

精品丛书 “24小时轻松掌握系列”

全新改版，重装上市

# C语言程序设计

24  
学时  
轻松掌握

何晓霞 陈旭斌 王立龙 张 华 编著

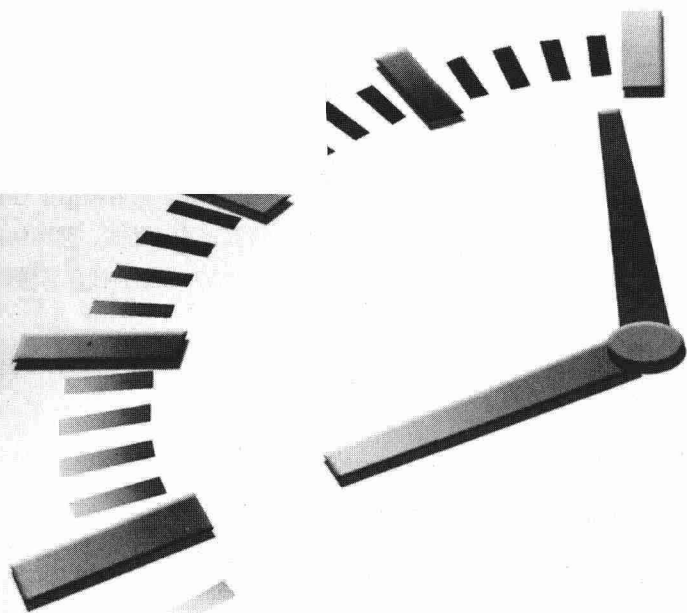
- 科学安排，24学时步步学习，轻松掌握。
- 授之以渔，总结规律确定思路，举一反三。
- 边学边练，理论习题有机结合，夯实基础。



内附书中范例源代码



中国铁道出版社  
CHINA RAILWAY PUBLISHING HOUSE



# C 语言程序设计

## 24 学时轻松掌握

何晓霞 陈旭斌 王立龙 张华 编著

中国铁道出版社  
CHINA RAILWAY PUBLISHING HOUSE

## 内 容 简 介

本书详细介绍了C语言的基础知识,内容概括了C语言大部分的主要知识点,如C语言的运行环境、基本数据类型、运算符和表达式、循环结构、函数和数组的定义及其应用等。

本书以时间为主线,内容安排合理;语言浅显易懂,注重实际操作。读者学起来更轻松,更易掌握,对入门者大有裨益。

本书主要面向初学者,也适合广大编程爱好者入门使用。

## 图书在版编目(CIP)数据

C语言程序设计 24 学时轻松掌握/何晓霞等编著. —北京:中国铁道出版社, 2008. 6

(24 小时轻松掌握系列)

ISBN 978-7-113-08971-9

I. C… II. 何… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2008) 第 080655 号

书 名: C 语言程序设计 24 学时轻松掌握

作 者: 何晓霞 陈旭斌 王立龙 张 华 编著

策划编辑: 严晓舟 荆 波

责任编辑: 苏 茜

编辑助理: 贾 星 高婧雅

封面设计: 付 巍

编辑部电话: (010) 63583215

封面制作: 白 雪

责任印制: 李 佳

出版发行: 中国铁道出版社(北京市宣武区右安门西街8号 邮政编码: 100054)

印 刷: 三河市华丰印刷厂

版 次: 2008 年 8 月第 1 版 2008 年 8 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 15.5 字数: 356 千

印 数: 5 000 册

书 号: ISBN 978-7-113-08971-9/TP·2920

定 价: 29.00 元(附赠光盘)

版权所有 侵权必究

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

## 只需 24 小时， 轻松具备一种电脑技能

进入 21 世纪的你，如果还不能熟练地使用电脑，不能不说是一种遗憾。

电脑的世界是十分美妙的世界，我们通过 Internet 了解世界，通过 E-mail 和朋友们沟通，上网购买所需要的图书……电脑，越来越成为生活的必需品，给我们的工作、学习和生活带来了巨大帮助。

### 只要会中文，就可以享受高科技带来的便利

可是，在今天，还是有不少读者朋友，不会使用电脑，或者说不能熟练地驾驭电脑，让电脑帮我们完成各种工作，体验电脑文化带给我们的神奇感觉，享受高科技的产品带给我们的便利。

很多读者向我们抱怨，电脑学习这么难，而且，没有足够的时间去学习……根据我们多年的教学经验，只要会中文，可以阅读中文书籍，就能够看懂电脑的中文应用界面，培养基本的电脑技能，并逐步地熟练。只要你能定期抽出一个小时的完整时间，认真地实践我们提供的技能培养计划，就一定可以成功地驾驭电脑，并可以体验学习新知识的快乐。

