



21st CENTURY
规划教材

面向21世纪高等院校计算机系列规划教材

COMPUTER COURSES FOR UNDERGRADUATE EDUCATION

C语言程序设计

王晓勇	张文祥	胡天云	主 编
肖四友	陈成钢	郝 婕	副主编



科学出版社

www.sciencep.com



面向21世纪高等院校计算机系列规划教材
COMPUTER COURSES FOR UNDERGRADUATE EDUCATION

C 语言程序设计

王晓勇 张文祥 胡天云 主 编
肖四友 陈成钢 郝 婕 副主编

科学出版社

北 京

内 容 简 介

本书共 12 章, 主要内容包括: C 语言基础知识、简单的 C 程序设计、控制结构、数组、函数、编译预处理、指针、结构体与共用体、文件及 C 综合程序设计案例等。

本书适合作为高等院校计算机语言基础教材, 也可供 C 语言初学者学习使用。

图书在版编目(CIP)数据

C 语言程序设计/王晓勇, 张文祥, 胡天云主编. —北京: 科学出版社, 2007

(面向 21 世纪高等院校计算机系列规划教材)

ISBN 978-7-03-018566-2

I. C… II. ①王… ②张… ③胡… 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 019650 号

责任编辑: 陈晓萍 李太铎/责任校对: 刘彦妮

责任印制: 吕春珉/封面设计: 飞天创意

科学出版社出版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

信浩彩色印装有限公司印刷

科学出版社发行 各地新华书店经销

*

2007 年 2 月第 一 版 开本: 787×1092 1/16

2007 年 2 月第一次印刷 印张: 19

印数: 1—2 600 字数: 430 000

定价: 26.00 元

(如有印装质量问题, 我社负责调换<路通>)

销售部电话 010-62136131 编辑部电话 010-62138978-8003

前 言

“C 语言程序设计”是高等院校普遍开设的一门计算机基础课程，通过该课程的学习，使学生掌握一种编程语言，训练学生的编程思维，使学生具备基本程序设计能力，为学生进一步学习与程序设计相关的知识打下基础。C 语言是一种通用的程序设计语言，既具有高级语言的特性，又具有直接操纵计算机硬件的能力，并以其丰富灵活的控制和数据结构、简洁而高效的语句表达、清晰的程序结构和良好的可移植性等特点使学习者爱不释手。

本书是结合作者多年的教学实践经验而编写，适合学生的学习，全面提高学生的综合素质。

本书系统介绍了 C 语言的语法知识，全书共分为 12 章。在第 1 章中，在介绍 C 语言基本语法知识的同时，特别介绍了目前 C 编程人员较喜欢使用的 Win-TC 开发环境。全书各章节以实例导入，更有利于学生对 C 语言基本语法、程序结构、算法等知识的掌握；特别是在最后一章给出两个 C 语言综合程序设计案例，分析了案例的设计思路，并给出其中一个案例的完整程序代码，通过对算法及程序功能模块的分析，使学生了解小型完整程序的设计过程。

与本书配套的《C 语言程序设计实训教程》可以作为“C 语言程序设计”课程的上机实验指导及课后自测练习用书。通过综合自测练习，使学生加深对 C 语言基本语法的理解与记忆以及对算法的理解。

本书由主编提出编写思路、编写大纲，由多位教师合作完成。第 1、2 章由肖四友编写，第 3~5 章由郝婕和胡天云编写，第 6、7 章由王晓勇编写，第 8、10 章由张文祥编写，第 9、11、12 章由陈成钢编写。附录部分由王晓勇、郝婕整理。最后，全书由王晓勇、张文祥审稿、定稿。

