



普通高等教育“十二五”规划教材

Case based C language program design

案例式C语言程序设计

主编:彭文艺



电子科技大学出版社

普通高等教育“十二五”规划教材

案例式 C 语言程序设计

主 编： 彭文艺



电子科技大学出版社

图书在版编目(CIP)数据

案例式 C 语言程序设计 / 彭文艺主编. —成都:

电子科技大学出版社, 2015. 3

ISBN 978-7-5647-2822-9

I. ①案… II. ①彭… 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 026938 号

案例式 C 语言程序设计

主 编 彭文艺

出 版: 电子科技大学出版社 (成都市一环路东一段 159 号电子信息产业大厦 邮编: 610051)

策划编辑: 汤云辉

责任编辑: 汤云辉

主 页: www.uestcp.com.cn

电子邮箱: uestcp@uestcp.com.cn

发 行: 新华书店经销

印 刷: 北京奇麒麟印刷有限责任公司分公司

成品尺寸: 185mm×260mm 印张 20.75 字数 518 千字

版 次: 2015 年 3 月第一版

印 次: 2015 年 3 月第一次印刷

书 号: ISBN 978-7-5647-2822-9

定 价: 39.00 元

■ 版权所有 侵权必究 ■

◆ 本社发行部电话: 028-83202463; 本社邮购电话: 028-83201495。

◆ 本书如有缺页、破损、装订错误, 请寄回印刷厂调换。

前 言

“程序设计”是高校理工科重要的计算机基础课程，该课程以培养学生掌握程序设计的思想和方法为目标，以培养学生的实践能力和创新能力为重点。C语言是得到广泛使用的程序设计语言之一，它既具备高级语言的特性，又具有直接操纵计算机硬件的能力，并以其良好的程序结构和便于移植的特性而拥有大量的使用者。目前，许多高校都把C语言列为首门要学习的程序设计语言。

虽然目前有关C语言的教材很多，但大多比较注重C语言知识的学习，不利于培养学生的程序设计能力和语言应用能力。本书以程序设计为主线，从应用出发，通过案例和问题引入相关的语法知识，重点讲解程序设计的思想和方法，并始终贯彻整本书，避免机械式地记忆语法知识，坚持通过写程序去掌握C语言知识的理念。

在教材的结构设计上，强调学以致用，使学生从一接触C语言，就开始练习编程。全书共分10章：第1章为C语言基础，通过简单的案例来介绍C语言的结构特点，以及如何用Visual C++6.0实现C语言程序的运行等；第2章为数据类型与运算符，主要介绍C语言的基本数据类型、输入和输出函数、常量和变量、运算符与表达式、运算符的优先级与结合性等；第3章为顺序结构，介绍了算法的基础知识、通过一系列典型案例逐步介绍了顺序结构程序设计；第4章为选择结构，介绍了if和switch选择结构语句、通过一系列典型案例逐步介绍了选择结构程序设计；第5章为循环结构，介绍了while、do-while和for循环语句、通过一系列典型案例逐步介绍了循环结构程序设计；第6章为数组，介绍数值数组和字符数组以及数组应用；第7章为指针，主要围绕指针的定义、指针的用途及如何应用展开；第8章为函数，介绍函数的概念、定义及函数的调用方式；第9章为结构体与联合体，介绍结构体和联合体的概念、结构体数组的使用、结构体指针的应用等内容；第10章为文件，介绍基本的文件知识，主要介绍C语言读写文件的方法。

本书具有如下特色。

(1) 按照学习逻辑安排章节顺序，有些C语言教材会存在前面章节的内容需要用到后面章节知识的情况，本书根据学习新知识时的思维过程设置内容的顺

序。

(2) 学以致用，注重应用。本书以“技术知识——案例分析——技术应用”为主线，先介绍基础知识，再讲授案例，最后给出应用实例，以便于读者掌握该章的重点知识并提高编程能力。

(3) 案例丰富，实用性强。本书所有案例均给出了算法分析和算法步骤，并上机调试运行后给出了结果，很多案例举一反三，一题多解，增强实用能力，便于读者理解和自学。

在本书的编写过程中，作者查阅、借鉴了许多参考资料，在此谨向诸多学者、同仁表示由衷的敬意和感谢。尽管在写作过程中作者投入了大量的时间和精力，但由于时间及水平有限，错误和不足之处仍在所难免，敬请读者批评指正。

彭文艺

目 录

第1章 C语言基础.....	1
1.1 程序设计基本概念	1
1.1.1 程序和程序设计	1
1.1.2 程序设计语言	3
1.2 C语言程序开发过程.....	5
1.3 C语言字符集、标识符和关键字.....	15
1.3.1 C语言字符集.....	15
1.3.2 C语言标识符.....	15
1.3.3 C语言关键字.....	16
1.4 C语言程序结构.....	17
习题1.....	20
第2章 C语言数据类型和表达式.....	21
2.1 C语言的数据类型.....	21
2.1.1 整型数据	21
2.1.2 实型数据	23
2.1.3 字符型数据	23
2.2 常量与变量	25
2.2.1 常量	25
2.2.2 变量	27
2.3 输入/输出	30
2.3.1 格式化输出函数 printf().....	31
2.3.2 格式化输入函数	33
2.3.3 字符输入/输出函数	36
2.4 运算符和表达式	37
2.4.1 算术运算符和算术表达式	37
2.4.2 赋值运算符与赋值表达式	38
2.4.3 自增自减运算符和自增自减表达式	40
2.4.4 逗号运算符和逗号表达式	41
2.4.5 sizeof运算符和sizeof表达式	41
2.5 数据类型转换	42
2.5.1 自动转换	42
2.5.2 赋值转换	43