### 科学安排，学会不难

我们把常用的电脑技能，分解成一个一个的学习单元。只要能定期抽出一个小时的空余时间，按照本书的安排，学习其中一个单元，一个小时一点进步，一个小时一点提高。由慢到快，电脑技能很快就可以上一个新的台阶。

按照我们的学习安排，只要 24 小时，一定可以掌握一种电脑应用技能。这个时候，学习的流程安排和内容就相当重要。

根据作者多年的经验，我们在这 24 个小时里面的每一个小时，或者安排读者学习某种技能；或者让读者跟我们学做某个实例；或者让读者强化训练某项技能。这 24 个小时的安排串联起来，就是一张电脑技能的学习地图，它伴随读者探索电脑奥秘的全过程。加上一定时间的训练，一定能教会读者应用电脑，并熟练起来。

### 按图索骥，提高更快

针对任何一项电脑技能的学习，24 小时培养计划，犹如学习中的 24 级台阶，由作者精心设计。读者可按这个学习顺序，由浅入深，由易到难，逐步掌握好有用的电脑技能。

学习是一个由慢到快的过程。每个人的情况不一样，一般来说，前面的基础打好了，后面的学习速度就会越来越快。所以，在一些内容的安排上，

我们遵循了这个特点。在最后的几个小时的学习计划中，学习内容具有并列特性，读者可根据自己的需要选择学习的顺序。

另外，作为正文的补充，有的图书我们还提供了附录，供读者查询某些资料。

### **边学边练，事半功倍**

学习电脑技能，还要讲究一定的技巧。有了完美的学习方案，还要有足够的练习。

根据我们的经验，电脑技能的学习，上机练习非常重要。所以，建议读者在学习的过程中，同时找一台电脑练习所学内容。

一本图书，一台电脑，一边学习，同时按书中所讲进行练习，可加深印象，更能巩固技能，越用越熟练，越用越体会到使用电脑的乐趣。希望我们的每一本书，加上读者的 24 小时自我训练，能使读者的电脑水平在某一个方面得到飞快地提升。

### **联系作者，答疑解惑**

每一个读者，都有不同的基础和学习经验。我们虽然设计了大多数读者的学习地图，但由于每位读者电脑配置不一定相同，学习碰到的问题也可能各不相同。所以，除了本书之外，我们特地开辟了读者答疑邮箱：  
**jb18803242@yahoo.com.cn**。

如果读者在应用电脑的过程中碰到疑难问题，可以发邮件给我们，我们很乐意为您解答，并将典型问题放在下一版的图书中。

编者  
2008 年 5 月

# 前言

目前,我国高校非计算机专业的计算机教育普遍实施“三个层次”的教学方式,即计算机应用基础、计算机技术基础和结合专业的计算机应用。“C 语言程序设计”属于计算机技术基础课程。C 语言已有较长的历史,很多语言都是在 C 语言基础上发展起来的。因此学习 C 语言很重要。通过学习 C 语言可了解面向过程的编程思想并把实际问题建立模型、拆分;使用 C 语言要面对很多底层的東西,如指针、字符串等,对使用更“高级”的语言很有帮助。

C 语言功能丰富,表达力强,使用灵活方便,应用面广,目标程序效率高,可移植性好,因此特别适合于编写系统软件。C 语言诞生后,许多用汇编语言编写的软件,都可以用 C 语言进行编写,而学习 C 语言要比学习汇编语言简单得多。

现在, C 语言不仅为计算机专业工作者所使用,而且为广大计算机应用人员所使用。学习 C 语言已成为广大计算机应用人员和爱好者的迫切需求。

由于 C 语言涉及到的概念比较复杂,规则复杂,使用灵活,容易出错,不少初学者会遇到困难,因此要求学习者应该经常上机进行实践,及时发现问题并及时解决。我们在书中对经常出现的难点和问题进行说明,希望会对读者有所帮助。本书在内容上力求做到实例丰富典型,图文并茂,讲解深浅适宜,叙述条理清楚,让读者轻松阅读。

本书在传授知识的同时,还侧重培养读者的自学能力,以便使读者能够轻松应对在 C 语言中出现的各种问题。

关于本书的特点:

1. 清晰易读,而且尽可能带有趣味性。对普通读者来说,许多 C 语言的书籍都过于烦琐,而本书试图对 C 语言进行一种清晰、全面的讲解,并且决定用适当的随图讲解的方法让其清晰易懂。

2. 适用于广泛的读者群。本书基于的读者都具有一点点编程经验,但他们都尚未精通某种具体的编程语言而编写,因此书中尽量减少“行话”而改用通俗易懂的词汇来定义用到的术语。

3. 本书中的 C 语言程序都是在 Turbo C 2.0 中进行编写调试的,当然 C 程序的运行环境有很多种。

全书共 24 学时,从章节内容上可分为以下几部分:

