

高等院校应用型本科“十三五”规划教材·计算机类

C语言程序设计

C Programming Language

► 主编 李亮



华中科技大学出版社

<http://www.hustp.com>

高等院校应用型本科“十三五”规划教材·计算机类

C语言程序设计

C Programming Language

► 主 编 李 亮
► 副主编 何友鸣



华中科技大学出版社

<http://www.hustp.com>

中国·武汉

内 容 提 要

本书较为全面系统地介绍了C语言程序设计基础。主要包括:C语言概述和程序设计的初步知识、三种基本结构(顺序结构、选择结构、循环结构)、数组、函数、指针、结构体、共用体、枚举类型、文件等。

本书参照普通高等教育C语言程序设计课程教学大纲的基本要求编写,充分体现实用性的原则,知识叙述简明扼要、通俗易懂,内容安排由浅入深、循序渐进,同时注意突出重点、分散难点;灵活运用任务驱动、案例教学、启发式教学等多种教学方法,对C语言程序设计的语言知识和程序设计的方法过程进行了系统介绍。

本书可作为高等院校各专业、计算机水平考试教材,也可作为相关从业人员培训或自学用书。

图书在版编目(CIP)数据

C语言程序设计/李亮主编. —武汉:华中科技大学出版社,2015.5

高等院校应用型本科“十三五”规划教材. 计算机类

ISBN 978-7-5680-0894-5

I. ①C… II. ①李… III. ①C语言-程序设计-高等学校-教材 IV. ①TP312

中国版本图书馆CIP数据核字(2015)第116369号

C语言程序设计

李 亮 主 编

C Yuyan Chengxu Sheji

策划编辑:曾 光

责任编辑:史永霞

封面设计:龙文装帧

责任校对:刘 竣

责任监印:张正林

出版发行:华中科技大学出版社(中国·武汉)

武昌喻家山 邮编:430074 电话:(027)81321913

录 排:华中科技大学惠友文印中心

印 刷:武汉市籍缘印刷厂

开 本:787mm×1092mm 1/16

印 张:15.25

字 数:395千字

版 次:2016年1月第1版第1次印刷

定 价:38.00元



华中出版

本书若有印装质量问题,请向出版社营销中心调换
全国免费服务热线:400-6679-118 竭诚为您服务
版权所有 侵权必究

前言

PREFACE

按照现代素质教育的观念,结合信息化社会对高素质、应用型人才的培养要求,在总结了多年从事 C 语言教学经验的基础上,特编写本教材。

C 语言是国内外广泛流行的计算机高级程序设计语言,也是国内外各高校开设的重要基础课程之一。C 语言兼顾了多种高级语言的特点,并且具备某些低级语言的功能。C 语言功能丰富,其硬件控制能力和运算表达能力很强,目标代码效率高,运行速度快,且具有良好的可移植性。C 语言的突出优点,使得它既是一个非常成功的系统描述语言,适合于编写系统软件(如操作系统、编译软件等),又是一个相当有效的通用程序设计语言,适合于编写各种应用软件(如图形软件、控制软件等)。同时,C 语言也是学习 C++、Java 及其他高级语言的基础。

目前,C 语言不仅是计算机科学与技术专业的必修专业基础课,而且已成为非计算机专业的一般计算机应用人员学习计算机高级语言的首选语言。

C 语言程序设计是一门实践性很强的课程,该课程的学习有其自身的特点。学生在学习的时候只通过理论学习是学不好的,必须通过大量的编程训练,在实践中掌握语言的基础知识,培养程序设计的基本能力,并逐步理解和掌握程序设计的思想和方法。因此,C 语言程序设计课程应该在学生掌握基本理论知识的基础上,重点培养学生的实践编程能力。

目前,介绍 C 语言的教材很多,但在多年的教学实践中,我们发现很多教材只注重 C 语言本身的语法知识的阐述,而忽略了培养学生的实践编程能力,导致学生记住了一大堆语法知识,却编不出来一个像样的程序,不利于培养学生的程序设计能力和语言应用能力。我们编写本书的目的是让学生在加强基础理论知识学习的基础上,注重培养实际的编写程序的能力。本书编写的指导思想是以零程序设计为起点,使读者通过一门计算机高级语言程序设计的学习,全面掌握

计算机程序设计的基本知识与基本操作技能,并熟悉 C 语言编程的有关应用,为读者进一步学习和应用计算机知识和操作技能打下扎实的基础。本书在内容上重基础、强能力,在行文上深入浅出、通俗易懂。

