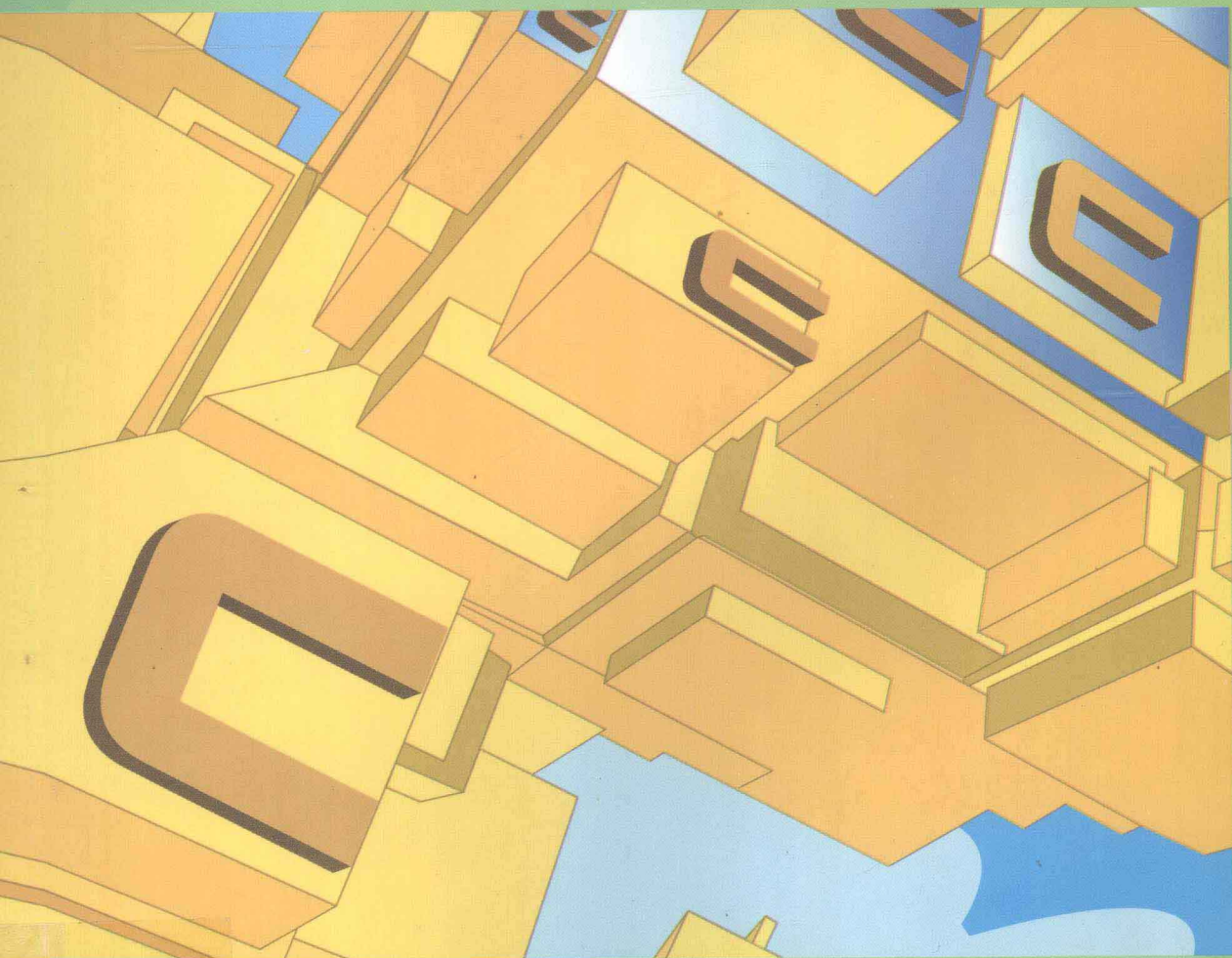




高职高专“十一五”规划标准化教材

C语言 程序设计教程



主 编 蒋腾旭 黄怡旋
副主编 谢 枫 周 军 张 律
刘 娅 周演汇



北京航空航天大学出版社



高职高专“十一五”规划标准化教材

C 语言程序设计教程

主 编 蒋腾旭 黄怡旋

副主编 谢 枫 周 军 张 律

刘 娅 周演汇

北京航空航天大学出版社

内容简介

本书是高职高专学生学习 C 语言程序设计的理想教材。全书共分 12 章,主要包括:C 语言概述、数据类型、运算符与表达式、程序控制语句、数组及其应用、函数、复合结构类型、指针与文件等。全书以 ANSI C 语言标准为基础,以培养学生 C 语言程序设计能力为主线,介绍了程序设计的基本概念、C 语言的语法规则和常用的 C 语言程序设计技术。

本书结合实际应用,以 Visual C++ 6.0 为开发环境,强调算法与编程思想的培养,力图给初学者一个好的程序设计入门知识介绍。为方便教学,在每章最后均安排了一定数量的习题。配套教学资料和习题解答均可在北京航空航天大学出版社网站(<http://www.buaapress.com.cn>)或江西九江职业大学计算机系网站(<http://www.jjvu.jx.cn/>)下载。