第 1 部分,第 1 学时。简单介绍 C 语言,着重分步骤地介绍运行一个 C 程序的全过程。

第 2 部分,第 2~4 学时。详细介绍 C 语言中基本的数据类型、运算符和表达式以及数据的输入与输出。这一部分是 C 语言的基础部分,同时也是在后面学习的过程中应该重点注意的内容。

第 3 部分,第 5~7 学时。这里主要介绍常用的几种循环结构。同所有的编程语言一样, C 语言同样用三种程序控制结构就可以完成所有的程序。但是,在本书中并没有讲到顺序结构,主要是由于顺序结构和我们平常的习惯一样,简单地说就是从头读到尾。

第4部分，第8~9学时。介绍C语言中函数的定义和应用。函数可以说是C语言的灵魂，C语言所有的语句都必须在函数体中执行。

第5部分，第10~11学时。介绍数组的知识。数组在C语言中占据相当关键的位置，在实际应用中比较常用。

第6部分，第12学时。这部分是关于预处理命令的知识。

第7部分，第13~17学时。这部分讲解的是指针，这是C语言的精华部分。指针的相关知识在整个C语言编程中会一直用到，它会让你的程序变得简洁、高效。

第8部分，第18~21学时。介绍结构体、联合体、枚举、类型说明、位运算及位段的用法。

第9部分，第22~23学时。这部分包括常见错误和程序调试及文件的基本用法。当然，在C语言编程中会遇到各种各样的错误，这部分只是介绍了一些典型、常见的错误，以方便读者学习简单的程序调试。

第10部分，第24学时。列举了几个实例来体会一下C语言的魅力。

本书主要面向编程初学者，适合于广大编程爱好者以及各行各业需要学习编程语言的人员使用。

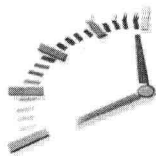
本书由源动力小组组织编写，由何晓霞、陈旭斌、王立龙、张华编著。王莹莹、张辰威、柳军旺、孙永全、张伟、寇国国、江孝林、陈运来、王跃、吴叶伟、柳吉、张博、赵晓明、李月、方杰、史建雪、王宾、石岩、李海峰、陈井彪、刘颖为本书编写提供了帮助，在此一并向他们表示感谢。

由于编者水平有限，加之时间仓促，书中难免有一些不足之处，欢迎专家和读者批评指正。

源动力小组  
2008年7月

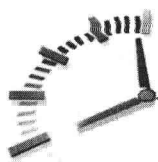
# 目 录

|        |                          |    |
|--------|--------------------------|----|
| 第 1 学时 | C 语言概述 .....             | 1  |
| 1-1    | C 语言的历史背景 .....          | 1  |
| 1-2    | 一个简单的 C 程序 .....         | 2  |
| 1-3    | Turbo C 2.0 集成开发环境 ..... | 6  |
| 1-4    | 本学时总结 .....              | 10 |
| 1-5    | 实战练习 .....               | 10 |
| 第 2 学时 | 基本数据类型 .....             | 11 |
| 2-1    | 常量与变量 .....              | 11 |
| 2-2    | 整型数据 .....               | 13 |
| 2-3    | 实型数据 .....               | 14 |
| 2-4    | 字符型数据 .....              | 16 |
| 2-5    | 本学时总结 .....              | 17 |
| 2-6    | 实战练习 .....               | 17 |
| 第 3 学时 | 运算符和表达式 .....            | 18 |
| 3-1    | 算术运算符和表达式 .....          | 18 |
| 3-2    | 赋值运算符和表达式 .....          | 19 |
| 3-3    | 逗号运算符和表达式 .....          | 20 |
| 3-4    | 关系运算符和表达式 .....          | 21 |
| 3-5    | 各类型数据之间的混合运算 .....       | 22 |
| 3-6    | 自增自减运算符 .....            | 23 |
| 3-7    | 逻辑运算符和表达式 .....          | 25 |
| 3-8    | 运算符的优先级及结合方向 .....       | 27 |
| 3-9    | 本学时总结 .....              | 27 |
| 3-10   | 实战练习 .....               | 27 |
| 第 4 学时 | 数据的输入与输出 .....           | 29 |
| 4-1    | 语句概述 .....               | 29 |
| 4-2    | 字符数据的输入与输出 .....         | 30 |
| 4-3    | 格式输入与输出 .....            | 32 |
| 4-4    | 本学时总结 .....              | 35 |
| 4-5    | 实战练习 .....               | 36 |
| 第 5 学时 | 选择结构 .....               | 38 |
| 5-1    | if 语句的 3 种形式 .....       | 38 |
| 5-2    | if 的嵌套 .....             | 41 |