考虑到国内当前情况,本书中的程序是用 Visual C++ 6.0 编译的,运行结果也是在 Visual C++ 6.0 环境下得到的。

由于作者水平有限,书中难免有疏漏之处,欢迎广大读者批评指正。

编 者

2015 年 2 月

目
录

CONTENTS

第1章 概述	(1)
1.1 C语言概述	(1)
1.2 C程序的结构	(2)
1.3 C程序的上机步骤	(5)
1.4 Microsoft Visual C++ 6.0 集成开发环境	(6)
1.5 程序设计基础	(10)
习题	(18)
第2章 数据类型、运算符与表达式	(20)
2.1 C语言的数据类型	(20)
2.2 常量与变量	(21)
2.3 运算符和表达式	(26)
2.4 数据类型转换	(35)
习题	(37)
第3章 顺序结构	(38)
3.1 C语言的基本语句	(38)
3.2 数据输入与输出	(41)
3.3 程序举例	(50)
习题	(53)
第4章 选择结构	(55)
4.1 if语句	(55)
4.2 switch语句	(64)
4.3 选择结构的嵌套	(67)
4.4 选择结构程序举例	(69)
习题	(74)
第5章 循环结构	(76)
5.1 循环语句	(76)
5.2 循环的嵌套	(81)

5.3	结束循环的语句	(82)
5.4	循环结构应用举例	(83)
	习题	(87)
第6章	数组	(89)
6.1	一维数组	(89)
6.2	二维数组	(93)
6.3	数组与循环计算	(98)
	习题	(110)
第7章	函数	(112)
7.1	函数的概念	(112)
7.2	定义函数的一般形式	(112)
7.3	函数的值和函数参数	(114)
7.4	函数的调用形式	(117)
7.5	函数的嵌套调用	(120)
7.6	函数的递归调用	(123)
7.7	数组作为函数参数	(125)
7.8	局部变量与全局变量	(128)
7.9	变量的存储类别	(131)
7.10	内部函数与外部函数	(135)
7.11	编译预处理	(136)
	习题	(142)
第8章	指针	(143)
8.1	指针与指针变量	(143)
8.2	指针与函数	(152)
8.3	指针与数组	(159)
8.4	指针与字符串	(167)
8.5	指针数组与命令行参数	(170)
8.6	程序举例	(175)
	习题	(177)
第9章	结构体、共用体与枚举类型	(180)
9.1	结构体类型的定义	(180)
9.2	结构体变量	(181)
9.3	结构体数组	(185)
9.4	结构体类型指针	(188)
9.5	结构体与函数	(192)
9.6	共用体类型	(196)
9.7	枚举类型	(200)
9.8	类型定义	(203)
	习题	(204)

第 10 章 文件	(206)
10.1 文件的基础知识	(206)
10.2 文件的打开和关闭	(209)
10.3 文件的读写	(210)
10.4 文件的定位与随机读写	(218)
习题	(220)
附录 A ASCII 码表	(222)
附录 B 运算符与结合性	(224)
附录 C C 语言关键字	(225)
附录 D C 常用的库函数	(226)
参考文献	(234)

第1章

概 述



1.1 C 语言概述

1.1.1 C 语言的产生和发展

C 语言是目前极为流行的语言之一,它既具有高级语言的功能,又具有机器语言的一些特性。它是 1972 年由美国贝尔(Bell)实验室的 Dennis Ritchie 和 Brian Kernighan 等人首先推出的,之后,C 语言又经过不断改进,逐步完善。直到 1978 年 Brian Kernighan 和 Dennis Ritchie(简称 K&R)合著了影响深远的著作《The C Programming Language》,首次系统地介绍了 C 语言,建立了所谓的 C 语言的 K&R 标准,该标准一度成为 C 语言的事实标准。

此后,C 语言的发展非常迅速,各种版本的 C 语言相继涌现。没有统一的标准,使得这些 C 语言之间出现了一些不统一的地方。为了改变这种情况,美国国家标准学会(American National Standards Institute,ANSI)于 1983 年制定了一套标准,称为 ANSI C(标准 C),它成为各种 C 语言版本的基础。