由于编写时间仓促，书中不足与疏漏之处在所难免，恳请广大读者批评指正。

编 者

2006 年 12 月

目 录

第1章 C语言概述	1
1.1 C语言概述	2
1.1.1 C语言简介及特点	2
1.1.2 C语言源程序结构	3
1.2 Win-TC 环境简介	6
1.2.1 Win-TC 的安装与使用	6
1.2.2 在 Win-TC 中编辑及执行 C 源程序	8
1.2.3 Win-TC 环境的特色	12
本章小结	13
习题	13
第2章 数据类型、运算符与表达式	14
2.1 C 的数据类型	15
2.2 常量与变量	16
2.3 C 的基本数据类型	19
2.3.1 整数数据	19
2.3.2 实型数据	21
2.3.3 字符型数据	22
2.3.4 符号常量	25
2.4 C 的运算符与表达式	26
2.4.1 C 的运算符简介	26
2.4.2 算术运算符与算术表达式	26
2.4.3 赋值运算符与赋值表达式	29
2.4.4 关系运算符与关系表达式	30
2.4.5 逻辑运算符与逻辑表达式	31
2.4.6 逗号运算符与逗号表达式	32
2.4.7 条件运算符与条件表达式	32
2.4.8 各类基本数据类型数据间的混合运算	33
本章小结	36
习题	36
第3章 简单的 C 程序设计	39
3.1 结构化程序设计	40
3.1.1 程序的三种基本控制结构	40

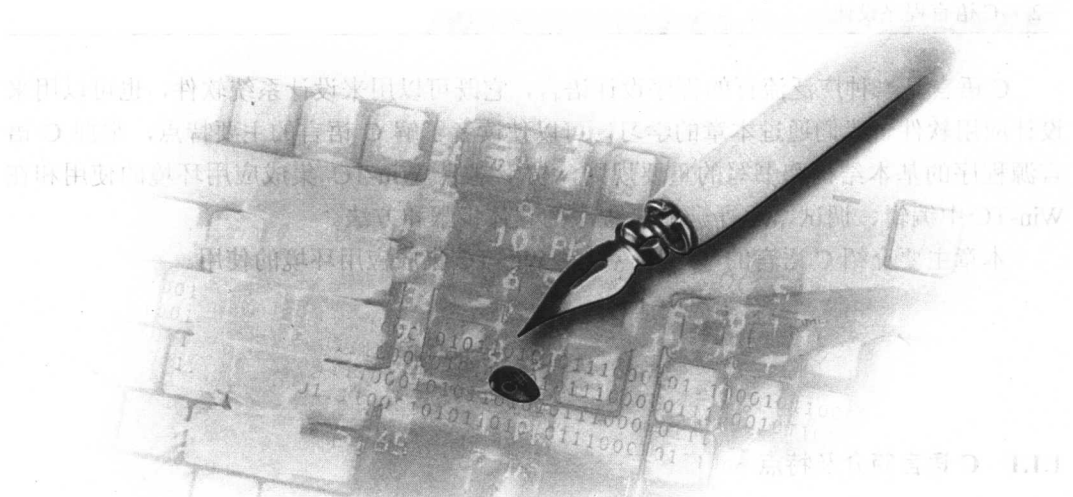
3.1.2 程序设计的一般步骤	41
3.1.3 C 语句概述	42
3.2 数据输入 / 输出在 C 语言中的实现	45
3.2.1 单个字符的输入/输出函数	45
3.2.2 数据的格式输入与格式输出	46
3.3 简单的 C 程序示例	53
本章小结	54
习题	55
第 4 章 选择结构 C 程序设计	58
4.1 if 结构	59
4.1.1 if 语句 (单分支结构)	59
4.1.2 if...else...语句 (双分支结构)	60
4.1.3 if...else...if 语句 (多分支结构)	60
4.2 switch 结构	63
本章小结	65
习题	65
第 5 章 循环结构 C 程序设计	70
5.1 while 语句	71
5.2 do...while 循环语句	73
5.3 for 语句	76
5.4 语句标号和 goto 语句	81
5.4.1 语句标号	81
5.4.2 goto 语句	81
5.5 几种循环的比较	82
5.6 break 和 continue 语句	84
5.6.1 break 语句	84
5.6.2 continue 语句	86
5.7 应用举例	88
本章小结	92
习题	92
第 6 章 数组	100
6.1 一维数组	101
6.1.1 一维数组的定义	101
6.1.2 一维数组在内存中的存储	102
6.1.3 一维数组元素的引用	103
6.2 二维数组	104
6.2.1 二维数组的定义	104