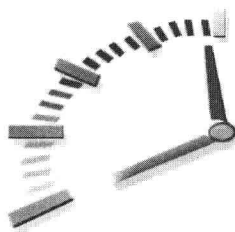
|         |                               |    |
|---------|-------------------------------|----|
| 5-3     | 条件运算符和条件表达式 .....             | 43 |
| 5-4     | switch 语句 .....               | 44 |
| 5-5     | 本学时总结 .....                   | 46 |
| 5-6     | 实战练习 .....                    | 46 |
| 第 6 学时  | 循环结构 .....                    | 48 |
| 6-1     | goto 语句 .....                 | 48 |
| 6-2     | while 语句 .....                | 49 |
| 6-3     | do-while 语句 .....             | 50 |
| 6-4     | for 语句 .....                  | 51 |
| 6-5     | 循环的嵌套 .....                   | 53 |
| 6-6     | 本学时总结 .....                   | 54 |
| 6-7     | 实战练习 .....                    | 54 |
| 第 7 学时  | 跳转语句 .....                    | 58 |
| 7-1     | 概述 .....                      | 58 |
| 7-2     | break 语句 .....                | 58 |
| 7-3     | continue 语句 .....             | 59 |
| 7-4     | break 和 continue 的流程图比较 ..... | 60 |
| 7-5     | 几种循环的比较 .....                 | 61 |
| 7-6     | 本学时总结 .....                   | 63 |
| 7-7     | 实战练习 .....                    | 63 |
| 第 8 学时  | 函数 .....                      | 65 |
| 8-1     | 函数的定义 .....                   | 65 |
| 8-2     | 函数定义的一般形式 .....               | 66 |
| 8-3     | 函数的调用方式 .....                 | 68 |
| 8-4     | 函数参数 .....                    | 69 |
| 8-5     | 函数的值 .....                    | 70 |
| 8-6     | 本学时总结 .....                   | 71 |
| 8-7     | 实战练习 .....                    | 71 |
| 第 9 学时  | 函数的嵌套和递归调用 .....              | 75 |
| 9-1     | 函数的嵌套调用 .....                 | 75 |
| 9-2     | 函数的递归调用 .....                 | 76 |
| 9-3     | 变量的存储类型 .....                 | 78 |
| 9-4     | 本学时总结 .....                   | 81 |
| 9-5     | 实战练习 .....                    | 81 |
| 第 10 学时 | 数组的定义和使用 .....                | 83 |
| 10-1    | 一维数组的定义与引用 .....              | 83 |
| 10-2    | 二维数组与多维数组的定义与引用 .....         | 85 |
| 10-3    | 本学时总结 .....                   | 88 |
| 10-4    | 实战练习 .....                    | 88 |

|         |                 |     |
|---------|-----------------|-----|
| 第 11 学时 | 字符数组和字符串.....   | 92  |
| 11-1    | 字符数组的定义与引用..... | 92  |
| 11-2    | 字符数组的引用范例.....  | 94  |
| 11-3    | 字符数组的输入/输出..... | 94  |
| 11-4    | 字符串处理函数.....    | 96  |
| 11-5    | 本学时总结.....      | 101 |
| 11-6    | 实战练习.....       | 101 |
| 第 12 学时 | 预处理命令.....      | 102 |
| 12-1    | 宏定义.....        | 102 |
| 12-2    | “文件包含”处理.....   | 110 |
| 12-3    | 条件编译.....       | 111 |
| 12-4    | 本学时总结.....      | 113 |
| 第 13 学时 | 指针.....         | 115 |
| 13-1    | 指针的定义.....      | 115 |
| 13-2    | 指针的使用.....      | 116 |
| 13-3    | 本学时总结.....      | 120 |
| 13-4    | 实战练习.....       | 120 |
| 第 14 学时 | 指针和数组.....      | 124 |
| 14-1    | 指向数组的指针.....    | 124 |
| 14-2    | 指针运算和类型.....    | 127 |
| 14-3    | 本学时总结.....      | 129 |
| 14-4    | 实战练习.....       | 129 |
| 第 15 学时 | 指针和字符串.....     | 134 |
| 15-1    | 字符串的表达形式.....   | 134 |
| 15-2    | 字符串指针.....      | 135 |
| 15-3    | 本学时总结.....      | 136 |
| 15-4    | 实战练习.....       | 136 |
| 第 16 学时 | 函数和指针.....      | 141 |
| 16-1    | 函数的指针.....      | 141 |
| 16-2    | 函数指针作函数参数.....  | 143 |
| 16-3    | 返回指针值的函数.....   | 144 |
| 16-4    | 本学时总结.....      | 146 |
| 16-5    | 实战练习.....       | 146 |
| 第 17 学时 | 指向指针的指针.....    | 149 |
| 17-1    | 指向指针的指针定义.....  | 149 |
| 17-2    | 本学时总结.....      | 151 |
| 17-3    | 实战练习.....       | 151 |
| 第 18 学时 | 结构体.....        | 157 |
| 18-1    | 结构体定义.....      | 157 |