C 语言本身也在发展。20 世纪 80 年代中期,出现了面向对象程序设计的概念,贝尔实验室的 B. Stroustrup 博士借鉴了 Simula 67 中的类的概念,将面向对象的语言成分引入到 C 语言中,设计出了 C++ 语言,C++ 语言赢得了广大程序员喜爱,不同的机器、不同的操作系统几乎都支持 C++ 语言。同时,C++ 语言也得到了国际标准化组织(ISO)的认可。

1.1.2 C 语言的特点

C 语言之所以能存在和发展,并具有生命力和成为程序员的首选语言之一,是因为其具有如下特点。

(1)C 语言既具有高级语言的通用性及易写易读的特点,又具有汇编语言的“位处理”“地址操作”等能力。这使得 C 语言不仅像 Pascal、Fortran、Basic 等高级语言一样用于应用软件的设计,还能像汇编语言一样用于计算机系统软件和控制软件的开发。

(2)C 语言是一种结构化程序设计语言,具有丰富的数据类型、众多的运算符,使程序员能轻松地实现各种复杂的数据结构和运算;C 语言具有实现结构化程序设计的控制结构,以及具备抽象功能并体现信息隐蔽思想的函数,可以实现程序的模块化设计。

(3)语句简练、紧凑,语法规定少,使用方便灵活;编译后生成的代码质量高,运行速度快。

(4)C 语言具备良好的可移植性。若程序员在书写程序时严格遵循 ANSI C 标准,则其源代码基本上不做修改,就能用于各种型号的计算机和各种操作系统。

(5)语言功能丰富,它不仅提供了丰富的运算符,还提供了各种功能强大的系统函数。

尽管 C 语言有很多优点,但也存在一些缺点和不足。比如它的类型检验和转换比较随便,优先级太多,不便记忆。这些都对程序设计者提出了更高要求,也给初学者增加了困难。



1.2 C 程序的结构

用 C 语言编写的源程序,简称 C 程序。C 程序是一种函数结构,一般由一个或若干个函数组成,其中必须有一个名为 main() 的函数,程序的执行就是从这里开始的。

下面先介绍 3 个简单的 C 程序,然后分析 C 程序的结构。

1.2.1 简单的 C 程序

【例 1-1】 在屏幕上输出一行文本信息。

```
int main(void)                /* 主函数 */
{
    printf("How are you!");    /* 在屏幕上输出一行文本信息 "How are you!" */
    return 0;}
```

说明:

- (1) 本程序的功能是在屏幕上显示一行文本信息 "How are you!"。
- (2) main() 为主函数名。每个 C 程序都必须有一个 main() 函数。
- (3) 花括号 "{}" 是函数体界定符,位于花括号 {} 中的内容称为函数体,每个函数都必须用一对花括号将函数体括起来。
- (4) 函数体中只有一条输出语句 "printf("How are you!");",其目的是将引号中的内容 "How are you!" 原样输出。printf() 为 C 语言的标准输出函数,是系统提供的库函数。
- (5) 语句后面有一个分号 ";",这是 C 语言的语句结束符。
- (6) /* */ 是注释语句,用来帮助读者阅读程序,在程序编译运行时 /* 和 */ 之间的内容是不起作用的,注释语句可写在程序中的任何位置。

【例 1-2】 编写一段 C 程序,计算并输出两数之和。

```
#include "stdio.h"           /* 编译预处理命令 */
int main(void)                /* 主函数 */
{
    int a,b,sum;              /* 定义 3 个整型变量 a,b,sum */
    a=21;                     /* 给变量 a 赋值 */
    b=34;                     /* 给变量 b 赋值 */
    sum=a+b;                  /* 计算 a+b 的值并送到变量 sum 中保存 */
    printf("The sum is %d",sum); /* 输出文字 "The sum is" 和变量 sum 的值 */
    return 0;}
```

程序运行结果如下:

The sum is 55

说明:

(1) #include 是编译预处理命令。由双引号括起来的“stdio.h”称为“头文件”，在 stdio.h 文件中定义了 I/O 库所用到的某些宏和变量，其作用是将双引号(或角括号)括起来的文件中的内容，读入到此命令的位置处。在使用 C 语言的输入/输出库函数时，一般需要用 #include 命令将“stdio.h”包含到源文件中。

