PC C 语言教程

李文兵 编著

天津科学技术出版社

A COURSE OF PC C

PC C 语言教程

李文兵 编著

天津科学技术出版社

内 容 提 要

本书分12章系统地介绍了ANSI标准C语言的词法、语法规则，以及使用技巧。两个附录分别给出了行编辑的用法和C86的错误信息。

作者在内容组织上注重C语言的系统性；在写法上注重深入浅出，通俗易懂；在练习举例上注重实用性。全书共有284个上机调试通过的程序，给出了解决各种问题的钥匙。

与本书配套的有系统软件、练习程序软盘、教学录像片。该书适于作大专院校、电职大、高等教育自学考试、软件水平考试、计算机中专和有关培训班教材，也可作为有关工程技术人员和管理人员的参考书和自学用书。

津新登字(90)003号

PC C语言教程

李文兵 编著

责任编辑：刘万年

*

天津科学技术出版社出版

天津市张自忠路189号 邮编300020

天津市武清县振兴印刷厂印刷

新华书店天津发行所发行

*

开本787×1092毫米 1/16 印张27.25 字数668 000

1994年4月第1版

1994年4月第1次印刷

印数：1—6 000

ISBN 7-5308-1481-8

TP·46 定价：16.00元

前　言

《PC　C 语言教程》与读者见面了。

此时此刻，作者最想说的一句话，就是感谢广大的读者、有关专家，以及出版社的同志们，是他们给了我信心和力量。应该说，没有广大读者、有关专家的鼓励和指点，没有出版社的大力帮助与支持，这本书以及作者的其他著作的出版就不会这么一路顺风。

本书的写作基础是作者近十年从事C语言教学与开发C语言、应用C语言的实践。到写前言时，作者已讲授C语言近千学时，并用C语言开发过各种软件。

本书的框架源于《IBM　PC　C语言简明教程》。《IBM　PC　C语言简明教程》出版以来，受到广大读者，特别是大专院校师生的欢迎，被许多院校及培训班、研修班选作教材，一直供不应求。为适应计算机事业的发展，进一步加强C语言的教学需要，作者沿用前书的思路与框架，根据近几年的发展与实践，写了这本《PC　C语言教程》。该书突出如下几个问题：

- 全书是根据ANSI标准而写的；考虑到应用上的需要，又没有拘泥于标准。
- 增加了国内外非常流行的Turbo　c、Microsoft　c编译系统的上机操作说明，且书中的程序都能在这三种编译系统下编译通过。
- 对读者感到困惑的问题，如表达式的副作用及序列点、递归调用、函数调用过程中的参数传递等问题，做了较为详尽的解答。
- 对指针和结构做了更加深入的探讨，给出了更加复杂的指针结构和结构数据。
- 增加了文件管理和硬件资源管理两章，给出了应用C语言的钥匙。

该书来自课堂，来自实践；内容系统且丰富；加上本书深入浅出、通俗易懂的风格，作者相信一定会受到广大读者的欢迎。

在该书的编著过程中，王玉华、崔竹、韩建枫、彭群力、于忠民、陈大红等同志，及时帮助调试了程序。贾秀荣同志绘制了书中全部图。对他们的真诚合作，在此表示衷心的感谢。

在该书的出版过程中，作者还得到了科海培训中心华根娣主任、中国软件行业协会考试指导中心的领导的无私帮助与大力支持，这里也向他们表示感谢。

由于时间仓促，水平有限，错误和不妥之处在所难免，诚恳希望广大读者和有关专家指正。

李文兵

1993.5.4 于天津

作者简介

1967年毕业于清华大学自动化专业。

现任天津师范大学计算机系副教授。主要从事计算机系统结构与系统软件的研究与教学以及过程控制、应用软件的开发工作。

作者是国内最早致力于C语言介绍、研究、开发的计算机工作者，是国内研究C语言的先行者之一。主要著作有：《IBM PC C语言简明教程》、《IBM PC C语言例题习题库函数》、《C语言学习指导》、《Turbo c 及其应用》、《PC C语言教程》。

Brief introduction
to the author

Dr. liwenbing received diploma in Automation from Qinghua university in 1967.

He is currently an associate professor in the department of computer at Tianjin normal university. He is mainly engaged in the research and teaching of computer system organization and system software, in the development of process control and applicational software.