本书既可作为高等职业院校计算机程序设计的入门教材,也可作为全国计算机等级考试(二级 C 语言)的参考教材,还可作为科技人员自学 C 语言的参考书。

图书在版编目(CIP)数据

C 语言程序设计教程/蒋腾旭主编. — 北京:北京航空航天大学出版社, 2009. 8

ISBN 978-7-81124-857-9

I. C… II. 蒋… III. C 语言—程序设计—高等学校:技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 122715 号

C 语言程序设计教程

主 编 蒋腾旭 黄怡旋

副主编 谢 枫 周 军 张 律

刘 娅 周演汇

责任编辑 胡 敏 刘 标

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100191) 发行部电话:010-82317024 传真:010-82328026

<http://www.buaapress.com.cn> E-mail: bhpess@263.net

北京市媛明印刷厂印装 各地书店经销

*

开本:787×1 092 1/16 印张:15.5 字数:397 千字

2009 年 8 月第 1 版 2009 年 8 月第 1 次印刷 印数:3 500 册

ISBN 978-7-81124-857-9 定价:26.00 元

前言

C 语言作为国际上广泛流行的通用程序设计语言,在计算机的研究和应用领域发挥着重要的作用。C 语言是一种典型的结构化程序设计语言,它处理能力强、使用灵活、应用范围广,具有良好的可移植性,既适合于计算机专业人员编写系统软件,又适合于开发人员编写应用软件,是广大计算机应用人员应掌握的基本功。

在许多高职院校,C 语言是学生接触的第一门计算机语言,是所有计算机语言的基础,是迈入计算机软件殿堂的敲门砖。但 C 语言往往令初学者感到掌握起来很吃力,原因之一是教材或多或少地存在一些不足,如太重视知识的系统性和全面性,学时与内容矛盾突出;过分讲解语法、忽略算法;数学类例题偏多,与学生生活联系不紧密,学生难以学以致用等。

本书在作者多年 C 语言教学的基础上,尽量克服以上不足,采用分散难点、减小台阶的思路,依据高职院校学生的学习实际,对 C 语言知识点进行合理取舍、适当归类编排,力图体现以下几个特色。

1. 本书对知识点的介绍详略得当、取舍合理。本书重点讲解 C 语言常用和基本知识,对 C 语言中那些过时的、偏僻的、对后续课程无多大帮助的知识进行了适当删除。

2. 本书以培养学生编程意识和编程能力为主线,较好地处理了算法和语法的关系,使初学者通过本门课程的学习,一方面能掌握 C 语言的基本概念、基本知识,另一方面更重要的是培养初学者逻辑思维能力、编程意识和思想,力争为后续课程的学习打下坚实的基础。

3. 本书能满足分层次教学需求。对非计算机专业,可只讲授前 9 章内容;对计算机专业,可讲授全部内容,其中第 12 章可根据需要进行选讲。

本书结合实际应用,以 Visual C++ 6.0 为开发环境,深入浅出地讲解了用计算机解决问题的方法;本书内容编排体系合理、逻辑清晰、例题丰富、通俗易懂,覆盖了《全国计算机等级考试考试大纲》(二级 C 语言)的程序设计考试要求。

C 语言程序设计是一门实践性很强的课程,读者一定要重视培养自己动手编程和上机调试运行程序的能力。全书各章最后配有一定量的主观题和上机实训题,方便读者课后复习强化掌握所学知识。

本书由江西九江职业大学计算机系蒋腾旭老师任主编。其中第 1、2 章由周

军编写,第3、4章由谢枫编写,第5、8章由蒋腾旭编写,第6、7章由张律编写,第9、10章、附录A、附录B由周演汇编编写,第11、12章、附录C、附录D由刘娅编写。全书由蒋腾旭、黄怡旋负责统稿和校稿。

由于编者水平有限及时间仓促,对于书中存在的疏漏甚至错误之处,恳请读者批评指正。

编 者

2009年5月

目 录

第 1 章 C 语言概述	1
1.1 程序设计语言与 C 语言	1
1.2 C 语言程序的开发环境	5
1.3 C 语言词汇	11
1.4 本章小结	12
习题一	12
第 2 章 C 语言数据类型	14
2.1 C 语言的数据类型	14
2.2 标识符	14
2.3 常量及其类型	15
2.4 变量及其类型	19
2.5 变量的初始化	24
2.6 各种基本数据类型的混合运算	25
2.7 本章小结	26
习题二	26
第 3 章 运算符与表达式	28
3.1 运算符及表达式概述	28
3.2 算术运算符与算术表达式	29
3.3 赋值运算符与赋值表达式	30
3.4 关系运算符与关系表达式	31
3.5 逻辑运算符与逻辑表达式	32
3.6 其他运算符及其表达式	34
3.7 运算符的优先级和结合性	37
3.8 本章小结	37
习题三	38
第 4 章 顺序结构程序设计	40
4.1 结构化程序设计的基本思想	40
4.2 C 程序的结构	41
4.3 C 语句的分类	41
4.4 算法及流程图	43
4.5 输入/输出函数的使用	48
4.6 顺序结构程序设计	57
4.7 本章小结	58