(2) 在 main() 函数中首先定义了 3 个整型变量 a、b、sum。

(3) 语句“a=21;b=34;”对变量 a、b 进行赋值。

(4) 语句“sum=a+b”计算 a+b 的值并将它送给 sum 变量。

(5) printf() 函数调用完成 sum 的打印，即将文字“The sum is”和运算结果 55 一起输出。其中“%d”是输入/输出格式符，用来指定输入输出时的数据类型和格式，%d 表示十进制整数类型，在执行输出时，此位置上以 sum 变量中的十进制整数数值代替。

【例 1-3】 从键盘上输入两个整数，比较两个数的大小，并输出较大者。

```
#include <stdio.h>           /* 编译预处理命令 */
int main(void)               /* 主函数 */
{
    int a,b,c;                /* 定义整型变量 a,b,c */
    printf("Enter Two Numbers:"); /* 输出提示信息 */
    scanf("%d%d",&a,&b);      /* 从键盘接收 2 个整数并送到变量 a,b 中 */
    c=max(a,b);               /* 调用 max() 函数,将得到的值赋给变量 c */
    printf("max=%d\n",c);     /* 输出文字“max=”和变量 c 的值,并换行 */
    return 0;
}

int max(int x,int y)          /* 定义 max() 函数,函数值为整型,x,y 是整型的形式参数 */
{
    int z;                    /* 定义 max() 函数中用到的整型变量 z */
    if(x>y) z=x;               /* 比较 x,y 的大小,将大的赋值给变量 z */
    else z=y;
    return(z);                 /* 将 z 中的值由函数名 max 带回调用处 */
}
```

程序运行结果如下:

```
Enter Two Numbers:64 28 ↵(回车)
max=64
```

说明:

(1) #include 是编译预处理命令。

(2) 本程序包括两个函数，主函数 main() 和被调用函数 max()，max() 函数的作用是将 x 和 y 中较大者的值赋给变量 z，并由 return 语句将 z 中的值返回给调用函数处。

(3) main() 函数中第一个 printf() 函数输出一行提示信息“Enter Two Numbers”，提示用户从键盘输入两个整数。

(4)scanf()是一个标准输入函数,它完成 a、b 两个变量的输入工作。语句中的“&”表示“取地址”,此处 scanf()函数的作用是从键盘上输入两个数 64 和 28,并将这两个数分别送到由 &a 和 &b 所标识的地址单元中,使得变量 a 的值为 64,变量 b 的值为 28;语句中的“%d”是输入、输出格式符,用来指定输入、输出时的数据类型和格式,%d 表示十进制整数类型,它指定输入的两个数据按十进制整数形式输入。注意两数之间要用空格作分隔。

(5)语句“c=max(a,b);”调用函数 max()求得 a、b 中的最大值并将其赋给变量 c,在调用时将实际参数 a、b 的值传送给函数 max()的形式参数 x、y。

(6)main()函数中的第二个 printf()函数在将“max=%d”输出时,其中的“%d”由 c 的值代之。

本例中所用到的函数调用、形参(形式参数的简称)和实参(实际参数的简称)等概念,这里只是做简单的解释,在以后的章节中会详细介绍。通过上述几个简单的例子,下面总结 C 程序的基本结构。

1.2.2 C 程序的结构

C 程序的一般形式如下:

```

编译预处理命令、全局变量、函数声明等
main()                /* 主函数 */
{
    声明部分
    执行部分
}
sub1()                 /* 自定义函数 sub1 */
{
    声明部分
    执行部分
}
...
subn()                 /* 自定义函数 subn */
{
    声明部分
    执行部分
}

```

C 程序结构的特点如下。

(1)C 程序是由函数构成的。一个 C 程序至少包含一个主函数,该函数有固定的名字 main(),任何 C 程序都是从 main()函数开始执行的。一个 C 程序可以包含一个 main()函数和若干个其他函数。函数是 C 程序的基本单位,被调用的函数可以是系统提供的库函数(如 scanf()函数和 printf()函数),也可以是用户根据需要自己编写的完成特定功能的函数(如例 1-3 中的 max()函数)。事实上,C 程序中的函数相当于其他语言中的子程序。C 语言的库函数非常丰富,标准 C 提供 100 多个库函数,Turbo C 和 MSC 4.0 提供 300 多个库函数。C 语言的这种特点为模块化的程序设计和实现提供了方便。