6.2.2 二维数组在内存中的存储	106
6.2.3 二维数组元素的引用	106
6.2.4 关于二维数组的理解	107
6.3 字符数组与字符串	109
6.3.1 字符串	109
6.3.2 字符数组	109
6.3.3 字符数组的输入与输出	110
6.3.4 常用字符串处理函数	111
6.4 数组应用举例	113
6.4.1 一维数组应用举例	113
6.4.2 字符数组应用举例	114
6.4.3 二维数组应用举例	115
本章小结	117
习题	118
第 7 章 函数	124
7.1 C 函数概述	125
7.1.1 函数的概念	125
7.1.2 C 函数分类	131
7.2 函数定义	131
7.2.1 函数定义的一般形式	132
7.2.2 return 语句	133
7.3 函数调用与函数声明	134
7.3.1 函数调用	134
7.3.2 函数声明	135
7.3.3 函数间参数传递的两种方式	137
7.4 函数的嵌套调用和递归调用	142
7.4.1 函数嵌套调用	142
7.4.2 函数递归调用	143
7.5 局部变量与全局变量	145
7.5.1 变量作用域	145
7.5.2 局部变量	145
7.5.3 全局变量	146
7.6 变量的存储类别	149
7.6.1 程序占用内存空间情况	149
7.6.2 变量的四种存储类别	149
7.6.3 各种存储类别变量使用举例	151
7.7 函数编写实例	153
本章小结	156

习题	156
第 8 章 编译预处理	163
8.1 宏定义	164
8.1.1 不带参数的宏定义 (简单替换)	164
8.1.2 带参数的宏定义	166
8.2 文件包含	170
*8.3 条件编译	172
本章小结	174
习题	175
第 9 章 指针	178
9.1 变量的地址和指针变量	179
9.2 指针变量的定义、初始化和运算	181
9.2.1 指针变量的定义和初始化	181
9.2.2 指针变量的运算	182
9.2.3 指针变量的引用	183
9.3 指针与数组	184
9.3.1 指向数组的指针变量的定义和初始化	184
9.3.2 通过数组的指针变量引用数组元素	185
9.3.3 用指针模拟可变长度的数组	186
9.4 指针变量作为函数的参数和返回值	187
9.4.1 普通指针变量作为函数参数	187
9.4.2 数组名作为函数参数	189
9.4.3 返回指针值的函数	190
9.5 字符串的指针和指向字符串的指针变量	191
9.5.1 字符串的表示形式	191
9.5.2 字符串指针作为函数参数	194
9.6 指针数组与指向指针的指针	195
9.6.1 指针数组的定义与使用	195
9.6.2 字符型指针数组	196
*9.6.3 指针数组作为 main() 函数的参数	197
9.7 指针运算举例	199
本章小结	202
习题	203
第 10 章 结构体与共用体	210
10.1 结构体	211
10.1.1 结构体类型的定义	211
10.1.2 结构体类型变量	212

10.1.3	结构体数组	217
10.2	共用体	219
10.2.1	共用体类型的定义	220
10.2.2	共用体变量的定义	220
10.2.3	共用体变量及其引用	221
10.3	结构体与共用体的比较	222
10.3.1	结构体与共用体的相同点	222
10.3.2	结构体与共用体的区别	222
10.4	typedef 命令	223
10.5	结构体与单向链表	223
10.5.1	指向结构体的指针	223
10.5.2	动态内存分配函数	224
10.5.3	结构体、指针在单向链表中的使用	227
	本章小结	231
	习题	232
第 11 章	文件	237
11.1	文件概述	238
11.1.1	缓冲文件系统	239
11.1.2	非缓冲文件系统	240
11.2	文件类型指针	240
11.3	文件的打开与关闭	241
11.3.1	文件的打开	241
11.3.2	文件的关闭	243
11.4	文件的读/写	244
11.4.1	字符读/写函数	244
11.4.2	字符串读/写函数	246
11.4.3	格式化读/写函数	248
11.4.4	数据块读/写函数	249
11.5	文件的定位	251
11.5.1	重返文件头函数	251
11.5.2	位置指针移动函数	252
11.6	文件检测函数	253
	本章小结	253
	习题	254
第 12 章	C 语言综合程序设计案例	257
12.1	算法和流程图	258
12.1.1	算法概念	258
12.1.2	流程图与算法的结构化描述	260

