



21世纪全国高等院校计算机教育“十一五”规划教材
丛书主编 全国高等学校计算机教育研究会课程与教材建设委员会主任 李大友

C语言程序设计

主 编	邹修明	马国光
副主编	罗德林	孙红敏
	燕居怀	常雪琴



中国计划出版社

21世纪全国高等院校计算机教育“十一五”规划教材

C 语言程序设计

本书编委会 编著

中国计划出版社

图书在版编目 (C I P) 数据

C语言程序设计 / 《C语言程序设计》编委会编著. —北京: 中国计划出版社, 2007. 8
21世纪全国高等院校计算机教育“十一五”规划教材
ISBN 978-7-80177-966-3

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

中国版本图书馆CIP数据核字 (2007) 第105686号

内 容 简 介

本书结构由易到难, 讲解深入浅出, 并通过大量的实例讲解了C语言程序设计的方法, 主要包括绪论、数据类型及其运算、结构化程序设计、选择结构程序设计、循环结构程序设计、数组、函数、编译预处理、结构体与共用体、指针、位运算及文件等。

本书既可作为高等院校相关课程的教材, 也可作为高职高专、培训机构的教学用书。

21世纪全国高等院校计算机教育“十一五”规划教材

C 语言程序设计

本书编委会 编著

☆

中国计划出版社出版

(地址: 北京市西城区木樨地北里甲11号国宏大厦C座4层)

(邮政编码: 100038 电话: 63906433 63906381)

新华书店北京发行所发行

河北省高碑店市鑫宏源印刷厂印刷

787×1092毫米 1/16 20.5印张 499千字

2007年8月第一版 2007年8月第一次印刷

印数1—5000册

☆

ISBN 978-7-80177-966-3

定价: 29.00元

丛书编委会

主 任：李大友

副主任：王行言 郑 莉

委 员：（按音序排列）

樊金生 冯春辉 高延武 韩金仓 刘凤田

刘 云 刘建臣 刘三满 罗德茹 彭宣戈

齐玉斌 孙晨霞 王潜平 王书海 姚 华

杨晓斌 张广斌 赵建明 赵连胜 邹修明

本书编委会

主 编：邹修明 马国光

副主编：罗德林 孙红敏 燕居怀 常雪琴

参 编：闫宗梅 王玉见 陈思思 李金凤

许丽娟 王秀娟 李 晶

丛 书 序

随着我国高等教育发展与改革的逐步深化，越来越多的高等院校将其自身定位于工程型或应用型，立足于培养能够满足各行各业需求的，素质高、能力强的应用型专业人才。与此同时，由于信息化是当今社会与经济必然趋势，因而应用信息技术的能力亦将会成为衡量人才水平的重要标尺。由此可见，培养既具有专业知识，又拥有良好信息技术应用能力的人才，是现今高等教育发展与改革的目标之一。

目前，教育部有关计算机教育的教学指导委员会、全国高等学校计算机教育研究会等学术团体、各高等院校的专家学者已经在计算机教学与教材改革方面做了大量的工作，许多一线教师已经在计算机教学和科研方面积累了许多宝贵经验。在这些条件下，通过将其教研成果汇总并转化为教材的形式向全国各高等院校推广，对于促进高等院校计算机教育的发展与改革，培养应用型专业人才，是一件十分有意义的事情。

鉴于以上情况，中国计划出版社与全国高等学校计算机教育研究会决定联合策划组织、编写出版了本套“21 世纪全国普通高等院校计算机教育‘十一五’规划教材”。为实施精品战略，出版社与全国高等学校计算机教育研究会在全国范围内进行了系统、详细的调查，对各层各类教学指导性文件进行了认真、深入的研究，对国内外已出版同类教材进行了客观、理性的分析，组织专家学者、一线教师及企业人员展开研讨，以期打造切实符合实际教学需求的精品教材。

为配合各学校的精品课程建设工程，本套教材以国家级精品课程指标为指引方向，借鉴其他兄弟出版社的先进经验和成功案例，提出了建设“立体化教学资源平台”的概念，其内容包括教材、教学辅导资料、教学资源包、网络平台等内容，并将在后续培训、论文发表等多方面满足教师与精品课程建设的需求。

本套教材具有以下特点。

1. 定位明确，应用为本

本套教材定位于高等院校学生计算机应用能力的培养，不仅要使学生理解计算机相关的基本理论与基本知识，还要使学生掌握利用计算机解决实际问题的能力。要使学生面对一个实际问题时，不仅要知其然，还要知其所以然，更要会其如何然，最终，要具备实际操作应用能力。