习题四	58
第 5 章 选择结构程序设计	61
5.1 选择结构程序设计的基本思想	61
5.2 选择结构程序设计语句	61
5.3 本章小结	71
习题五	72
第 6 章 循环结构程序设计	76
6.1 循环结构程序设计的基本思想	76
6.2 循环结构程序设计语句	76
6.3 循环转移控制语句	90
6.4 本章小结	93
习题六	93
第 7 章 数组及其应用	98
7.1 数组结构	98
7.2 一维数组的定义及应用	98
7.3 二维数组的定义及应用	104
7.4 字符串与字符数组	110
7.5 本章小结	120
习题七	120
第 8 章 函 数	125
8.1 函数概述	125
8.2 函数的定义	127
8.3 函数的参数和返回值	128
8.4 函数的声明和调用	132
8.5 变量的作用域与生存期	140
8.6 内部函数和外部函数	146
8.7 本章小结	147
习题八	148
第 9 章 编译预处理	151
9.1 宏定义	151
9.2 文件包含	157
9.3 条件编译	159
9.4 本章小结	161
习题九	161
第 10 章 复合结构类型	165
10.1 结构体	165



10.2 共用体.....	173
10.3 枚举类型.....	177
10.4 类型定义符 typedef	179
10.5 本章小结.....	180
习题十.....	181
第 11 章 指针与文件	184
11.1 指针与指针变量.....	184
11.2 指针与数组.....	188
11.3 指针与字符(串).....	193
11.4 指针与函数.....	194
11.5 指针的其他使用.....	198
11.6 文件及其定义.....	201
11.7 文件操作.....	202
11.8 本章小结.....	211
习题十一.....	211
第 12 章 位 运 算	215
12.1 概 述.....	215
12.2 位运算符.....	215
12.3 位 段.....	219
12.4 本章小结.....	222
习题十二.....	222
附 录	223
附录 A 常用字符与 7 位 ASCII 码对照表	223
附录 B 运算符的优先级和结合性	224
附录 C 常用库函数	225
附录 D 常见错误分析与程序调试.....	229
参考文献	238

第 1 章

C 语言概述

本章要点

C 语言表达能力强,应用面广,目标程序效率高,可移植性好,既具有高级语言的优点,又具有低级语言的许多特点,是一种功能很强、很灵活的编程语言。本章要点如下:

- C 语言的发展简史和特点。
 - C 语言程序的结构与书写规则。
 - C 语言基本词汇的简单介绍。
 - 在 Visual C++ 6.0 和 WIN-TC 集成环境下,运行 C 程序的一般过程。
-

1.1 程序设计语言与 C 语言

计算机是由硬件系统和软件系统两大部分构成的,硬件是物质基础,软件可以说是计算机的灵魂,没有软件,计算机是一台“裸机”,是什么也不能干的,有了软件,才能成为一台真正的“电脑”。所有的软件,都是要用计算机语言即程序设计语言来编写的。

1.1.1 程序设计语言的发展

程序设计语言(program language)也称计算机语言,即编写计算机程序所用的语言。计算机程序设计语言的发展,经历了机器语言—汇编语言—高级语言的发展历程。

1. 机器语言

机器语言是第一代计算机语言。它直接用二进制代码 0 和 1 表示每一条指令,是一种最低级的计算机语言。比如,某种计算机的指令为 1011011000000000,它表示让计算机进行一次加法操作;而指令 1011010100000000 则表示进行一次减法操作。机器语言直接面向硬件,能直接被计算机所认识,执行速度快,但通用性差,读写困难,编程效率极低,可移植性差。

2. 汇编语言

汇编语言是第二代计算机语言。为了克服机器语言的缺点,产生了汇编语言,汇编语言是用符号来代替机器语言的语言,比如用“add ax,cx”代表加法指令,“mov al,5”代表数据传递。汇编语言具有机器语言的全部优点,并且比机器语言更容易理解、易于调试和修改。但是,汇编语言对计算机硬件的依赖性大,要求熟悉计算机的内部结构,在编写复杂程序时,相对于高级语言代码量较大并且难记,移植性也不好。



3. 高级语言

高级语言是第三代计算机语言。它是接近人类习惯使用的自然语言和数学语言的计算机程序设计语言。它独立于计算机,用户可以不了解机器指令,也可以不必了解机器的内部结构和工作原理,就能用高级语言编写程序。高级语言通用性好,易学习,易使用,可移植性好,重用率高,易于交流和推广。但用高级语言编写的程序计算机不能直接运行,必须经过编译和翻译才能被计算机执行。高级语言有很多,目前使用较普遍的有 Fortran, Basic, Pascal, C, C++, Visual C++, Visual Basic, Visual Foxpro, Delphi 和 Java 等。