12.2 简单的“小学生算术自测系统”案例	264
12.2.1 功能分析	264
12.2.2 各模块设计	264
12.2.3 完整程序源代码清单	269
12.2.4 小学生算术自测系统设计小结	275
12.3 学生成绩管理系统示例源代码	275
附录	287
附录 1 ASCII 码与字符对照表	287
附录 2 常用库函数	288



第 1 章 C 语言概述

本章要点

- C 语言简介及特点
- C 语言源程序结构
- Win-TC 的安装与使用
- 在 Win-TC 中编辑源程序
- 在 Win-TC 中编译运行 C 程序

本章学习目标

- 熟练掌握 C 语言的特点
- 掌握 C 语言源程序的结构、书写规则
- 掌握 Win-TC 的安装与使用
- 掌握在 Win-TC 中编辑与编译运行 C 程序

C 语言是一种广泛流行的程序设计语言，它既可以用来设计系统软件，也可以用来设计应用软件。我们通过本章的学习，可以使读者了解 C 语言的主要特点，掌握 C 语言源程序的基本结构和书写的重要规则，初步掌握 Win-TC 集成应用环境的使用和在 Win-TC 中编辑、调试、运行一个 C 程序的具体步骤和方法。

本章主要介绍 C 语言的基本特点以及 Win-TC 集成应用环境的使用。

1.1 C 语言概述

1.1.1 C 语言简介及特点

1978 年，由美国电话电报公司（AT&T）贝尔实验室正式发布了 C 语言，由 B.W.Kernighan 和 D.M.Ritchie 合著了著名的《The C Programming Language》一书，简称为《K&R》，也有人称之为《K&R》标准。但是，在《K&R》中并没有定义一个完整的标准 C 语言，后来由美国国家标准学会在此基础上制定了一个 C 语言标准，于 1983 年发布，通常称之为 ANSI C。

早期的 C 语言主要应用于 UNIX 系统。由于 C 语言的强大功能和各方面的优点逐渐为人们所认知，到了 20 世纪 80 年代，C 语言开始进入其他操作系统，并很快在各类大、中、小型和微型计算机上得到了广泛的使用，成为当代最优秀的程序设计语言之一。

C 语言是一种结构化语言：层次清晰，便于按模块化方式组织程序，易于调试和维护，表现能力和处理能力极强；具有丰富的运算符和数据类型，便于实现各类复杂的数据结构；还可以直接访问内存的物理地址，进行位（bit）一级的操作；可以实现对硬件的编程操作，因此 C 语言是集高级语言和低级语言的功能于一体，既可用于系统软件的开发，也适合于应用软件的开发；C 语言还具有效率高、可移植性强等特点，因此被广泛移植到了各种类型计算机上，从而形成了多种版本的 C 语言。

目前，最流行的 C 语言有以下几种：Microsoft C 或称 MS C；Borland Turbo C 或称 Turbo C，基于 Turbo C 2.0 的 Win-TC；AT&T C 等。这些 C 语言版本不仅实现了 ANSI C 标准，而且在此基础上各自进行了一些扩充，使之更加方便、完美。

在 C 语言的基础上，1983 年又由贝尔实验室的 Bjarne Stroustrup 推出了 C++。C++ 进一步扩充和完善了 C 语言，成为一种面向对象的程序设计语言。C++ 提出了一些更为深入的概念，它所支持的这些面向对象的概念易于将问题空间直接地映射到程序空间，为程序员提供了一种与传统结构程序设计不同的思维方式和编程方法，因而也增加了整个语言的复杂性，掌握起来有一定难度。

但是，C 语言是 C++ 的基础，C++ 和 C 语言在很多方面是兼容的。因此，掌握了 C 语言，再进一步学习 C++ 就能以一种熟悉的语法来学习面向对象的语言，从而达到事半功倍的目的。

1.1.2 C语言源程序结构

用户根据某个目的利用C语言编写的源程序,简称为C程序。C程序是一种函数结构,它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。为了说明C语言源程序结构的特点,先看以下几个程序。这几个程序由简到难,表现了C语言源程序在组成结构上的特点。虽然有关内容还未介绍,但读者可从这些例子中了解到组成一个C源程序的基本部分和书写格式。