|         |                    |     |
|---------|--------------------|-----|
| 18-2    | 结构类型变量的说明 .....    | 158 |
| 18-3    | 结构体的引用 .....       | 159 |
| 18-4    | 结构体的初始化 .....      | 160 |
| 18-5    | 结构数组 .....         | 160 |
| 18-6    | 本学时总结 .....        | 162 |
| 18-7    | 实战练习 .....         | 162 |
| 第 19 学时 | 结构体指针和链表 .....     | 167 |
| 19-1    | 指向结构变量的指针 .....    | 167 |
| 19-2    | 指向结构数组的指针 .....    | 169 |
| 19-3    | 结构指针变量作函数参数 .....  | 170 |
| 19-4    | 动态存储分配 .....       | 171 |
| 19-5    | 链表的简单操作 .....      | 173 |
| 19-6    | 本学时总结 .....        | 177 |
| 19-7    | 实战练习 .....         | 177 |
| 第 20 学时 | 联合体、枚举及类型说明 .....  | 179 |
| 20-1    | 联合的定义 .....        | 179 |
| 20-2    | 联合变量的说明 .....      | 180 |
| 20-3    | 联合变量的赋值和使用 .....   | 180 |
| 20-4    | 枚举 (enum) 类型 ..... | 182 |
| 20-5    | 枚举类型变量的赋值和使用 ..... | 183 |
| 20-6    | 类型说明 .....         | 184 |
| 20-7    | 本学时总结 .....        | 185 |
| 第 21 学时 | 位运算及位段 .....       | 186 |
| 21-1    | 位运算符和位运算 .....     | 186 |
| 21-2    | 位段的定义 .....        | 188 |
| 21-3    | 位段的用法 .....        | 190 |
| 21-4    | 本学时总结 .....        | 191 |
| 21-5    | 实战练习 .....         | 192 |
| 第 22 学时 | 常见错误和程序调试 .....    | 195 |
| 22-1    | 错误分析 .....         | 195 |
| 22-2    | 程序调试 .....         | 201 |
| 22-3    | 本学时总结 .....        | 201 |
| 22-4    | 实战练习 .....         | 201 |
| 第 23 学时 | 文件 .....           | 203 |
| 23-1    | 文件的概述 .....        | 203 |
| 23-2    | 文件打开和关闭函数 .....    | 204 |
| 23-3    | 文件读写函数 .....       | 206 |
| 23-4    | C库文件 .....         | 214 |
| 23-5    | 本学时总结 .....        | 215 |
| 23-6    | 实战练习 .....         | 215 |

|                          |     |
|--------------------------|-----|
| 第 24 学时 综合例题.....        | 218 |
| 附录 A 编程风格及常见编译错误信息 ..... | 229 |
| A-1 编程风格 .....           | 229 |
| A-2 编译错误信息.....          | 231 |



# 第 1 学时 C 语言概述

随着计算机的普及和发展，C 语言在各个领域的应用越来越广泛。几乎各类计算机都支持 C 语言的开发环境，这为 C 语言的普及和应用奠定了基础。

## 本学时导读

- C 语言的历史背景及特点
- 一个简单的 C 程序
- Turbo C 2.0 集成开发环境
- 本学时总结
- 实战练习

## 1-1 C 语言的历史背景

C 语言是国际上广泛流行的计算机语言，既可以写系统软件，也可以用来写应用软件。C 语言是一种编译性程序设计语言，它与 UNIX 操作系统紧密地联系在一起。UNIX 系统是通用的、交互式的计算机操作系统，它诞生于 1969 年，是由美国贝尔实验室的 K.Thompson 和 D.M.Ritchie 用汇编语言开发成功的。

C 语言的前身是 BCPL 语言。1967 年，英国剑桥大学的 Martin Richard 推荐 BCPL (Basic Combined Programming Language)。1970 年，贝尔实验室的 K.Thompson 以 BCPL 为基础，开发了 B 语言，并用 B 语言编写了 UNIX 操作系统。1972 年，贝尔实验室的 D.M.Ritchie 在 B 语言的基础上设计出 C 语言。C 语言保持了 BCPL 语言与 B 语言的精练，接近硬件的优点，又克服了它们过于简单的缺点。1973 年，K.Thompson 和 D.M.Ritchie 合作把 90% 以上 UNIX 用 C 语言改写，并加进了多道程序的功能，称为 UNIX 第五版，开创了 UNIX 系统发展的新局面。