1.1.2 C 语言简介

1. C 语言的发展过程

在 C 语言诞生之前,系统软件主要用汇编语言编写,汇编语言可读性及可移植性差,而一般的高级语言又难以实现对计算机硬件的直接操作,于是人们就希望有一种兼有汇编语言和高级语言的新语言出现,而 C 语言就是在这种背景下产生的。

C 语言的前身要追溯到 ALGOL 60 语言,虽然 ALGOL 60 语言引进局部变量和递归的概念,但离硬件较远,不宜编写系统软件。1963 年,英国剑桥大学推出 CPL 语言。CPL 语言相对于 ALGOL 60 语言接近硬件一点,但规模较大,不易实现。1967 年英国剑桥大学的 Martin Richards 在 CPL 语言的基础之上推出了 BCPL,1970 年美国贝尔实验室的 Ken Thompson 以 BCPL 为基础推出了 B 语言,B 语言很简单且接近于硬件,但由于过于简单,功能有限。1973 年贝尔实验室的 D. M. Ritchie 在 B 语言的基础上设计出了 C 语言。C 语言保持了 BCPL 和 B 语言接近于硬件的优点,克服了它们过于简单、数据无类型的缺点,同时还保留了高级语言可读性、可移植性好的优点。后来 C 语言又被多次改进,出现了很多版本,现在 C 语言已经成为世界上应用最广泛的几种计算机语言之一。

2. C 语言的特点

C 语言具有汇编语言和高级语言的特点。具体来说,其主要特点如下所述。

① C 语言简洁、紧凑、使用方便、灵活。共有 9 种控制语句,32 个关键字,程序书写自由,主要用小写字母表示。

② 数据结构丰富,具有各种各样的数据类型,能够实现各种复杂的数据结构运算,特别是指针类型数据,比 Pascal 语言更为灵活、多样。

③ C 语言运算符丰富,共有 34 种运算符,把括号、赋值等都作为运算符处理,从而可以实现其他高级语言中难以实现的运算。

④ 机器生成的目标代码质量高,程序执行效率高,与汇编语言相比程序可移植性好。

⑤ 可以直接访问硬件,进行位操作,可以用来编写系统软件,这是其他高级语言不具备的特点。

3. C 源程序的结构特点

由汇编语言和高级语言编写的程序称为源程序。由 C 语言构成的程序称为 C 源程序。首先看以下几个案例。

【例 1.1】打印“九江职业大学”的汉语拼音。

程序源码如下:



```
main()
{
    printf("jiu jiang zhi ye da xue!"); /* 输出计算结果 */
}
```

程序运行结果如下:

```
jiu jiang zhi ye da xue!
```

分析:其中 main()是主函数,每一个C程序中都必须要有主函数,本程序中只有一个语句 printf("jiu jiang zhi ye da xue!");,printf()是C语言的输出函数,该函数的作用是把双引号内的字符照原样输出,语句后面用/*和*/括起来的内容为注释部分,目的是增加程序可读性,但程序不执行注释部分。

【例 1.2】求任意两个整数的积。

程序源码如下:

```
#include <stdio.h>
int mul(int a,int b);          /* 函数说明 */
main()                          /* 主函数 */
{
    int x,y,z;                  /* 变量说明 */
    printf("input two numbers:\n");
    scanf("%d%d",&x,&y);        /* 输入 x,y 值 */
    z = mul(x,y);               /* 调用 mul 函数 */
    printf("mul = %d",z);       /* 输出 */
}
int mul(int a,int b)            /* 定义 max 函数 */
{
    int c;
    c = a * b;
    return c;                   /* 把结果返回主调函数 */
}
```

分析:程序中#include <stdio.h>的作用就好像在文件中#include出现的地方输入了stdio.h文件的整个内容。include文件为共享许多文件都需要的信息提供了一条捷径。

stdio.h文件是作为所有C编译器程序包的一部分提供给用户的。它包括有关输入和输出函数的信息,如printf()等,供编译器使用。它的名字是standard input/output header(标准输入/输出头文件)的缩写。C程序员将出现在文件首部的信息集合称为头文件(header),一般C的执行程序都有几个头文件伴随。

有些程序需要包含stdio.h,而有些则不需要。C的具体实现文档应该包括对C库中函数的描述。这些函数描述指明了需要哪些头文件,省略相应的头文件对于某些程序并无影响,但是最好不要出现这种情况,以便于程序的阅读性和维护。每次使用库函数时,都要用到相应函数所指定的include文件。因为编译器利用include中的信息来组建程序,所以,编译器不需要的任何信息都不会成为程序的一部分。因而,包括了不需要的文件并不会造成程序的冗长。