例 1.1 输出字符串“c语言世界,您好!”。

```
main()
{
    printf("c语言世界,您好! \n");
}
```

`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,且只能有一个主函数(`main` 函数)。函数调用语句 `printf` 函数的功能是把要输出的内容送到显示器去显示。`printf` 函数是一个由系统定义的标准函数,可在程序中直接调用。

例 1.2 输入一个数求其正弦函数的值。语句前出现的序号是为了在例题分析时表示语句的位置。

```
1  #include "stdio.h"
2  #include "math.h"
3  main()
4  {
5      double x,s;
6      printf("input number:\n");
7      scanf("%lf",&x);
8      s=sin(x*3.14/180);
9      printf("sine of %lf is %lf\n",x,s);
10 }
```

例题分析:

第1、2行, `include` 称为文件包含命令扩展名为 `.h` 的文件,也称为头文件或首部文件。

第5行,定义两个实数变量,以被后面程序使用。

第6行,显示提示信息。

第7行,从键盘获得一个实数 `x`。

第8行,求 `x` 的正弦,并把它赋给变量 `s`。

第9行,显示程序运算结果。

第10行,“`}`”表示 `main` 函数结束。

程序的功能是从键盘输入一个数 `x`,求 `x` 的正弦值,然后输出结果。在 `main()` 之前

的两行称为预处理命令（详见后面章节）。预处理命令还有其他几种，这里的#include 称为文件包含命令，其意义是把尖括号<>或英文引号""内指定的文件包含到本程序来，成为本程序的一部分。被包含的文件通常是由系统提供的，其扩展名为.h，也称为头文件或首部文件，C语言的头文件中包括了各个标准库函数的函数原型。因此，凡是在程序中调用一个库函数时，都必须包含该函数原型所在的头文件。在本例中，使用了三个库函数：输入函数 scanf，正弦函数 sin，输出函数 printf。sin 函数是数学函数，其头文件为 math.h 文件，因此在程序的主函数前用#include 命令包含了 math.h。scanf 和 printf 是标准输入/输出函数，其头文件为 stdio.h，因此在主函数前也用#include 命令包含了 stdio.h 文件。

需要说明的是，C语言规定对 scanf 和 printf 这两个函数可以省去对其头文件的包含命令。所以在本例中也可以删去第二行的包含命令#include。同样，在例 1.1 中使用了 printf 函数，也省略了包含命令。

在例题中的主函数体中又分为两部分，一部分为说明部分，另一部分为执行部分。说明是指对变量的类型说明。例题中未使用任何变量，因此无说明部分。C语言规定，源程序中所有用到的变量都必须先说明、后使用，否则将会出错。这一点是编译型高级程序设计语言的一个特点，与解释型的 Basic 语言是不同的。说明部分是 C 源程序结构中很重要的组成部分，本例中使用了两个变量 x 和 s，用来表示输入的自变量和 sin 函数值。由于 sin 函数要求这两个量必须是双精度浮点型，故用类型说明符 double 来说明这两个变量。说明部分后的四行为执行部分或称为执行语句部分，用以完成程序的功能。执行部分的第一行是输出语句，调用 printf 函数在显示器上输出提示字符串，请操作人员输入自变量 x 的值；第二行为输入语句，调用 scanf 函数，接收键盘上输入的数并存入变量 x 中；第三行是调用 sin 函数并把函数值送到变量 s 中；第四行是用 printf 函数输出变量 s 的值，即 x 的正弦值。程序结束。

运行本程序时，首先在显示器屏幕上给出提示串“input number:”，这是由执行部分的第一行完成的。用户在提示下从键盘输入某一数，如 5，按下回车键，接着在屏幕上给出计算结果。

例 1.3 输入两个整数，程序执行后输出其中较大的数。

```
int max(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"); /*显示提示信息*/
    scanf("%d%d",&x,&y);        /*输入x,y值*/
    z=max(x,y);                 /*调用max函数*/
    printf("maxmum=%d",z);      /*输出*/
}

int max(int a,int b)           /*定义max函数*/
{
```

```
    if(a>b)return a;          /*把结果返回主调函数*/  
    else return b;  
}
```