2. 案例驱动，能力培养

本套教材通过从实际应用中提炼出的案例来辅助知识的讲授与能力的培养，在案例设计时从其科学性、实用性及开放性出发，尽量营造贴近实际应用的环境，激发学生的学习兴趣，从而提高教学效率，提高学生的实际应用能力。

3. 资源丰富，便于教学

我社免费为选用本套教材中图书的教师提供如下资源服务：

- 多媒体电子课件（PowerPoint 格式）
- 所有案例的相关素材（图片、声音与源程序等）与最终结果

- 所有习题的素材与答案
- 两套模拟测试题及答案
- 不定期组织教师培训

高等院校计算机教育的发展与改革不会停止，各院校的实际情况又有所不同，我们恳请各位老师在使用过程中提出批评与建议，以便及时改进教材欠妥与不足之处，使本套教材日趋完善。

我们相信在各位专家学者与一线教师的支持与帮助下，本套教材一定能成为特点鲜明、质量上乘的精品教材，同时，我们也希望通过本套教材的出版为高等院校计算机教育的发展与改革做出自己的一份贡献。

丛书编委会

前 言

程序设计语言是高等院校计算机专业的基础课程，也是非计算机专业计算机应用教育的重要课程，它已成为高等院校基础教学的重要组成部分。C 语言作为程序设计语言中的基础语言，在高等院校计算机专业教学中占据着重要地位。

本书以 C 语言的数据类型为主线编排，具有以下特点：

(1) 突出 C 语言实用的重点概念，加强程序阅读、编写和调试能力。本书不追求 C 语言的语法全面性，而是较深入地研究实用的重点概念。

(2) 力求与 C++ 兼容。具有对象程序设计功能的 C++，代表了程序设计语言的发展方向。从 C 到 C++ 是程序设计语言发展的必然趋势。本书从语法和程序结构两方面都力求与 C++ 兼容。书中例题的程序全在 C++ 6.0 系统上调试和运行，也可在 Turbo C 2.0 系统上运行。

(3) 追求可读性和可理解性。本书遵守循序渐进的原则，突出重点内容，运用文字、插图和测试相结合的方式，将抽象概念形象化，具有真实感，便于读者理解、掌握。

(4) 加强程序风格训练，提高程序的可读性。全书程序均从源程序复制过来，具有良好的程序风格，可作为读者编程和上机实践的样本。

本书结构由易到难，讲解深入浅出，并通过大量的实例讲解了 C 语言程序设计的方法，主要内容包括绪论、数据类型及其运算、结构化程序设计、选择结构程序设计、循环结构程序设计、数组、函数、编译预处理、结构体与共用体、指针、位运算及文件等。

本书既可作为高等院校相关课程的教材，也可作为高职高专、培训机构的教学用书。

本书由邹修明、马国光主编，罗德林、孙红敏、燕居怀、常雪琴担任副主编，其中第 1、2、3、4 章由邹修明编写，第 5、6 章由马国光编写，第 7 章由罗德林编写，第 8、9 章由孙红敏编写，第 10 章由燕居怀编写，第 11、12 章由常雪琴编写，闫宗梅、王玉见、陈思思、李金凤、许丽娟、王秀娟、李晶参与了本书大部分章节的编写，并进行了代码调试等具体工作。

由于时间仓促与编者水平有限，不足与欠妥之处在所难免，恳请广大读者不吝指正。

编者

2007年6月

目 录

第1章 绪论	1
1.1 程序设计语言	1
1.1.1 程序设计语言的发展	1
1.1.2 C语言的发展及特点	2
1.2 程序运行环境	2
1.2.1 Turbo C 3.0的安装	2
1.2.2 简单程序的编写和运行过程	3
1.3 C程序介绍	7
1.3.1 程序	8
1.3.2 程序的结构特点	10
1.3.3 C语言的词汇	10
1.4 小结与提高	11
1.5 思考与练习	11
第2章 数据类型及其运算	13
2.1 C语言的数据类型	13
2.2 常量与变量	14
2.2.1 常量	14
2.2.2 变量	17
2.2.3 变量存储	19
2.2.4 类型转换	21
2.3 运算符和表达式	22
2.3.1 算术运算符和算术表达式	22
2.3.2 赋值运算符和赋值表达式	25
2.3.3 逗号运算符和逗号表达式	26
2.3.4 运算符的优先级别	27
2.4 输入/输出函数	27
2.4.1 格式输入函数	28
2.4.2 格式输出函数	30
2.5 小结与提高	34
2.6 思考与练习	34
第3章 结构化程序设计	39
3.1 结构化程序设计概述	39