He is among the first computer man in P. R. C to introduce, study, develop C language. His most important works include: 'A concise course of IBM PC C', 'Examples, Exercises, and library function of IBM PC C', 'Learner's guide to C', 'Turbo c and its applications', 'A course of PC C'.

目　录

1章 绪 论

§1.1 C 语 言

1. C 语言产生的历史背景

C 语言是 70 年代贝尔(Bell)实验室为描述 UNIX 操作系统和 C 编译程序，而开发的一种系统描述语言。

1969 年，美国贝尔实验室的两个研究人员 Ken Thompson 和 Dennis M. Ritchie，开始合作研究 UNIX。他们用了不到一年的时间，就在 PDP-11 机上，用汇编语言研制成功了。

1970 年，Ken Thompson 为了提高 UNIX 的可读性和可移植性，在 BCPL 语言的基础上，开发了一种新的语言，起名叫“B”。之所以叫这个名字，据说其中一个因素就是根据 BCPL 的字头。由于 B 语言存在着一些缺点，例如，它没有定义数据类型，无法支持多种数据类型，致使它并未流行起来。

1971 年，Dennis M. Ritchie 用了一年左右的时间，在 B 的基础上开发了第一个 C 编译程序。因为这个编译程序是在 B 语言的基础上开发的，无论是在英文字母序列中也好，还是在 BCPL 这个名字中也好，排在 B 后面的均为 C，故起名为“C”语言。

1972 年，这个 C 编译程序投入使用。

1973 年，Ken Thompson 和 Dennis M. Ritchie 两个人合作，把 UNIX 全部用 C 语言重写了一遍，这就为 UNIX 的移植和发展奠定了基础。之后，C 语言又经过多次改进，主要在贝尔实验室内部使用，从未公开。

1975 年，UNIX 的第 6 版本公诸于世后，C 语言便被举世瞩目。C 语言凭着它本身的优点，很快就得到了广泛的应用。这期间越来越多的程序编制人员放弃了已运用自如的 FORTRAN 语言、PL/1 语言、甚至刚刚掌握的 PASCAL 语言或是 APL 语言，而偏爱上了 C 语言。

1978 年，发表了 UNIX 第 7 版本。就是根据这个版本 Brian W. Kernighan 和 Dennis M. Ritchie 合著了一本名为《The c programming language》的书。这本书在国际上流行很广。以致人们把它与 UNIX 第7版本视为一体，而称之为标准版本。在日本，石田晴久先生把它译成日文出版。在我国，史美林、孙玉方和孟庆昌等专家，首先把它译为中文，为推广、普及 C 语言做出了贡献。

1978年以后，C 语言便被移植到小至8位微机、大至巨型机象 Cray1 等大多数计算机上。目前，C 语言已独立于 UNIX 和 PDP-11，运行在各种操作系统上和各种机种上。特别是在16位至32位微机上使用非常普遍。

2.C 语言的特点

(1)移植性好　C 语言作为一种标准语言而得到广泛应用,原因就在于 C 语言能被移植到多种操作系统上。

C 语言之所以移植性好,主要在于其处理系统很紧凑。这是因为作为语言,它没有定义输入输出,而且关键字用得少。在语法上仅仅定义了最基本的控制语句;而输入、输出和字符处理等,是作为外部函数,链接到所编写的程序上才使用的。

虽然 C 语言能够使用的环境是多种多样的,但是用 C 语言所编写的程序在这些环境之间进行移植时,还需要注意两个问题:

①各变量类型及其字长问题　各变量类型的字长因所用的 CPU 的字长而异,故移植程序时必须注意。

②与操作系统接口有关的库函数问题　把 C 语言从一种操作系统下,移植到另一种操作系统下,必须修改有关的库函数。因为这些库函数是在特定环境下定义的,环境一改变,它就不适用了。因此,ANSI 规定了 C 编译系统所必须提供的标准库函数集合,其目的就是为增强 C 程序的可移植性。

(2)应用性广　C 语言应用性很广,用 C 语言编写的源程序中最大且最有名气的要算作 UNIX 了。除此之外,还有 CP/M68k、Multiplan 这样的 OS 及其软件。

在这样大的软件里之所以使用 C 语言,主要有两个原因,即 C 语言移植性好和硬件控制能力强。如果应用程序用 C 来描述,则由于 C 移植性好,应用程序的移植性也就好。此外,由于 C 的类型检验比较弱,能直接访问物理地址等特点,使得它在直接操作硬件功能时,能实现同汇编语言相同的效果。这就是过去用汇编语言所编写的操作系统可以使用 C 语言的原因。

