
    sum();
    printf("num=%d\n",sum);
    //num是随着调用的次数，不断增加的
    return 0;

}
```

num=0  
num=1  
num=2

```
/**
 *思考与实现
 *使用有参宏实现判断两个数的大小
 */
#include <stdio.h>
//定义一个有参宏，求两个数的最大值
#define Max(a,b)a>b?a:b
int main(int argc, const char * argv[]) {
    //调用有参宏
    int m= Max(34,88);
    printf("%d\n",m);
    return 0;
}
```

88



想对作者说点什么



外贸找客户的几种方法