#include 语句是 C 预处理器伪指令的一个例子。通常来说, C 编译器在编译之前先完成源代码的一些工作, 这就叫做预处理过程。

上面例子中程序的功能是由用户输入两个整数, 执行后输出它们的积。本程序由两个函数组成, 主函数和 mul 函数。函数之间是并列关系, 可从主函数中调用其他函数。mul 函数的功能是计算两个数的积, 然后把积返回给主函数。mul 函数是一个用户自定义函数。因此, 在主函数中要给出说明(程序第一行)。可见, 在程序的说明部分中, 不仅可以有变量说明, 还可以有函数说明。

该程序的执行过程是, 首先在屏幕上显示提示串, 请用户输入两个数, 回车后由 scanf 函数语句接收这两个数送入变量 x 和 y 中, 然后调用 mul 函数, 并把 x 和 y 的值传送给 mul 函数的参数 a 和 b。在 mul 函数中计算 a 和 b 的积并赋值给变量 c, 把 c 返回给主函数的变量 z, 最后在屏幕上输出 z 的值。

【例 1.3】求整数 a 和 b 的和。

程序源码如下:

```
main()
{
    /* 程序使用数据的描述部分 */
    int a,b;                /* 定义两个整数变量 a 和 b, 用来保存加数和被加数 */
    int sum;                /* 定义变量 sum, 用来保留 a 和 b 的和 */
    /* 数据的使用部分 */
    scanf("%d,%d",&a,&b);    /* 从键盘上任意输入值给变量 a 和 b */
    sum = a + b;            /* 计算 a 和 b 的和, 结果保存在 sum 中 */
    printf("%d",sum);       /* 输出计算结果 */
}
```

从以上三个程序中, 可以看出:

1) 一个 C 程序最少由两个部分组成。

① 数据描述, 即程序中使用了哪些数据。

程序中的语句:

```
int a,b;    /* 定义两个整数变量, 用来保存加数和被加数 */
int sum;    /* 定义一个变量, 用来保留 a 和 b 的和 */
```

这是数据描述部分。

② 操作步骤的描述, 即将对数据进行什么操作。

下面的语句就是对数据进行操作的部分:

```
printf("jiu jiang zhi ye da xue");    /* 输出双引号中的字符 */
sum = a + b;                          /* a 和 b 相加后赋值给变量 sum */
scanf("%d,%d",&a,&b);                  /* 从键盘上输入任意值给变量 a,b */
```

数据描述部分就像做一道菜时准备的原料, 而操作步骤的描述则是制作菜的过程, 是对原料进行加工的部分。

2) 一个 C 语言源程序可以由一个 main() 主函数或一个 main() 主函数和若干个其他函数组成。



3) `main()`函数可以放在程序中的任何地方。一个C程序总是从`main()`函数开始执行,而不论`main()`函数在程序中的位置如何。

4) C程序的每一个语句都必须以分号结尾,所以书写格式自由,一行内可以写很多语句。

5) 用`/* */`对C程序中的任何部分作注释,以便增加程序的可读性,但编译时不起作用。

6) C程序本身没有输入/输出语句,故输入/输出语句由库函数`scanf()`、`printf()`完成,把输入/输出操作放在函数中处理,可以使编译程序简单,很容易在各种机器上实现,这也是C程序可移植性好的根本原因。

4. C程序的开发过程

计算机只能识别和执行由0,1组成的二进制指令,而不能识别和执行高级语言写的指令,所以必须要用“编译程序”的软件,把高级语言编写的源程序翻译成二进制形式的目标程序,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,形成可执行的目标程序。故开发一个C程序要经过编辑、编译、连接和运行四个步骤。

(1) 编辑

编辑是指C语言源程序的输入和修改。使用文本编辑器来创建源代码的文件,最后以文本文件的形式存放在磁盘上,文件名由用户自行定义,扩展名一般为`.c`。

(2) 编译

编译是指编译器把C语言源程序翻译成二进制目标程序。目标程序文件的主文件名与源程序的主文件名相同,扩展名为`.obj`。如果在编译的过程中出现错误,系统会给出“出错信息”,需要回到编辑阶段进行修改,直到编译通过为止。

(3) 连接

编译成功后的目标程序仍然不能运行,需要用连接程序将编译过的目标程序和程序中用到的库函数连接装配在一起,形成可执行的目标程序。可执行文件的主文件名与源程序的主文件名相同,其扩展名为`.exe`。

(4) 运行

连接后生成的可执行文件装入内存后即可运行,并输出运行结果。

如果在运行程序的过程中得不到预期的结果,就要重复进行编辑、编译、连接和运行四个步骤,如图1-1所示。

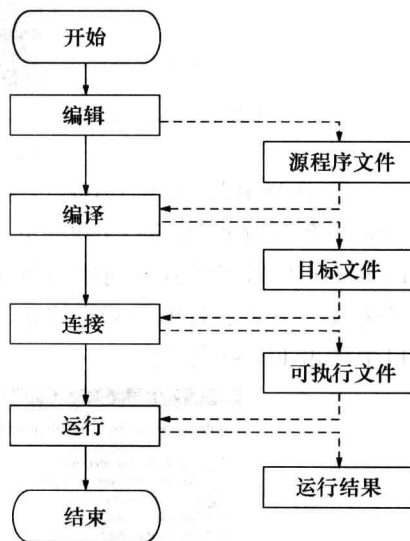


图1-1 C程序开发过程示意图

1.2 C语言程序的开发环境

下面分别就两种不同环境下运行C程序作简单介绍。

1.2.1 Visual C++ 6.0 编程环境

Visual C++ 6.0是微软公司开发C/C++的集成开发环境,可以利用它编辑、编译、调试及运行C/C++语言源程序。该系统是在Windows 9X、Windows NT、Windows 2000和