例题分析：例题中程序的功能是由用户输入两个整数，程序执行后输出其中较大的数。本程序由两个函数组成，主函数和 max 函数，可从主函数中调用其他函数。max 函数的功能是比较两个数，然后把较大的数返回给主函数，它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）。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，关于函数的详细内容将在第 7 章介绍。

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并送入变量 x、y 中，然后调用 max 函数，并把 x、y 的值传送给 max 函数的参数 a、b。在 max 函数中比较 a、b 的大小，把大者返回给主函数的变量 z，最后在屏幕上输出 z 的值。

对上述例题进行分析后可归纳出 C 源程序的结构特点：

- 1) 一个 C 语言源程序可以由一个或多个源文件组成。
- 2) 每个源文件可由一个或多个函数组成。
- 3) 一个源程序不论由多少个文件组成，都有且只有一个 main 函数，即主函数。
- 4) 源程序中可以有预处理命令（include 命令仅为其中的一种），预处理命令通常应放在源文件或源程序的最前面。
- 5) 每一个说明，每一个语句都必须以分号结尾。但预处理命令、函数头和花括号“{”之后不能加分号。
- 6) 标识符、关键字之间必须至少加一个空格以示间隔。若已有明显的间隔符，也可不再加空格来间隔。

注意：函数分为两种，一种为 C 库函数，由 C 编译系统提供，使用前不必说明，但必须在文件前把该函数所在的头文件用包含命令 include 包含到文件中；另一种是用户自定义的函数，它们在被调用之前应先予以说明。各函数之间是并列的，没有从属关系，函数之间可以互相调用（但主函数不能被其他函数调用）。

从书写清晰，便于阅读、理解、维护的角度出发，在书写程序时，应遵循以下规则：

- 1) 一个说明或一个语句占一行。
- 2) 用 {} 括起来的部分，通常表示了程序的某一层结构。{} 一般与该结构语句的第一个字母对齐，并单独占一行。
- 3) 低一层次的语句或说明可比高一层次的语句或说明缩进若干格后书写。以便看起来更加清晰，增加程序的可读性。在编程时应力求遵循这些规则，以养成良好的编程风格。

1.2 Win-TC 环境简介

目前在程序设计基础、C 语言程序设计等有关 C 语言的课程教学中,很多教师采用的是 Turbo C 2.0 (简称 TC 2.0) 的开发环境。由于该开发软件是基于 MS-DOS 的,存在诸如不支持鼠标操作、字体大小不能调整、文本复制步骤繁琐、必须另外启动 UC DOS 等中文环境才能支持中文输入/输出等缺点,而这些缺陷或多或少影响了教学活动开展的效果。而如果采用像 Visual C++ 这样的开发工具,对于 C 语言的开发者来说又显得有点大材小用,而且对于初学者来说,掌握 Visual C++ 开发工具并不是一件容易的事情。

基于这样的认识,本书使用了一个基于 TC 2.0 编译内核开发出来的编译环境——Win-TC,它实际上是对 TC 2.0 的一个封装。对于刚开始学 C 语言的学生来说,通常的做法是先学习 TC 2.0 的使用,通过一两个星期的学习掌握其基本的使用方法后再学习 Win-TC 的使用。这样既让学生了解了 C 语言基本调试工具 TC 2.0 的基本使用方法,又能在以后的大部分教学时间中充分利用 Win-TC 的优点服务于教与学。

1.2.1 Win-TC 的安装与使用

1. Win-TC 的安装

Win-TC 是一个 Turbo C 2.0 的 Windows 平台开发工具,该软件使用 Turbo C 2.0 为内核,提供 Windows 平台的开发界面,它支持 Windows 平台下的功能,例如剪切、复制、粘贴、查找和替换等。本教材使用的是 Win-TC v1.9.1 简繁体版,可以正常运行于 Windows 98 及以上的简体及繁体 Windows 操作系统中。下面是 Win-TC v1.9.1 的安装过程。

首先,把购买的 Win-TC v1.9.1 软件存入硬盘中,双击 wintc191.exe 文件,进入 Win-TC v1.9.1 软件的安装界面,如图 1.1 所示。