直到 1975 年 UNIX 第六版公布后，C 语言的突出的优点才引起人们的普遍注意。1977 年，出现了不依赖于 C 语言的编译文本《可移植 C 语言编译程序》，使 C 移植到其他程序时所需的工作大幅简化，也推动了 UNIX 操作系统的普及。随着 UNIX 的日益广泛使用，C 语言也得以迅速推广。C 语言和 UNIX 可以说是一对孪生兄弟，在发展过程中相辅相成。1978 年以后，C 语言已先后移植到大、中、小、微型机上，已独立于 UNIX 和 PDP。现在，C 语言已风靡全世界，成为世界上应用最广泛的几种计算机语言之一。

C 语言问世以来，相继出现的各种版本对其进行了发展和扩充，基于这一点，美国国家标准化协会 (ANSI) 于 1983 年制定了新的标准，称为 ANSI C。ANSI C 比原来的标准 C 有了很大的发展。K&R 在 1988 年修改了他们的经典著作 *The C Programming Language*，按照 ANSI C 标准重写了该书。1987 年，ANSI 又宣布了新标准——87 ANSI C 为 ISO C 的标准。目前，流行的 C 编译系统都是以它为基础。本书的叙述基本上以 ANSI C 为基础。目前，广泛流行的各种版本 C 语言编译系统虽然基本部分是相同的，但也有一些不同。在微型机上使用的有 Microsoft C、Turbo C、Quick C、



BORLAND C 等，它们的不同版本又略有差异。因此，本书主要介绍 Turbo C（以下简称 TC）。

### 1-2 一个简单的 C 程序

【例 1-1】相关的程序代码见光盘文件“第 1 学时\1-1.C”。

```
main()                /*求两数之和*/
{
    int a,b,sum;        /*定义变量*/
    a=123; b=456;        /*以下 3 行为 C 语句*/
    sun=a+b;
    printf("sum is %d\n",sum);
}
```

以上是一个关于求和的简单程序，在 TC 环境中的运行结果如图 1-1 所示。

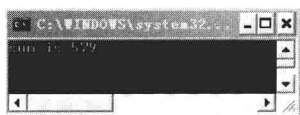


图 1-1 程序结果

图 1-1 中给出了一个求和的结果，下面看一下在 Turbo C 2.0 中的具体实现过程，操作步骤如下：

**STEP 1** 双击桌面上的 TC 图标或者从“开始”菜单的“所有程序”子菜单中选择 TC 程序，进入 Turbo C 的集成环境。TC 环境将会自动建立一个名称为 NONAME.C 的文件。最上面一行显示当前光标所在的位置及生成的文件名，如图 1-2 所示。

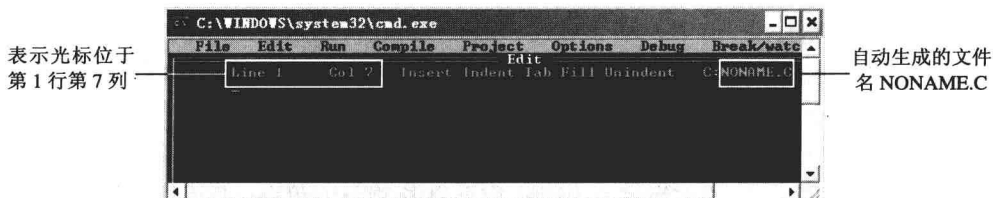


图 1-2 Turbo C 2.0

**STEP 2** 在编辑区填写代码。首先，将 main 函数写在程序的首部，本例中 main 函数中没有参数和返回值，其中定义 3 个变量 a、b 和 sum，如图 1-3 所示。

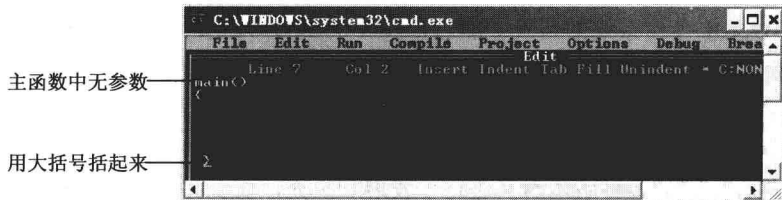


图 1-3 没有参数和返回值 main 函数



**注意** main 是主函数的函数名，表示这是一个主函数。每一个 C 源程序都必须有一个主函数（main 函数），函数体有大括号{}括起来。一个 C 程序总是从 main 函数开始执行，而不论 main 函数在程序的什么位置。

**STEP3** 填写 main 函数体。首先，定义需要的变量，而后对变量进行赋值并运算，如图 1-4 所示。

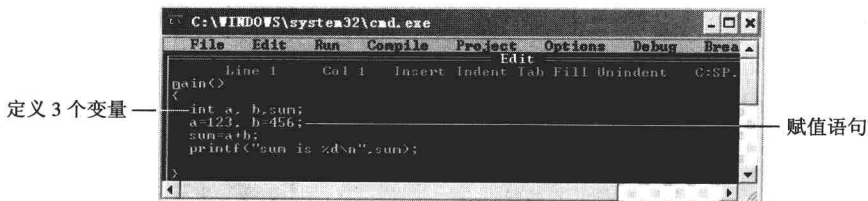


图 1-4 填写 main 函数



- ① 变量必须在它被定义以后才可以使用。
- ② 在 C 语言中赋值语句应该使用“=”，在这里不表示等号，而是赋值符号。C 语言中的等号为“==”，两个“=”相连。
- ③ 在 C 语言中，每个语句和数据定义的最后必须有一个分号。分号是 C 语句的必要组成部分。