Windows XP 等版本下运行的,是一种图形化操作界面,可以使用鼠标进行各种操作,使用起来非常方便。

在 Visual C++ 6.0 环境中运行 C 源程序步骤如下所述。

1. 编写源程序

① 启动 Visual C++ 6.0,进入 Visual C++ 6.0 的主窗口,如图 1-2 所示。

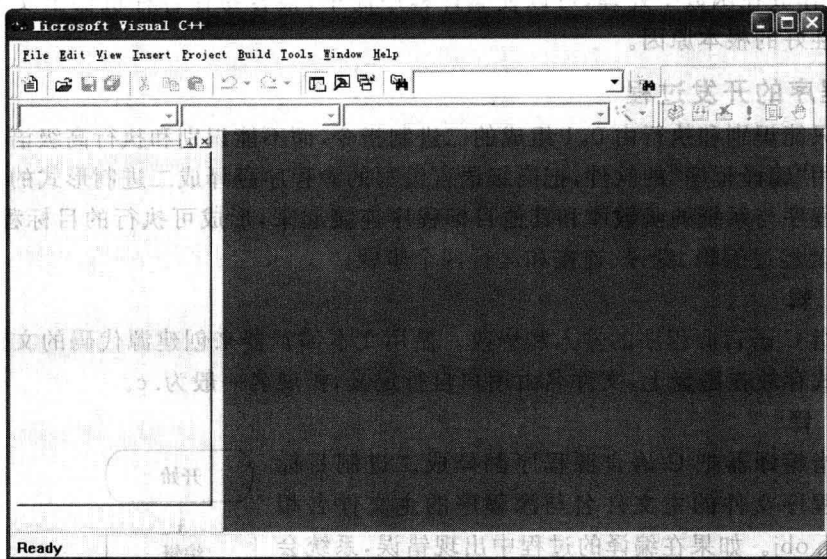


图 1-2 Visual C++ 6.0 的主窗口

② 选择 File|New Projects 菜单项,选择左边列表中的倒数第三项 Win32 Console Application,在右边的 Project name 下方的文本框中输入项目名称(例如:c10_01),如图 1-3 所示。Location 是指编程者所创建的项目及其项目中的文件所存放的位置,可选择默认目录,也可自己更改。设置好后单击 OK 按钮。此时将会弹出一个新的窗口,保持默认状态下的设置,在窗口中单击 Finish 按钮。

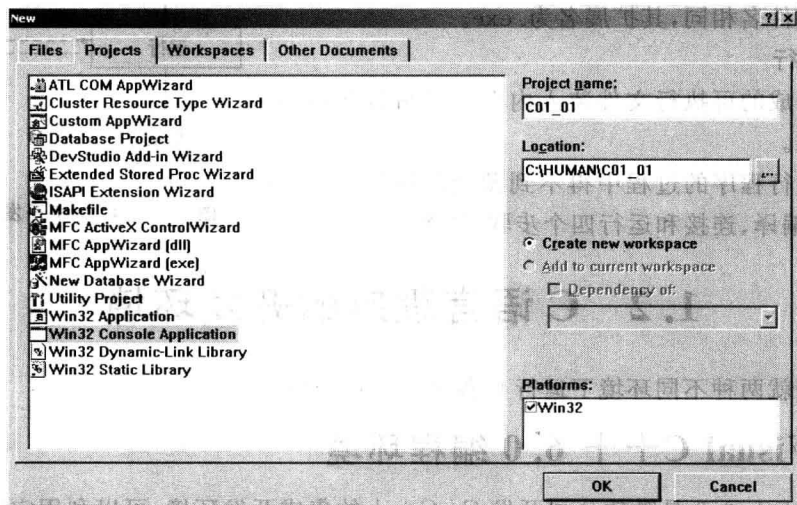


图 1-3 New 窗口



③ 单击 OK 按钮, 返回到 Visual C++ 6.0 的主窗口。这样就建立了一个新项目。

④ 接下来在该项目中添加一个程序文件, 用来输入程序代码。再次选择 File|C++ Source File 菜单项, 如图 1-4 所示。在右边的 File 文本框中输入文件名, 比如: “hello”, 并选中 Add to project 选项。

⑤ 单击 OK 按钮后, 返回到 Visual C++ 6.0 的主窗口, 光标在编辑窗口中闪烁, 编辑窗口已经被激活, 可以输入源程序了, 如图 1-5 所示。至此, 编辑源程序结束。