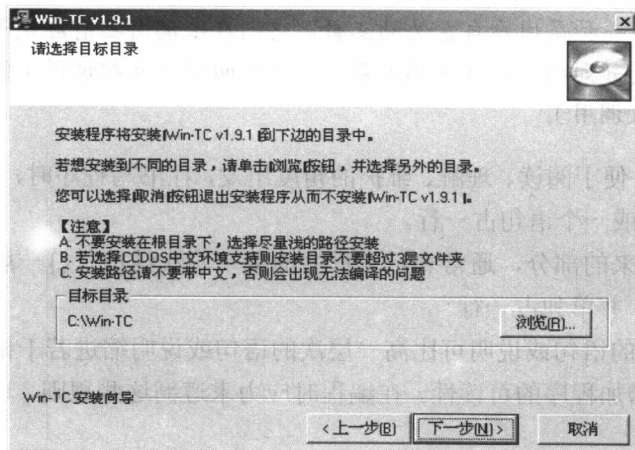


图 1.1 Win-TC 安装界面

单击“浏览”按钮，然后选择 Win-TC 的安装路径，再选择“下一步”按钮，如图 1.2 所示。

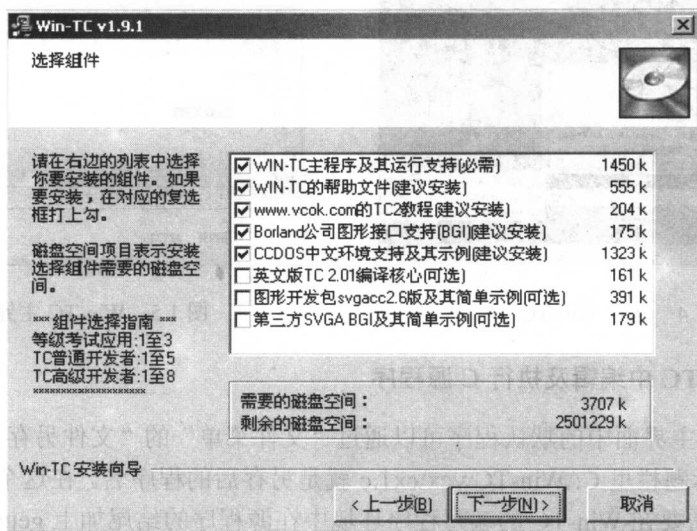


图 1.2 Win-TC 安装组件

在图 1.2 中，一般选择第一～第五个复选项就可以了。单击“下一步”按钮，如图 1.3 所示，进入安装进度，即完成 Win-TC 的安装过程。

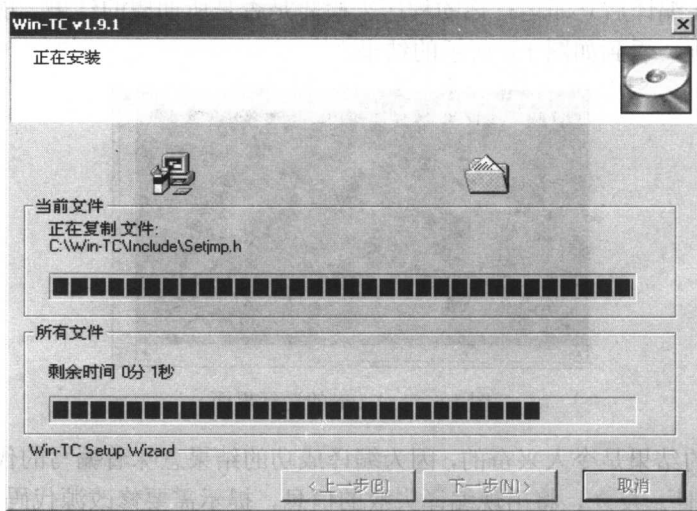


图 1.3 Win-TC 的安装进度

2. Win-TC 的使用

单击 Windows 系统的任务栏上的“开始”菜单，选择“程序”项，然后选择“Win-TC”，运行 Win-TC，如图 1.4 所示。运行画面如图 1.5 所示。