3.1.1 自上而下分析设计问题	40
3.1.2 模块化程序设计	40
3.2 C语句概述	41
3.2.1 C语句分类	41
3.2.2 赋值语句	43
3.3 程序的三种基本结构	43
3.3.1 顺序结构	44
3.3.2 选择结构	44
3.3.3 循环结构	45
3.4 字符输入/输出函数	45
3.4.1 字符输入函数	46
3.4.2 字符输出函数	48
3.5 顺序程序举例	49
3.6 小结与提高	52
3.7 思考与练习	52
第4章 选择结构程序设计	56
4.1 关系运算符与关系表达式	56
4.1.1 关系运算符及其优先次序	56
4.1.2 关系表达式	57
4.2 逻辑运算符与逻辑表达式	58
4.2.1 逻辑运算符及其优先次序	58
4.2.2 逻辑表达式	58
4.3 if语句	60
4.3.1 if语句的三种形式	60
4.3.2 if语句的嵌套	64
4.3.3 条件运算符	67
4.4 switch语句实现多分支选择结构	67
4.5 程序举例	70
4.6 小结与提高	73
4.7 思考与练习	74
第5章 循环结构程序设计	79
5.1 while和do...while语句	79
5.1.1 while语句	79
5.1.2 do...while语句	84
5.1.3 while语句与do...while语句比较	88
5.2 for语句	89
5.3 continue语句和break语句	92
5.3.1 continue语句	92

5.3.2 break语句	93
5.4 循环嵌套	94
5.5 程序举例	96
5.6 goto语句	100
5.7 小结与提高	101
5.8 思考与练习	101
第6章 数组	107
6.1 一维数组的定义和引用	107
6.1.1 一维数组的定义	107
6.1.2 一维数组的引用	108
6.1.3 一维数组的初始化	109
6.1.4 一维数组的程序举例	111
6.2 二维数组的定义和引用	113
6.2.1 二维数组的定义	114
6.2.2 二维数组的引用	114
6.2.3 二维数组的初始化	115
6.2.4 二维数组的程序举例	116
6.3 字符串的处理	119
6.3.1 字符数组的定义及引用	119
6.3.2 字符串和字符串的结束标志	120
6.3.3 字符串处理函数	121
6.3.4 字符串程序举例	125
6.4 小结与提高	127
6.5 思考与练习	128
第7章 函数	133
7.1 函数概述	133
7.2 函数定义的一般形式	134
7.3 函数的调用	136
7.3.1 函数调用的一般形式	136
7.3.2 函数调用的方式	136
7.3.3 对被调函数的声明	137
7.4 函数参数和函数的值	138
7.4.1 形参和实参	138
7.4.2 函数的返回值	140
7.4.3 标准库函数举例	141
7.5 函数的嵌套调用	142
7.6 函数的递归调用	143
7.6.1 递归调用的概念	143

7.6.2	递归调用的执行过程	144
7.6.3	递归调用函数应用举例	145
7.7	数组作函数的参数	146
7.8	局部变量与全局变量	150
7.8.1	局部变量	150
7.8.2	全局变量	152
7.9	变量的存储类别	154
7.9.1	动态存储方式与静态存储方式	154
7.9.2	用auto声明局部变量	154
7.9.3	用static声明局部变量	155
7.9.4	register变量	156
7.9.5	用extern声明外部变量	157
7.9.6	用static声明全局变量	159
7.10	内部函数与外部函数	159
7.11	小结与提高	161
7.12	思考与练习	162
第8章	编译预处理	168
8.1	宏定义	168
8.1.1	不带参数的宏定义	168
8.1.2	带参数的宏定义	171
8.1.3	取消宏定义	174
8.2	文件包含	175
8.3	条件编译	177
8.4	小结与提高	179
8.5	思考与练习	180
第9章	结构体与共用体	184
9.1	结构体	184
9.1.1	结构体类型的定义	184
9.1.2	结构体变量的定义	186
9.1.3	结构体变量的引用	188
9.1.4	结构体变量的初始化	189
9.1.5	结构体数组	191
9.1.6	结构体作函数的参数	194
9.1.7	结构体举例	196
9.2	共用体	198
9.2.1	共用体类型的定义	198
9.2.2	共用体类型变量的定义	199
9.2.3	共用体变量的引用	200