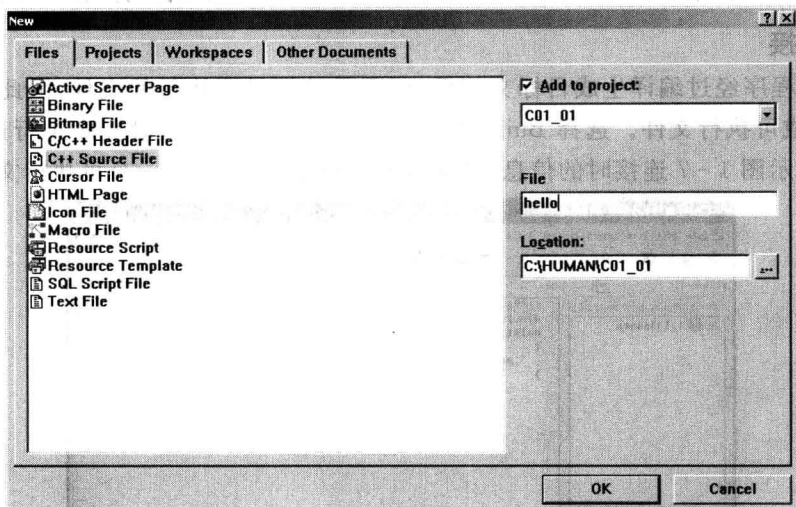


图 1-4 Files 中的选项

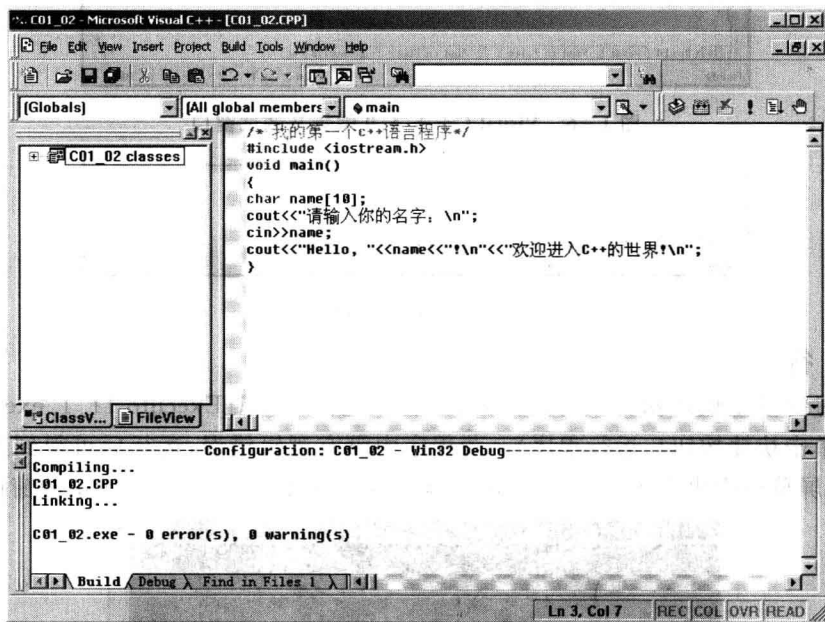


图 1-5 Visual C++ 6.0 编辑窗口

⑥ 经检查无误后, 选择 File|Save 菜单项 (或者按快捷键 Ctrl+S), 将源程序保存在当前指定的文件中。也可以选择 File|Save As 菜单项, 在弹出的 Save As (另存为) 对话框中指定文件路径和文件名进行保存。



2. 编译

编写好程序后,要对该程序进行编译,将源程序编译成目标程序后,机器才能识别该程序,选择 Build|Compile hello.c 菜单项,就可以对程序进行编译。此时在主窗口下面的调试信息窗口中列出了编译的信息;如果程序有错误,就会指出错误的位置和出错原因,那么应找出错误并改正、再运行,直到没有错误为止。如图 1-6 所示,从图中可以看出,源程序有 2 个 error 和 0 个 warning。用鼠标拖动调试信息窗口中右侧滚动条中的滑块,可以看到出错的位置和性质。

3. 连接

C 语言源程序经过编译生成目标文件后,还不是计算机可执行的文件,因此,还要对程序进行连接,生成可执行文件。选择 Build|Build cl1_1.exe 菜单项,对程序进行连接。在调试信息窗口中显示图 1-7 连接时的信息,说明没有发现错误,生成了一个可执行文件 1_1.exe。