**STEP4** 利用 printf 函数进行输出，如图 1-5 所示。printf 函数是 C 语言定义的标准输出函数，它用于向终端输出多个各种类型和数据格式的数据，基本格式是：  
printf(<格式控制字符串>,<输出表达式列表>);  
其用法在后面会进行详细介绍。

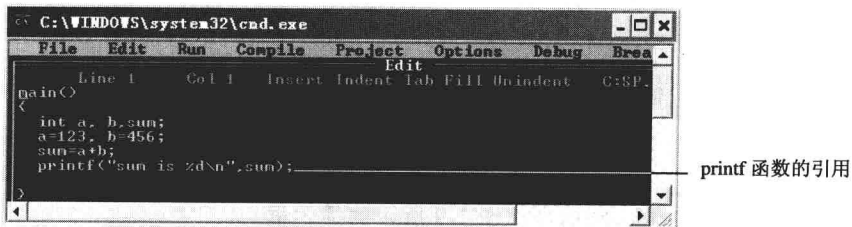


图 1-5 printf 函数的使用

**STEP5** 编辑好之后，进行保存。按【F10】键选中主菜单，利用【←】、【→】键选中 File 菜单，按【Enter】键打开 File 菜单，通过【↑】、【↓】键选择 Save 命令，按【Enter】键，然后输入文件保存的目录并将文件命名为 SP.C。也可以直接按【F2】键进行保存，如图 1-6 所示。

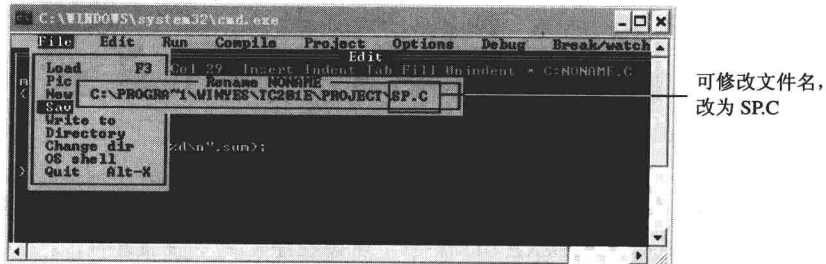
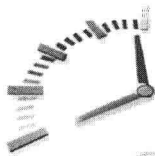


图 1-6 保存文件



**STEP6** 保存后就可以进行编译和连接,按【F10】键选中主菜单,利用【←】、【→】键选中 Compile 菜单,通过【↑】、【↓】键选择 Compile to OBJ 命令编译程序(见图 1-7),然后按【Enter】键执行。完成后会生成 SP.OBJ 文件,为连接程序做准备。