9.2.4 共用体变量的初始化	200
9.3 枚举类型	202
9.3.1 枚举类型的定义	203
9.3.2 枚举变量的定义和初始化	203
9.3.3 枚举变量举例	204
9.4 用typedef定义类型	206
9.5 小结与提高	208
9.6 思考与练习	210
第10章 指针	215
10.1 地址和指针	215
10.1.1 内存单元地址	215
10.1.2 指针的定义	216
10.2 变量的指针和指向变量的指针变量	217
10.2.1 指针变量的定义	217
10.2.2 指针变量的初始化	218
10.2.3 指针变量的引用	219
10.2.4 指针变量作函数参数	225
10.3 数组的指针和指向数组的指针变量	228
10.3.1 指向一维数组的指针	228
10.3.2 指向二维数组的指针	231
10.3.3 指向字符串的指针变量	234
10.3.4 指向数组的指针变量作函数的参数	237
10.4 函数的指针和指向函数的指针变量	239
10.4.1 用函数的指针变量调用函数	239
10.4.2 用指向函数的指针作函数的参数	240
10.5 返回指针值的函数	242
10.6 指针数组和指向指针的指针	244
10.6.1 指针数组的概念	244
10.6.2 指向指针的指针	249
10.6.3 指针数组作main函数的参数	251
10.7 结构体的指针与指向结构体的指针变量	252
10.7.1 指向结构体的指针	252
10.7.2 动态存储分配	256
10.7.3 链表	258
10.8 小结与提高	264
10.9 思考与练习	265
第11章 位运算	269
11.1 位运算符和表达式	269

11.1.1 按位与	269
11.1.2 按位或	270
11.1.3 按位异或	271
11.1.4 取反	272
11.1.5 左移和右移	273
11.1.6 位运算赋值运算符	274
11.1.7 不同长度的数据进行位运算	274
11.2 位运算举例	274
11.3 位段	275
11.4 小结与提高	277
11.5 思考与练习	278
第12章 文件	280
12.1 文件概述	280
12.2 文件类型指针	281
12.3 文件的打开与关闭	282
12.3.1 文件的打开	282
12.3.2 文件的关闭	284
12.4 文件的读/写	284
12.4.1 fputc函数和fgetc函数	284
12.4.2 字符串读/写函数fgets和fputs	286
12.4.3 fprintf函数和fscanf函数	288
12.4.4 fread函数和fwrite函数	290
12.5 文件的定位	293
12.5.1 rewind函数	293
12.5.2 fseek函数和随机读写	294
12.5.3 ftell函数	296
12.6 小结与提高	297
12.7 思考与练习	298
附录 I 常用字符与ASCII代码对照表	301
附录 II C语言中的关键字	302
附录 III 运算符与结合性	303
附录 IV C语言库函数	305
主要参考文献	311

第 1 章

绪论

本章从计算机程序设计语言（Programming Language）着手，介绍程序设计语言的发展，以及程序的编辑、编译及运行。熟悉C语言程序的上机环境，是学习C语言程序设计的基础。



本章主要内容

- 程序设计语言
- 程序运行环境
- C程序介绍

1.1 程序设计语言

所谓程序就是指令代码的有序集合，人们为了针对某些实际问题而编制程序，实现某些特定的功能。一台计算机若没有装入任何软件，就称之为裸机，裸机只能识别“0”和“1”两种代码，早期的程序设计者只能用二进制代码表示的机器指令来编写程序。程序设计语言就是用户用来编写程序的语言，它是人与计算机进行信息交流的工具。对于理想的设计语言来说，所提供的语法应该能够满足描述算法、数据结构等方面的要求。程序设计语言按其发展阶段分为：机器语言、汇编语言和高级语言。

1.1.1 程序设计语言的发展

把计算机指令的集合称为机器语言（Machine Language），用机器语言编制的程序称为机器语言程序。机器语言编写的程序是由二进制代码组成的代码序列，它是唯一不需要经过翻译解释就能被机器直接识别的语言。它与具体机器无关，但需程序设计员熟记大量的指令代码，易出错，大大降低了程序的通用性，而且所编写的程序只能在相同的硬件环境下使用，程序的可移植性差。

为了摆脱用机器指令代码编写程序的困难，出现了用指令符号编制程序的方法。编制该程序时只要记住指令的助记符就可以了，这种指令助记符是指令英文名称的缩写，因而比指令的编码容易记忆，这种指令符号的扩大就是汇编语言。用汇编语言编制的程序，仍要记住机器指令的助记符，计算机是不能直接执行的，且所编的程序只针对某一类机器。