由此看来,实际上 C 所能迅速应用到的领域正是以前汇编语言所应用的领域,因此,C 被称作是"便携汇编语言"。它是介于从 Ada、Smalltalk 到 FORTRAN 这样的高级语言和汇编语言之间的一种语言。

事实上,由于 C 语言绘图能力很强,能处理汉字,能直接管理硬件,能连接 BIOS、OS、其它高级语言,能与汇编语言混合编程,故用它来设计管理信息系统,方法会更灵活,功能会更强,效果会更加完美。

因此,我们可以说,C 语言既具有高级语言的特点,又具有汇编语言的特点;它既是成功的系统描述语言,又是一个实用的程序设计语言。这就是所谓的 C 语言的双重性。

(3)程序设计自由度大　C 语言程序设计在语言自身的规格上能够相当自由地进行,故说 C 语言是水平较低的语言,也可以说是较为接近汇编语言的一种语言。

由于 C 语言是水平较低的语言,故对变量类型、变量范围和存贮空间的存取的制约就少,这样,程序设计就能很自由地进行。

对于程序员来说,若使用高级语言,只要按照语法规则来编写,就能够比较简单地编写出正确的程序,并能自动地进行错误检验。而 C 语言,由于其自由度较大,错误检验在各个阶段难以进行,要用它编写出正确的程序是较难的。

这意味着,同其它语言相比,C 语言具有面向专业的特征。

(4)表示方法简洁

①语句简洁　C 语言和 PASCAL 语言都是由 ALGOL 语言发展过来的,是一对孪生姐妹。但 C 语言要比 PASCAL 语言表达简洁,如表1.1所示。

表1.1　　C与pascal的比较

C语言	PASCAL语言
{ }	begin end
if(e)s;	if e then s
int i;	var i:integer
int f( );	function f( ):integer
int a[ ];	var a:array[…]of integer
int *p;	p:↑integer

②表达式能用赋值语句　如，

```
while((c=getchar( ))!=EOF)putchar(c);
```

这里，把getchar( )的返回值赋给c，这个值就是c=getchar( )的值。

③变量更新表示简单　在C语言中

```
i=i+1;
```

可以表示成：

```
i+=1;
```

这种表示方法也适用于其它运算符。

④变量的自增/自减表示简单

```
i++    i--      (1)
++i    --i      (2)
```

注意：(1)和(2)的相同点是运算后i都是自动+1/-1；(1)和(2)的区别是：++i/--i是先进行加/减1再使用，而i++/i--则是先使用再进行加/减1。

⑤有条件运算符(?:)　如，

```
(x!=0)?1/x:INFINITY
```

这一表示的含义是：若(x!=0)为真，则求解1/x，否则值为INFINITY的值。在其它语言里，要表达上述功能，必须使用if语句。

⑥关键字少　美国标准(ANSI)只定义了32个关键字，即：

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

注意，这些关键字都是用小写字母定义的，使用时，不能用大写字母书写；关键字不能挪作他用，不允许用它们定义变量名、函数名等。

⑦语句少　C语言只定义了如下10个语句：

if	do-while	switch	continue	goto
while	for	break	return	;(空语句)

§1.2 C 程 序

1.C 程序的格式

用 C 语言所编写的程序，称作 C 语言源程序，简称 C 程序(或源程序)。C 程序一般由一个或多个函数组成，这些函数可集中放在一个或分散放在几个文件中。为表示这些文件都是 C 程序文件，各文件都必须以.C 结尾，如 sum1.c 就是一个 C 程序文件，该文件只含有一个函数 main( )，如练习1.1所示。

【练习1.1】

```
A>type exp1_1.c
main( )  /*...add of a and b...*/
{
  int a,b,sum;
  a=123;b=456;
  sum=a+b;
  printf("sum is %d\n",sum);
}
A>
```

就该例，对 C 程序，说明如下：

(1)C 程序无行号，这里的行号是为了便于讲解而加的。

(2)sum1.c 是文件名。在这里，我们看到 C 程序文件的结尾是.C。.C 即表示 C 源文件。

(3)/*与*/之间的内容是注释。本例中，它说明 main( )函数是求解 a 和 b 两个自变量的值之和。注释不被编译，只起备忘和帮助阅读的作用。

(4)在 C 程序中，其后紧跟( )的标识符是函数名。我们已经看到，该程序只有一个函数，即函数 main( )。这里，main 是表示程序的主体部分，即是主程序的约定词。main()可叫主函数。