(2)一个函数由函数首部和函数体两个部分组成。

①函数首部,即函数的第一行,包括函数类型、函数名、函数参数(参数类型,参数名),例 1-3 中的函数 max()的首部如下:

int	max	(int	x,	int	y)
↓	↓	↓	↓	↓	↓
函数类型	函数名	形参 x 的类型	形参名	形参 y 的类型	形参名

②函数体,即函数首部下面的花括号{}括起来的部分。

函数体一般包含两个部分,即“声明部分”和“执行部分”。如例 1-3 中的 max()函数:

```
int max(int x,int y)
{
    int z;           /* 变量声明部分 */
    if(x>y) z=x;      /* 执行部分 */
    else z=y;
    return(z);
}
```

(a)声明部分:用来定义程序中所用到的变量、声明要用到的函数等。如例 1-3 中 main()函数中的“int a,b,c;”和 max()函数中的“int z;”,就属于变量的定义。

当然,有的函数可以没有变量声明部分,比如例 1-1 中的 main()函数。

(b)执行部分:由若干语句组成,完成对数据的运算及各种处理。

(3)一个 C 程序总是从 main()函数开始执行,main()函数可以放在程序中的任何位置。

(4)C 程序的书写格式自由,一行内可以写一个或几个语句,一个语句也可以分写在多个行上。C 程序没有行号,每个语句和变量定义必须用一个分号结尾。

(5)可以用/*……*/对程序的任何部分进行注释,增加程序的可读性。

1.3 C 程序的上机步骤

前面已经给出了一些用 C 语言编写的程序,但它们是不能直接运行的,因为计算机只能识别和执行由 0 和 1 组成的二进制的指令,而不能识别和执行用高级语言编写的程序。为了使计算机能执行用高级语言所编写的程序,必须先用一种称为“编译程序”的软件,把程序翻译成二进制形式的“目标程序”(object program),然后将该目标程序与系统的函数库和其他目标程序连接起来,形成可执行的程序才能被机器所执行。相对于目标程序,用高级语言编写的程序被称为“源程序”(source program)。

在此将介绍利用 Visual C++ 6.0 编译程序,把 C 语言源程序编译连接生成可执行程序(*.exe 文件)。假设 C 源程序名为 MY.c,其编辑、编译、连接、执行过程如图 1-1 所示。具体操作步骤如下。

(1)编辑源程序,并以扩展名.c 的文件存盘。

(2)对源程序进行编译,将源程序转换为扩展名为.obj 的目标程序,但目标程序仍不能运行。若源程序有错,必须予以修改,然后重新编译。

(3)对编译通过的源程序连接,即加入库函数和其他二进制代码生成可执行程序。连接过程中,若出现未定义的函数等错误,必须修改源程序,并重新编译和连接。

(4)执行生成的可执行代码,若不能得到正确的结果,必须修改源程序,重新编译和连接。若能得到正确结果,则整个编辑、编译、连接、执行过程顺利结束。

下面介绍 Visual C++ 6.0 集成开发环境的安装与使用。

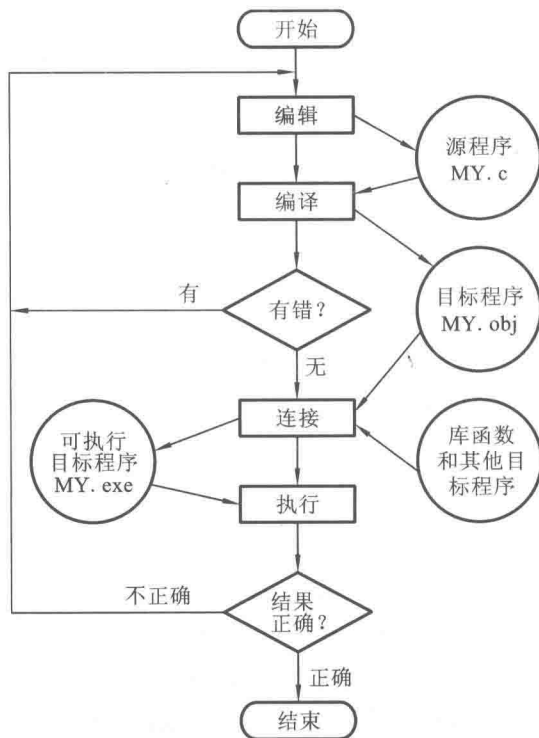


图 1-1 程序的编辑、编译、连接和执行过程



1.4 Microsoft Visual C++ 6.0 集成开发环境

Visual C++是 Microsoft(微软)公司的 Visual Studio 开发工具箱中的一个 C++ 程序开发包。Visual C++ 软件包中的 Developer Studio 是一个集成开发环境,它集成了各种开发工具和 VC 编译器。程序员可以在不离开该环境的情况下编辑、编译、调试和运行一个应用程序。