为解决这些问题，出现了用高级语言来编写程序。

高级语言编写的程序如同汇编语言一样，计算机是不能直接执行的，必须将源程序（输入的程序代码）经过“翻译”生成目标程序（机器语言程序）才能执行。只是高级语言程序是由预先存放在机器中的“解释程序”或“编译程序”来完成这一“翻译”工作的。

综上所述，计算机语言分为低级语言和高级语言两大类。低级语言包括机器语言和汇编语言，它们都是面向机器的语言，用这种语言编制的程序只适用于某种特定类型的计算机。高级语言又可分为面向过程的语言和面向问题的语言，前者在编程时不仅要告诉机器“做什么”，而且要告诉机器“怎么做”；后者只要告诉机器“做什么”，也常称之为人工智能语言。

1.1.2 C 语言的发展及特点

C 语言是广泛流行的计算机高级语言的一种，既可用于编写系统软件，也可用来编写应用软件。它是一种面向过程的语言，既具有低级语言的某些特性（例如，对内存地址的操作、位操作等），又具有高级语言的特点。

C 语言是在 B 语言的基础上发展起来的，后来经过多次改进，直到 1975 年 UNIX 第 6 版公布后，C 语言的优点才被人们普遍注意。目前流行的各种 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 等，它们不同版本又略有差异，本书所讲的是 Turbo C。

C 语言对程序员要求较高。使用 C 语言编写的程序会感到限制少、灵活性大、功能强，可以编写出任何类型的程序。具有语言简单易懂、运算符、数据结构丰富、具有结构化的控制语句、语法控制不太严格，程序设计自由度大、实现汇编语言的大部分功能、程序执行的效率高、可移植性好等特点。

1.2 程序运行环境

在编好一个 C 语言程序后，如何上机运行呢？通常要经过以下几个步骤：上机输入与编辑源程序→对源程序进行编译→与库函数连接→运行目标程序显示运行结果。

常用的 C 语言程序设计的实验环境有：Turbo C 3.0、Win-T C、Visual C++ 6.0 等。本书采用 Windows XP 操作系统下的 Turbo C 3.0 进行 C 语言程序编辑、编译、运行和调试，Turbo C 3.0 是将所有功能融为一体的集成开发软件，具有速度快、效率高、功能强等优点。在 Windows 操作系统下，使用非常方便，可以进行复制、粘贴、剪切等操作。

1.2.1 Turbo C 3.0 的安装

Turbo C 3.0 安装程序比较小，仅 2.28MB 左右，可以从互联网上下载，要保证计算机安装有解压缩软件，因为 Turbo C 3.0 源安装程序是一个压缩包，如图 1-1 所示。

将 Turbo C 3.0 源安装程序解压到任意盘根目录下，即可使用。本书将其解压到 D 盘下，解压后，到 D 盘根目录下，可以找到 tc3 文件夹，如图 1-2 所示。



TC3



tc3

图 1-1 源安装程序图标

图 1-2 源安装程序解压后图标

双击打开 tc3 文件夹后，会出现四个文件夹，分别是 BGI、BIN、INCLUDE、LIB。双击打开 BIN 文件夹后，可以找到 TC.exe 文件。运行 TC.exe 后，即是 C 语言程序的运行环境，如图 1-3 所示。