在执行 C 程序时，一调用函数名为 main( )的函数，程序便开始执行；而从 main( )返回时，程序执行便终止。

在一个完整的 C 程序中一定要有一个函数 main( )，不管这个函数放在什么地方，C 程序总是从它开始执行。函数 main( )可以调用其它函数；其它函数之间也可以有调用和被调用的关系；此外，一个函数还可以调用其本身，这叫做递归调用。

(5)花括号{ }相当于 ALGOL/PASCAL 语言中的关键字 begin 和 end，也相当于 PL/1 语言里的 do 和 end。它是函数体的界限符。

(6)3行为说明语句。说明变量 a、b 和 sum 的值为整型数(int)。该说明语句的具体任务是通知计算机以下事项：

①该程序使用了 a、b 和 sum 三个变量；

②这些变量的值是整数。

这个程序里出现的变量名a、b和sum的含义，正如图1.1所示，是给计算机内存单元起的名字，各单元内存放着各自的数据，分别为：123、456、579。请注意，所用到的这三个变量名所表示的三个内存单元未必是相邻的。之所以称作变量，是因为在这种状态下重新赋值后，例如：

```
a=234;b=567;sum=a+b;
```

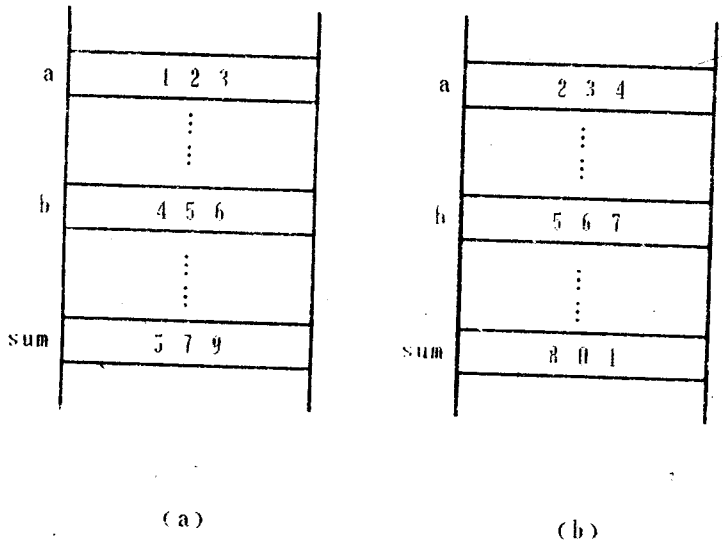

图 1.1　变量的含义

则a、b、sum的内容便成为如图1.1(b)所示的情况，发生了变化，故把它们叫做变量。

(7)4、5行中出现的=，是赋值运算符，表示把赋值运算符右边的数值或是运算结果赋值到其左边的变量所指定的存储单元中。人们习惯把赋值运算符左边的操作数叫左值；右边的叫右值。

(8)6行中的printf( )是按格式输出函数。它是编译系统中编好的函数。由于这类函数是编译系统中带来的，故俗称库函数。这里先介绍一下这个库函数，以备使用。

%d是printf( )函数的最简单的打印格式，习惯上，叫转换控制。它指出把后面的相应参数值(这里是sum的值)，以所需要的最少位数，按10进制打印，前后不留空格。

(9)语句后面要加；，表示语句结束。带有；号的语句叫简单语句，它是执行单位。在C程序中printf( )语句和赋值语句是很常用的语句。

在C程序里，不是每个简单语句必须占据一行。一行中可以有若干个语句，如练习1.1中第4行所示；但每个简单语句后，必须要有一个分号(;)。

有一点要特别注意，即花括号}前一个语句末的分号不能少；而在}后面又不能有分号。在这一点上，C语言和PASCAL语言不同。在PASCAL中end前一个语句不能有分号，有了则看作end和其前一个语句之间还有一个空语句。

(10)该程序的执行结果是：

```
A>exp1_1
sum is 579
```

这里必须强调的是，在C程序里提倡把一个程序分割为若干个函数来编写，又提倡把各个函数分别放在不同的文件里。这主要是考虑到一个有实用价值的程序往往需要反复进行修改、调试。如果把程序编制成一个大程序，尽管有时看起来还算紧凑、简洁，但只要修改一处，也要重新编译一次整个程序，这在时间上会造成很大的浪费。如果我们把一个程序分成若干个函数来编写，并把这些函数分别放在不同的文件里，那么我们修改某个函数，就只需重新编译一下这个函数所在的文件，而其它未被修改的函数所在的文件就无须重新编译，这样，编译时间会大大减少。待各个文件的编译都通过之后，通过函数调用，就可以把这些函数链接成一个完