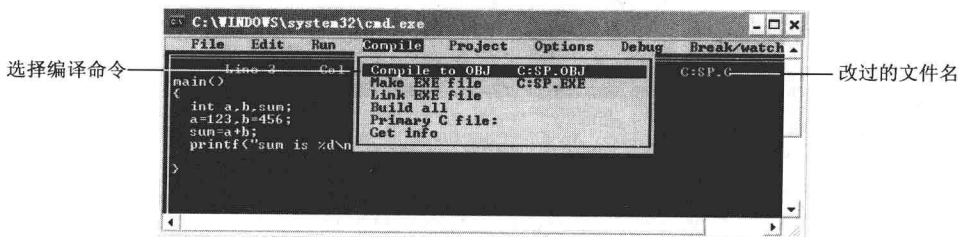


图 1-7 在 Turbo C 2.0 中编译

**STEP7** 如果没有错误,则编译成功,如图 1-8 所示。

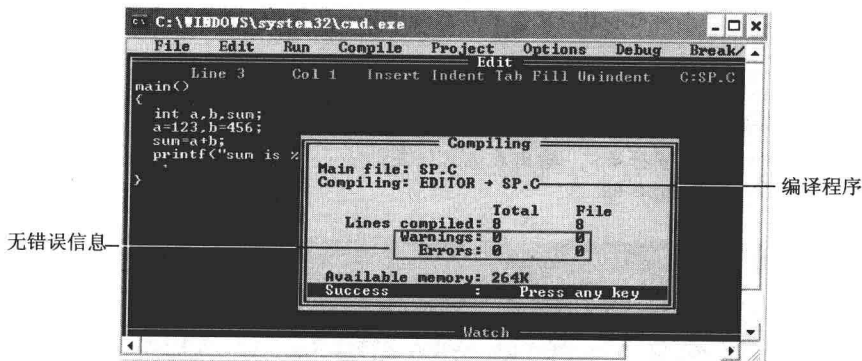


图 1-8 编译通过

**STEP8** 如果有错误,则出现如图 1-9 所示的提示框。

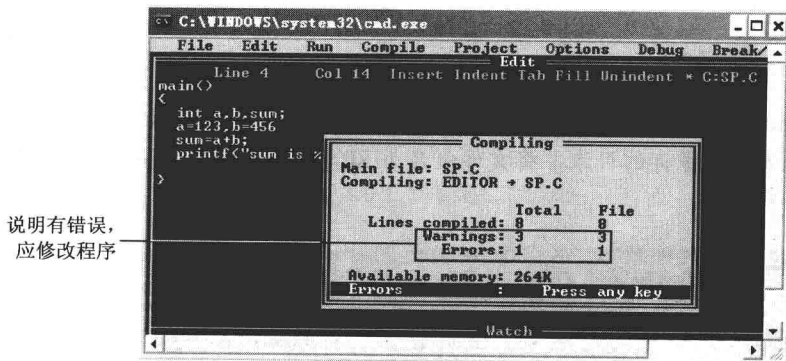
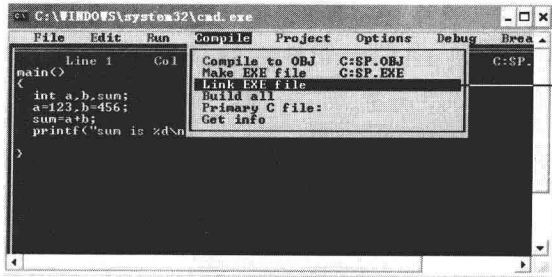


图 1-9 编译未通过(有错误)

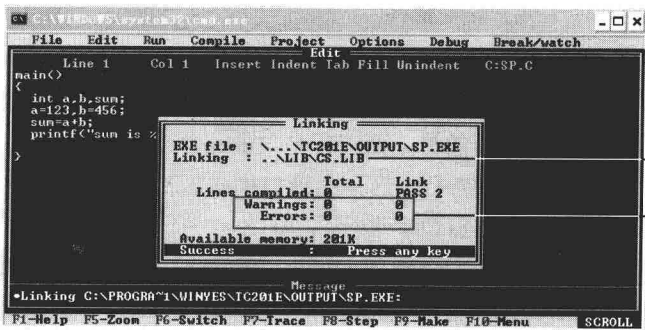
**STEP9** 按【F10】键,选中主菜单,利用【←】、【→】键选中 Compile 菜单,通过【↑】、【↓】键选择 Link EXE file 命令,连接程序。或者按【Alt+C】组合键,选择 Compile 菜单,通过【↑】、【↓】键选择 Link EXE file 命令,如图 1-10 所示,连接程序生成可以执行的.EXE 程序。



选择连接命令

图 1-10 连接过程

**STEP10** 如果程序正确且连接成功，则如图 1-11 所示；如果有错，则需要改正，继续编译后连接。



连接程序

无错误和警告

图 1-11 连接通过

**STEP11** 编译连接成功后就可以运行该程序，按【Alt+R】组合键打开 Run 菜单，通过键盘上的【↑】、【↓】键来选择 Run 命令（见图 1-12），再按【Enter】键确认。也可以直接按【Ctrl+F9】组合键，运行程序。

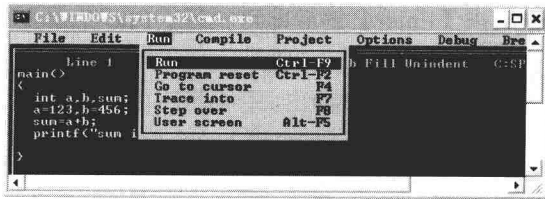


图 1-12 在 Turbo C 2.0 中运行程序

**STEP12** 按【Enter】键后，对于不需要输入原始数据的程序而言，将会显示编译好的源代码，如本例的情况，如图 1-13 所示。而对于需要输入数据的情况在后面会见到。

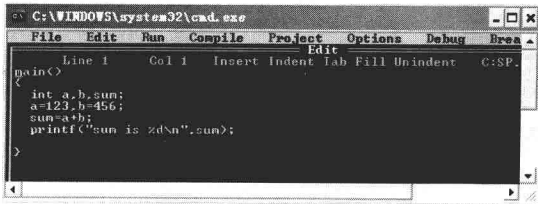


图 1-13 在 Turbo C 2.0 中运行结果