1. 创建一个新的 C 语言的工程

1) 启动 Microsoft Visual C++ 6.0

可以通过“开始”菜单、桌面快捷方式或者快速启动工具栏等方式启动 Microsoft Visual C++ 6.0,如图 1-2 所示。



图 1-2 Visual C++ 6.0 启动界面

2) 新建工程

在 Visual C++ 6.0 中,一个源程序文件必须属于一个工程,所以必须先创建一个工程。

(1)选择“文件”菜单的“新建”命令,打开“新建”对话框,单击该对话框中的“工程”选项卡,如图 1-3 所示。

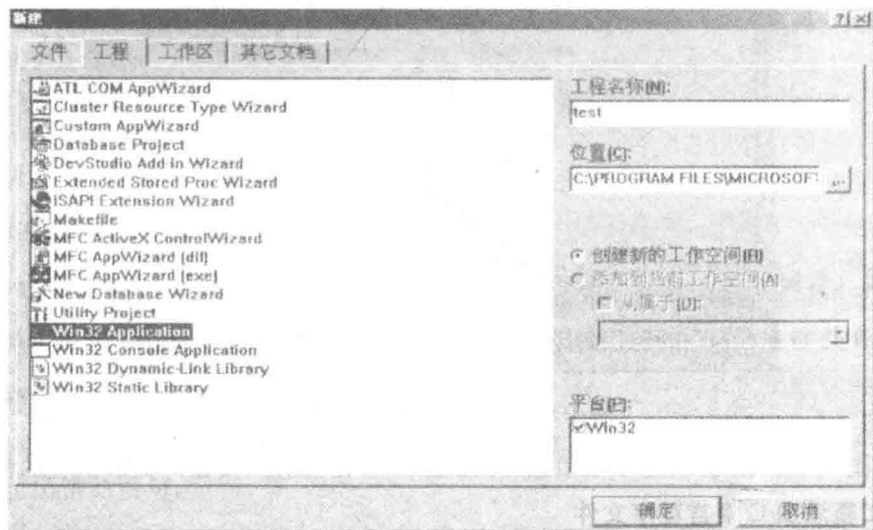


图 1-3 “新建”对话框(“工程”选项卡)

(2)在图 1-3 中选择“Win32 Console Application”选项,并在“工程名称”文本框中输入工程的名称,比如“test”,在“位置”框中选择工程保存的文件夹,单击“确定”按钮,打开图 1-4 所示的工程向导对话框。