整的程序来执行。

还需要指出的是，程序中函数彼此间的排列顺序如何是没有关系的，当然是尽可能整理成便于理解整个程序的结构顺序为好。此外，因为C语言不是PL/1和PASCAL那样的程序块结构语言，不能在函数定义中再定义函数。比如象下列形式的函数定义，即：

```
f(  )
{
  ……
  g(  )
  {
    ……
  }
  ……
}
```

就是错误的。从这个意义上可以说，C程序中的函数都是同一个级别的。

2. 最小C程序

在BASIC语言程序里，最小程序是仅有一个STOP语句的程序；而最小C程序，则如下所示：

```
mop(  ){  }
```

显然，mop( )是个函数名。这里没有用main( )这个名字，为的是使它更具有一般性。

在这个程序里，因为{ }内没有任何可执行的语句，所以该函数一旦被调用，什么也不做就立即返回到调用它的函数。虽然如此，这个函数还是很有价值的。在程序开发时，往往暂时先用这个形式写出来，为将来要设置的函数事先安排一个位置。

3. 函数的定义形式

上面所介绍的最小C程序，实际就是一个最小函数；同时我们已经知道C程序是由函数组成的。那么，一般函数的定义形式如何呢？这里介绍一下这个问题。

函数的定义，一般以如下形式给出：

```
函数属性说明  函数类型说明  函数名(参数表)
参数类型说明
{
  变量定义部分
  执行语句部分
}
```

函数定义中必不可少的部分是：

```
函数名(  )
{
}
```

其它部分根据需要来确定有无。

分析函数的定义形式，可以把函数定义分为两部分，即函数说明部分和函数体。

(1)函数说明部分　函数属性说明指出函数是内部函数，还是外部函数；分别用关键字

static 和 extern 表示。内部函数只能被该函数所在文件内的函数调用；而外部函数则可以由其它文件内的函数调用。没有属性说明的函数，按外部函数对待。

函数类型说明定义函数返回值的数据类型。C 程序所使用的基本数据类型有 int(整型)、char(字符型)、float(浮点型)。C 程序里没有类型说明的函数按整型对待。C 程序的函数相当于其他语言的子程序和过程，在 C 语言里称作函数。也有人把有返回值的叫函数，无返回值的叫过程。

函数名是识别函数的名字。有效的名字是以英文字母(a～z，A～Z)或下线符(_)开始的英文字母、数字和下线符的序列。

字母和数字的序列叫标识符。标识符可任意长。ANSI 规定，外部名必须至少能由前6个字符唯一区分。所谓外部名是指链接过程中所涉及到的标识符，其中就包括函数名，以及后面将要提到的全局变量名。例如，下面的外部名将被视为同一个标识符。

abcdefg　abcdefgh　abcdefghi

内部名必须至少能由前31个字符唯一地区分。内部名指的是仅出现于定义该标识符的文件中的那些标识符。

此外，在 C 程序里，函数名用大写字母和小写字母命名，被当成是两个函数。

参数表在函数名后的圆括号(　)内。这里的参数是形式参数，简称形参(亦叫哑元)。当该函数被调用时，形参要被实在参数(简称实参，亦叫实元)所替换。函数没有形参时，也不能省略掉圆括号(　)；因为圆括号是函数的特征符，只有后接(　)的标识符才被看作是函数名。前面所遇到的函数 main(　)就是没有形参的函数。

参数类型说明指定参数的数据类型。

有些专家把 C 语言分为传统的和现代的两种。象上述的参数类型说明方式就被说成是传统风格。而在现代风格中，参数类型说明可以放在函数名后面的(　)中。两种风格对比情况如下所示：

传 统 风 格	现 代 风 格
int max(a,b) int a,b; { … }	int max(int a,int b) { … }

ANSI 允许在函数定义前先给出函数原型，即预先定义函数的参数类型及函数返回值的类型，这样可以对参数及函数返回值强行进行类型检验。例如，下面的函数 fun(　)就是被原型化了的。

```
float fun(int,float);
main(  )
{
      ⋮
}
float fun(a,b)
int a;
float b;
```