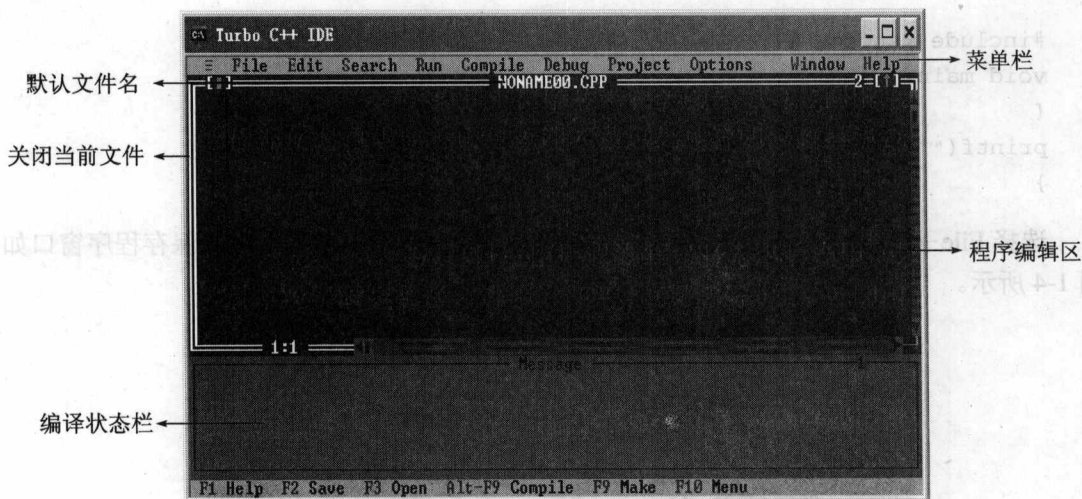


图 1-3 程序运行窗口

注意：可将 BIN 文件夹下的 TC.exe 发送到桌面快捷方式或将 BIN 文件夹下的“TC 指向 MS-DOS 程序的快捷方式”拷贝到桌面上，实现快速启动。

打开窗口后，默认的文件名是 NONAME00.CPP，“菜单栏”包括 File、Edit、Run、Complie、Options、Window 等。编译状态栏是在编译程序时，显示程序编译结果的，主要是检查程序是否存在错误。

1.2.2 简单程序的编写和运行过程

运行一个简单的小程序，首先要将程序输入到程序编辑区中，保存该程序文件后称之为程序源文件。然后将程序源文件进行编译，如上所述是将程序源文件进行编译解释后转换成目标文件（即机器可识别的机器语言）如未发现错误，编程人员就可以在用户界面中显示程序的运行结果。本书采用 Turbo C 3.0 编写程序，它的基本操作与 Windows 应用程序相似：单击进行选择操作命令，按住鼠标左键拖动可以实现程序代码的选择等。除了使用鼠标来完成命令的操作，也可以使用键盘操作来完成相应的命令，如：启动菜单项，可以使用 Alt+字母键。例如：启动 File 菜单，可使用 Alt+F 组合键，菜单项中第一个大写的红色字母就是用键盘启动菜单项的。

Turob C 3.0 使用时可能占满整个屏幕,那么怎样使窗口变小,可以看到其他应用程序或桌面图标呢?可以使用快捷键进行操作:同时按下 Alt+Enter 组合键可实现窗口屏幕缩小,同理再同时按下 Alt+Enter 组合键,可实现窗口变大,占满整个屏幕。下面具体介绍一下 New、Save、Compile、Run、UserScreen 等命令的使用。

1. 新建和保存

选择 File→New 命令后,即可创建一个新的编辑窗口,将以下 C 语言源程序输入到编辑区中。

```
#include <stdio.h>
void main()
{
    printf("*****");
}
```

选择 File→Save 命令或按 F2 键后,可对新建的源文件进行保存,出现保存程序窗口如图 1-4 所示。

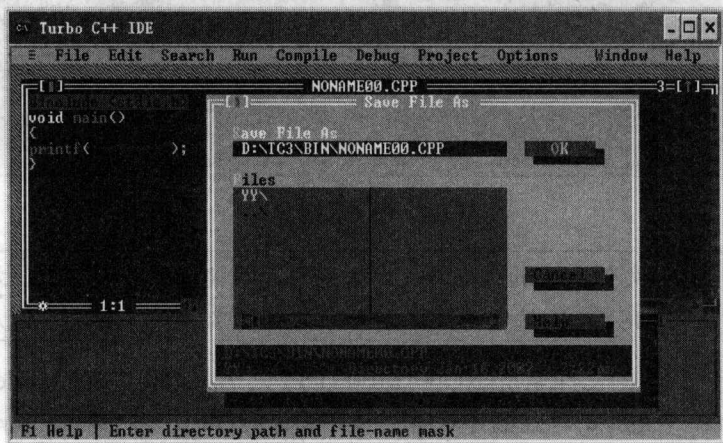


图 1-4 保存源程序窗口

注意: 在保存前,先在 D:\TC3\BIN\ 下建立一个文件夹,以便将所有的源程序保存在该文件夹下。这里建立的文件夹名为 yy。文件名默认的是 NONAME00.CPP,可以给该文件起个相应的文件名,以便以后的查找,这里以 1.CPP 保存。

2. 编译

将输入好的源程序进行编译,翻译成机器能够识别的机器语言。可通过选择 Compile→Compile 命令或按 Alt+F9 组合键来实现,如图 1-5 所示。
编辑好的程序,如未发生错误输入或程序语法上的错误,应该出现编译成功的界面,这里包括文件的基本信息:文件路径、编译行数、警告所在的行数及错误所在的行数。