第八章 C语言预处理

8.1 预处理概述

1、前面各章中，已经多次使用过#**include**命令。使用库函数之前，应该用#include引入对应的头文件。这种以 # 号开头的命令称为**预处理命令。**

2、C语言提供了多种预处理功能，如**宏定义、文件包含、条件编译**等。合理地使用预处理功能编写的程序便于阅读、修改、移植和调试，也有利于模块化程序设计

8.2 宏定义

1、在C语言源程序中允许用**一个标识符**来表示**一个字符串**，称为“宏”。被定义为“宏”的标识符称为“宏名”

2、在编译预处理时对程序中所有出现的“宏名”，都用宏定义中的字符串去进行**文本替换**，，这称为“宏代换”或“宏展开”

8.2.1 无参宏定义

1、无参宏的宏名后不带参数，一般形式为：

```
#define 标识符 字符串
```

注意1：“字符串”可以是常量、表达式

2、举例

```
#define  PI  3.14
#define EXP x*10*1.5
```

编译器在预处理时，会将源程序中所有的“PI”替换成“3.14”，将所有的EXP，替换成表达式“x*10*1.5，下面我们来验证一下：

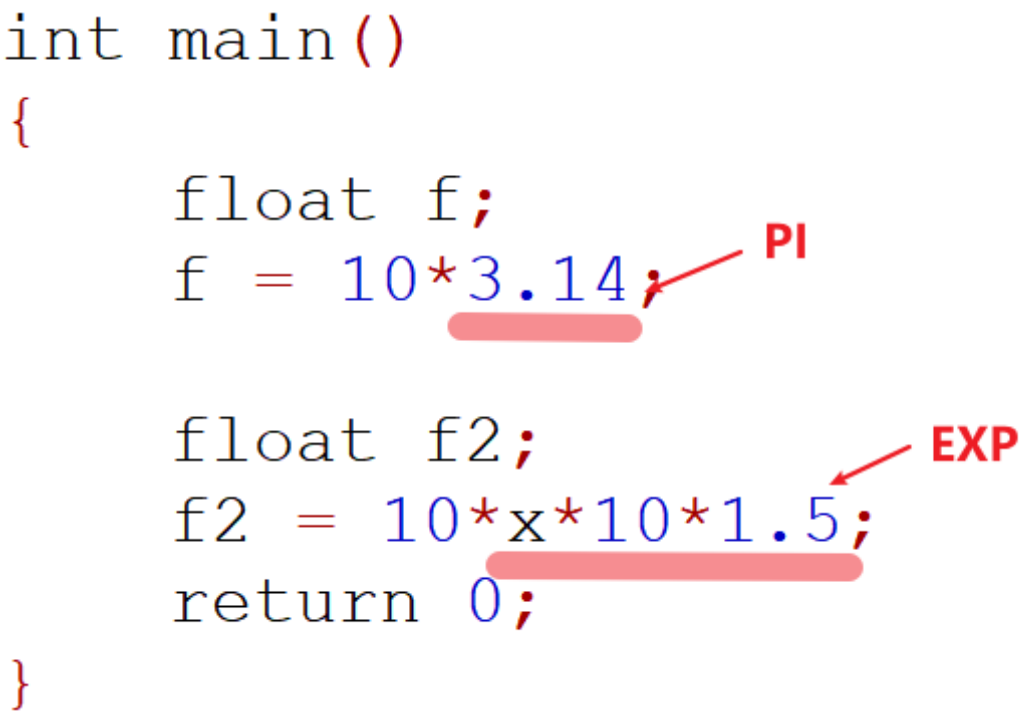
```
#include <stdio.h>

#define  PI  3.14
#define EXP x*10*1.5

int main()
{
    float f;
    f = 10*PI;
```

```
    float f2;
    f2 = 10*EXP;
    return 0;
}
```

我们可以在CLion的“Terminal”下输入编译命令：**gcc -E main.c -o main.i**， 通过预处理将main.c 编译成main.i 文件，下面我们可以打开main.i文件，看看里边的内容。



3、我们发现在刚才的代码中，宏定义中出现了x，但是我们在使用时却没有定义x变量，预处理还是能够顺利通过，这是为什么呢?

> 宏定义是简单的文本替换，不会做语法检查！！！

8.2.2 有参宏定义

1、 C语言允许宏带有参数。在宏定义中的参数称为形式参数，在宏调用中的参数称为实际参数。

2、对带参数的宏，在调用中，不仅要宏展开，而且要用实参去代换形参。带参宏定义的一般形式为:

```
#define 宏名(形参表) 字符串
```

3、使用方法

```
#define F(x)  x*x*x
int main()
{
    int r = F(3);
    return 0;
}
```

8.2.3 宏定义注意事项

1、宏定义的“字符串”如果是表达式，表达式中的变量需要用 () 包含起来，因为在预处理阶段展开宏时，可能会用表达式来替换宏定义中的变量

```
#define F(x) (x)*(x)*(x)

int main()
{
    F(1+2);
    return 0;
}
```

上面的例子中展开后，(1+2)*(1+2)*(1+2)，如果不加（）呢？展开的表达式就变成这样了1+2*1+2*1+2*1+2,这显然不是我们想要的结果。

2、经典笔试题：使用宏定义返回两个整数中的最大值

```
#define Max(a, b) (a)>(b)?(a):(b)
```

8.3 文件包含

1、在前面的学习中我们已经使用了C语言得文件包含命令，例如：#include <stdio.h>，文件包含语法格式如下：

```
#include <文件名>
#include "文件名"
```

注意：

- 使用尖括号表示在开发环境的头文件目录下查找，而不在源文件目录去查找
- 使用双引号则表示首先在当前的源文件目录中查找，如果没有找到才到包含目录中去查找
- “文件名”可以是相对路径， 也可以是绝对路径，建议不要使用绝对路径

8.4 条件编译

1、常用条件编译指令

条件编译指令	说 明
#if	如果条件为真，则执行相应操作
#elif	如果前面条件为假，而该条件为真，则执行相应操作
#else	如果前面条件均为假，则执行相应操作
#endif	结束相应的条件编译指令
#ifdef	如果该宏已定义，则执行相应操作
#ifndef	如果该宏没有定义，则执行相应操作

2、ifdef

```
#ifdef 标识符
    程序段 1
#else
    程序段 2
#endif
```

它的功能是，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、#if

经常使用#if 0 对代码进行注释

```
#if 0
    代码块
#endif
```