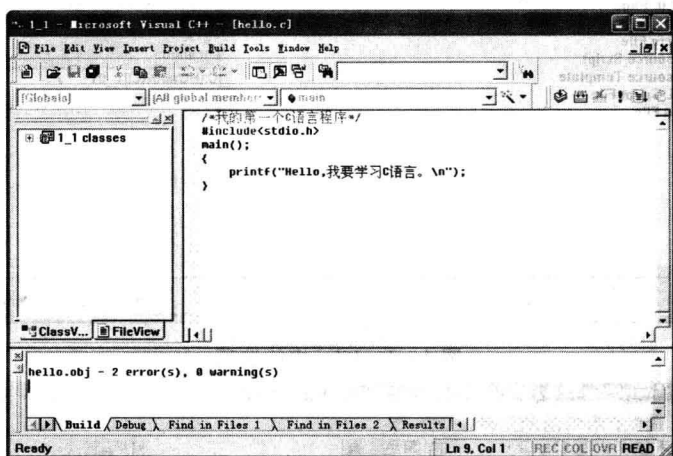


图 1-6 Visual C++ 6.0 编辑及调试窗口

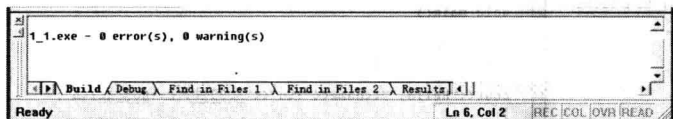


图 1-7 调试信息窗口

4. 运行

程序经过链接生成可执行文件后,即可运行。选择 Build|Execute 1_1.exe 菜单项,(也可直接通过单击快捷按钮!运行程序)。若程序中没有逻辑错误,运行后可得到正确结果,运行完程序后,屏幕的中央出现一个黑色窗口,运行结果显示在如图 1-8 所示的窗口中。

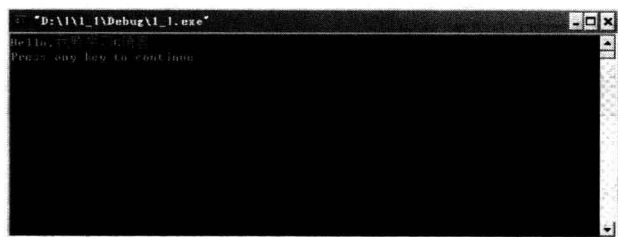


图 1-8 结果显示窗口



1.2.2 WIN-TC 编程环境

WIN-TC 是一个 TC2 Windows 平台开发工具。该软件使用 TC2 为内核,提供了 Windows 平台的开发界面,因此也就支持 Windows 平台下的功能,例如剪切、复制、粘贴和查找替换等。而且在功能上也有它的特色,例如语法加亮、C 内嵌汇编、自定义扩展库的支持等。

1. 编写程序

安装 TC 后,单击 WIN-TC 的图标。如果系统是 WIN 2000 或 WIN XP 的话,将会先看到一个操作系统警告的对话框,提示以后的程序需要按照提示那样在结尾加上 `getch()` 暂停以查看屏幕中显示的输出结果,如图 1-9 所示。WIN-TC 默认打开的文件是 WIN-TC 安装目录下的 `noname.exe`,单击“知道了”按钮即可。选择“文件”|“新建文件”菜单项,出现如图 1-10 所示的界面,就可以开始编写新的 C 源程序。

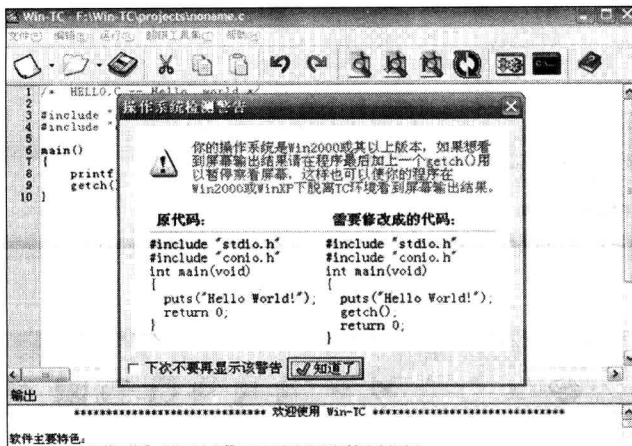


图 1-9 操作系统警告的对话框

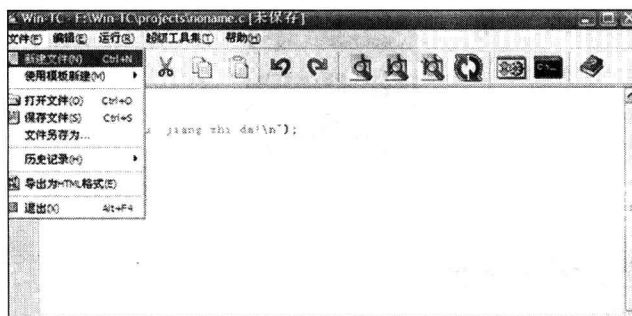


图 1-10 新建文件窗口

编辑完后,经检查无误,选择“文件”|“保存文件”菜单项(或者按快捷键 `Ctrl+S`),将源程序保存在当前指定的文件中,文件名为默认的 `NONAME.c`。也可以选择“文件”|“文件另存为”菜单项,在弹出的“文件另存为”(另存为)对话框中重新为源程序命名。

2. 编译及链接

编写好程序后,要对该程序进行编译,选择“运行”|“编译连接”或“编译连接并运行”菜单