{

⋮

}

(2)函数体　是进行预定处理的部分。

函数体从花括号{开始,直到与此对应的花括号}为止。

变量说明通常接在{后面。变量也是用标识符表示的,其基本类型也是 int、float 和 char。

接在变量说明后的是执行语句部分,这部分是实际生成命令码的地方。最后到函数体末是花括号},表示函数体终了。

4. C 程序的显著特点

通过前面的学习,我们已经了解到了 C 程序的一些风格和特点。这里,再强调一下。C 程序有如下四个显著特点。

(1)程序结构的模块化　C 程序是由文件组成的。文件是 C 程序的编译单位。就是说,C 程序在编译时,一次只能编译一个文件,不能是两个或两个以上文件,也不能是文件的一部分。因此,通常,把一个 C 源文件叫做一个程序单元。

一个文件是由一个或多个函数组成的。函数是 C 程序的基本单位。一个函数,一般,用来实现某种功能。

含有主函数 main(　)的文件,我们叫它主文件。

一个函数(除最小函数外)是由若干条语句组成的。语句是 C 程序的最小单位。

可见,组成 C 程序的成分,从大到小排列顺序是:程序、文件、函数、语句。而 C 程序结构的模块化主要体现在它是由其基本模块——函数组成上。

(2)书写的小写化　C 语言所定义的关键字、转换控制字符、控制语句,以及所有库函数用的都是小写字母,我们在使用时,也必须用小写。同样,我们在编写程序时,也提倡用小写。这主要是为了照顾人们的习惯。

(3)语句的表达式化　出现在 C 程序中的任何合理的数学表达式,均可被 C 编译系统编译执行。这说明 C 语言表达能力极强,适用于解决任何领域的问题。也正因为 C 程序可含有大量表达式语句,故有些专家称 C 语言为表达式语言。

(4)输入输出的库函数化　C 语言没有定义 I/O 语句,其程序是靠有关库函数来实现数据的输入输出功能的。

5. C 程序的执行过程

一个 C 程序的执行要经过编辑、编译、链接、执行四个步骤。而编译和链接过程如图 1.2 所示。

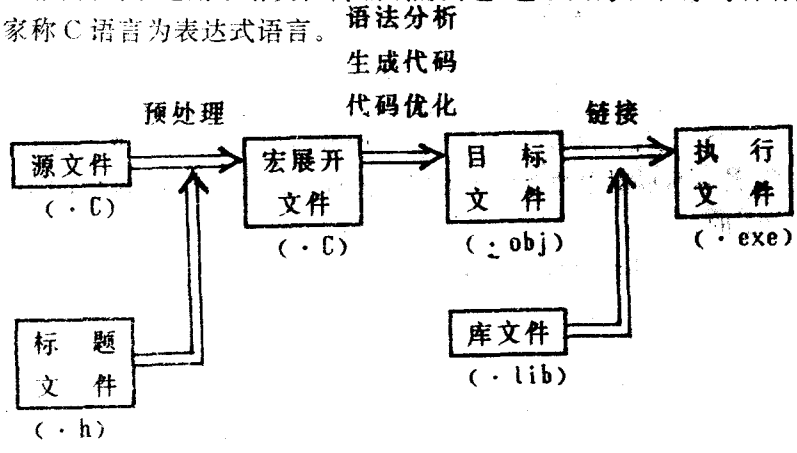

图 1.2　C 程序编译及链接过程

C 编译系统中含有预处理程序，用来处理源文件中带有＃的所谓控制行（称为宏指令）；这是编译过程中的第一步。为便于使用，这里，先介绍两条宏指令。

①宏定义：

```
#define  宏名  宏体
```

该宏指令通知预处理程序，把源程序中出现的该宏名都要用该宏体置换掉。例如，若源程序中有如下宏指令：

```
#define  INFY  -1
```

则该源程序经预处理程序处理后，源程序中的所有宏名 INFY，都将被宏体－1所置换。

②宏包含：

```
#include  "文件名"
```

该宏指令指示预处理程序，把文件名所指定的文件内容替换掉该宏指令，也就是，把指定的文件内容嵌入到现行程序中。例如，若源程序中有如下宏指令：

```
#include  "stdio.h"
```

则该源程序经预处理程序处理后，文件 stdio.h 的全部内容便嵌入到该源文件中；其内容便可以被该源文件使用。