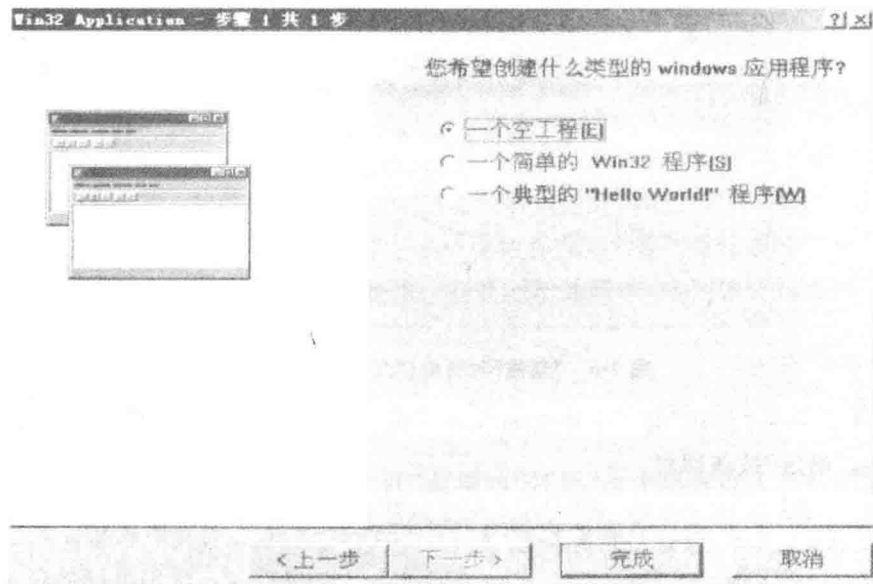


图 1-4 工程向导对话框

(3)在图 1-4 中选择第一个选项,也就是默认选项“一个空工程”,然后单击“完成”按钮,打开图 1-5 所示的“新建工程信息”对话框。

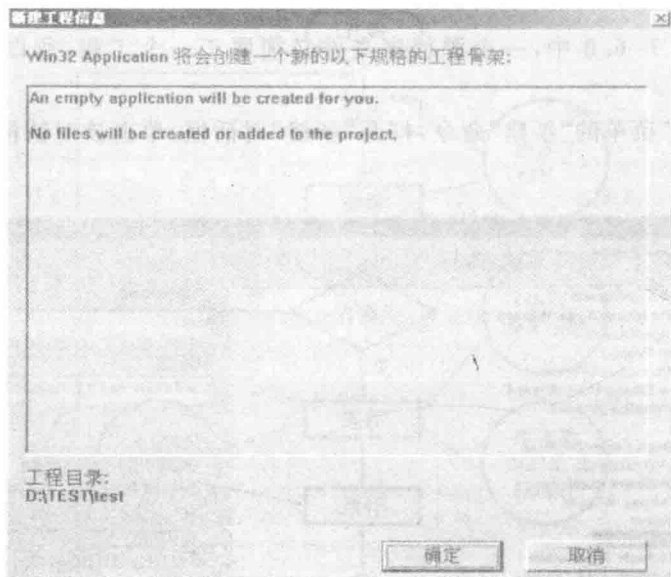


图 1-5 “新建工程信息”对话框

(4)在图 1-5 所示的“新建工程信息”对话框中,单击“确定”按钮即完成工程的创建。

2. 向工程添加 C 语言程序文件

创建好一个空工程后,需要为工程添加一个 C 语言源代码程序文件,具体添加方法为:单击“文件”菜单的“新建”命令,打开图 1-6 所示的“新建”对话框。



图 1-6 “新建”对话框(“文件”选项卡)

在图 1-6 所示的“新建”对话框中选择“C++ Source File”,并在“文件名”文本框中输入文件名,单击“确定”按钮即可。

注意:文件的扩展名应该输入.c,例如“t1.c”。

3. 编写 C 语言源程序

添加一个文件后,可以在 Visual C++ 程序编辑窗口中编写 C 语言源程序,如图 1-7 所示。



图 1-7 程序编辑窗口

4. 编译、连接和执行

编写好源程序后,可以单击“组建”工具栏的  图标,对源程序进行编译。如果编译没有错误,可以单击“组建”工具栏的  图标,对源程序进行连接。如果连接没有错误,可以单击“组建”工具栏的  图标,执行程序。

如果编译过程出错,在组建窗口中将显示出错信息,如图 1-8 所示。用户可以双击出错信息,定位出错的语句。

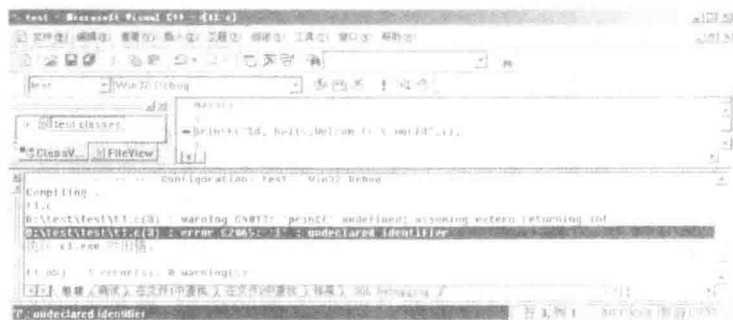


图 1-8 编译出错信息

5. 关闭工作空间

在 Visual C++ 6.0 中,调试运行完一个应用程序后,用户需要通过“文件”菜单的“关闭工作空间”命令关闭当前工作空间,然后才能新建新的工作空间或打开新的 C 语言源程序文件。

6. 打开 C 语言源程序文件

1) 打开文件

可以通过 Visual C++ 6.0 的“文件”菜单的“打开”命令或标准工具栏的  图标,打开一个 C 语言源程序文件。注意,在打开一个 C 语言源程序文件时,应该关闭当前工作空间,否则,连接时会出错。

2) 编译、连接和执行

编译一个打开的 C 语言源程序文件时,单击“组建”工具栏的  图标进行编译,将弹出图 1-9 所示的自动创建工作空间对话框。