

# 第十四讲

## 预 处 理

## ■ 14.1 预处理的工作原理

**#define** 宏 代表其他东西

```
#define PI 3.14159
```

**#include** 包含 将其他文件内容包含进来

```
#include <stdio.h>
```

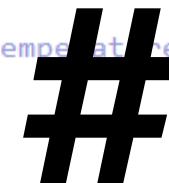
C program

Preprocessor

Modified C program

Compiler

Object code



```
1 /* Converts a Fahrenheit temperature to Celsius */
2
3 #include <stdio.h>
4
5 #define FREEZING_PT 32.0f
6 #define SCALE_FACTOR (5.0f / 9.0f)
7
8 int main(void)
9 {
10     float fahrenheit, celsius;
11
12     printf("Enter Fahrenheit temperature: ");
13     scanf("%f", &fahrenheit);
14
15     celsius = (fahrenheit - FREEZING_PT) * SCALE_FACTOR;
16     printf("Celsius equivalent is: %.1f\n", celsius);
17
18     return 0;
19 }
```

```
1 Blank line
2 Blank line
3 Lines brought in from stdio.h
4 Blank line
5 Blank line
6 Blank line
7 Blank line
8 int main(void)
9 {
10     float fahrenheit, celsius;
11
12     printf("Enter Fahrenheit temperature: ");
13     scanf("%f", &fahrenheit);
14
15     celsius = (fahrenheit - 32.0f) * (5.0f / 9.0f);
16
17     printf("Celsius equivalent is: %.1f\n", celsius);
18
19     return 0;
20 }
```

## ■ 14.3 宏定义

**#define** 标识符 替换列表

**#define N 100**

**#define N=100**    **#defXe N 100;**

参数

**#define** 标识符(x1 , x2 , ... , xn) 替换列表

**#define MAX(x,y) ((x)>(y)?(x):(y)) 括号?**    i = MAX(j+k, m-n);

**#define IS\_EVEN(n) ((n)%2==0)**    **#define PRINT\_INT(n) printf("%d\n", n)**

**#define TOUPPER(c) ('a'<=(c)&&(c)<='z'? (c)-'a'+'A':(c))**

## ■ 14.3 宏定义

#运算符

将参数转换为字符串

```
#define PRINT_INT(n) printf(#n " = %d\n", n)
```

```
PRINT_INT(i/j);      printf("i/j" " = %d\n", i/j);
```

```
printf("i/j = %d\n", i/j);
```

## ■ 14.3 宏定义

**##运算符** 将两个记号粘结在一起

```
#define MK_ID(n) i##n
```

```
int MK_ID(1), MK_ID(2), MK_ID(3);
```

```
int i1, i2, i3;
```

## ■ 14.3 宏定义

预定义宏

提供当前编译或编译器信息

名字	描述
_LINE_	当前程序行的行号,十进制整数常量
_FILE_	当前源文件名,字符串面量
_DATE_	编译日期(mm dd yyyy), 字符串面量
_TIME_	编译时间(hh:mm:ss), 字符串面量
_STDC_	如果编译器符号C标准,则值为1
_STDC_VERSION_	支持的C标准版本

## ■ 14.4 条件编译

**#if** 和 **#endif**

提供当前编译或编译器信息

```
#define DEBUG 1
```

```
#if DEBUG
printf("Value of i: %d\n", i);
printf("Value of j: %d\n", j);
#endif
```

```
#define DEBUG
```

```
#ifdef DEBUG
printf("Value of i: %d\n", i);
printf("Value of j: %d\n", j);
#endif
```

```
#if defined(WIN32)
```

...

```
#elif defined(MAC_OS)
```

...

```
#elif defined(LINUX)
```

...

```
#endif
```