2.5.3 强制转换	44
2.6 部分库函数	45
习题 2	48
第 3 章 顺序结构程序设计	51
3.1 算法	51
3.1.1 算法的概念	51
3.1.2 算法的特性	51
3.1.3 算法的评价标准	51
3.1.4 算法的描述	52
3.2 C 语言的语句	54
3.3 顺序结构程序设计	56
习题 3	58
第 4 章 选择结构程序设计	59
4.1 关系运算符与关系表达式	59
4.1.1 关系运算符	59
4.1.2 关系表达式	60
4.2 逻辑运算符与逻辑表达式	60
4.2.1 逻辑运算符	60
4.2.2 逻辑表达式	62
4.3 选择结构语句	63
4.3.1 单分支 if 语句	64
4.3.2 双分支 if-else 语句	69
4.3.3 多分支 if 语句	74
4.3.4 switch 语句	78
4.3.5 if 语句的嵌套	83
4.4 程序设计举例	87
习题 4	94
第 5 章 循环结构程序设计	98
5.1 while 语句	99
5.1.1 while 语句形式	99
5.1.2 while 语句执行过程	99
5.1.3 while 语句的应用	100
5.2 do-while 语句	104
5.2.1 do-while 语句的形式	104
5.2.2 do-while 语句的执行过程	104

5.2.3 do-while 语句的应用	104
5.2.4 while 语句与 do-while 语句的区别	105
5.3 for 语句	106
5.3.1 for 语句的形式	106
5.3.2 for 语句的执行过程	107
5.3.3 for 语句的应用	107
5.3.4 for 语句的变形	108
5.3.5 for 语句与 while 语句比较	109
5.4 break 语句和 continue 语句	110
5.4.1 break 语句	110
5.4.2 continue 语句	112
5.4.3 break 语句和 continue 语句的区别	113
5.5 循环的嵌套结构	114
5.5.1 双重循环的嵌套	115
5.5.2 多重循环的嵌套	118
5.6 循环结构程序设计举例	119
习题 5	126
第 6 章 数组	133
6.1 一维数组	134
6.1.1 一维数组的定义	134
6.1.2 一维数组的访问	135
6.1.3 一维数组赋初值	136
6.1.4 一维数组的应用	139
6.2 二维数组	145
6.2.1 二维数组的定义	145
6.2.2 二维数组的访问	147
6.2.3 二维数组的赋值	147
6.2.4 二维数组的应用	148
6.3 字符数组	154
6.3.1 字符数组的定义	154
6.3.2 字符数组初始化	154
6.3.3 字符数组输入输出	155
6.3.4 字符串处理函数	158
6.4 程序举例	162
习题 6	167

第 7 章 指针	171
7.1 指针与指针变量	171
7.1.1 指针概念	171
7.1.2 指针变量	171
7.1.3 指针变量的定义	172
7.1.4 指针变量初始化	173
7.1.5 指针运算符	174
7.1.6 指针运算	175
7.1.7 多级指针	180
7.2 指针与数组	180
7.2.1 一维数组元素的指针访问	180
7.2.2 二维数组元素的指针访问	186
7.2.3 指向一维数组的指针变量	189
7.2.4 指针数组	191
7.3 字符指针与字符串	193
7.3.1 字符串的表现形式	193
7.3.2 用字符指针处理字符串	195
7.3.3 用字符指针数组处理多个字符串	197
7.4 动态内存分配	199
习题 7	202
第 8 章 函数与编译预处理	206
8.1 模块化程序设计	208
8.2 函数的定义	211
8.2.1 标准库函数	211
8.2.2 函数的定义	212
8.3 函数的调用	214
8.3.1 函数调用形式	214
8.3.2 函数调用方式	214
8.3.3 函数调用结果的返回	217
8.3.4 函数调用过程	219
8.4 函数间数据传递	221
8.4.1 实参的值传递	221
8.4.2 实参的地址传递	223
8.4.3 数组名作为实参的地址传递	224
8.5 函数的嵌套调用	227
8.6 递归函数与递归调用	229

8.7 指针与函数	232
8.7.1 指向函数的指针	232
8.7.2 返回指针的函数	235
8.7.3 带参数的 main()函数	236
8.8 变量作用域与存储方式	237
8.8.1 变量作用域	237
8.8.2 变量的存储方式	242
8.9 编译预处理	245
8.9.1 宏定义	246
8.9.2 文件包含	249
8.9.3 条件编译	250
习题 8	253
第 9 章 结构体、联合体与枚举	256
9.1 结构体	256
9.1.1 结构体类型的定义	256
9.1.2 结构体类型变量的定义	257
9.1.3 结构体变量的访问	260
9.1.4 结构体类型变量的初始化	263
9.2 结构体数组与链表	263
9.2.1 结构体数组的定义与访问	263
9.2.2 结构体数组初始化和应用	266
9.2.3 链表	267
9.3 联合体	278
9.3.1 联合体类型的定义	279
9.3.2 联合体类型变量的定义	279
9.3.3 联合体类型变量的访问	280
9.4 枚举	283
9.4.1 枚举类型定义	283
9.4.2 枚举类型变量的定义	284
9.5 类型定义	284
9.6 程序设计举例	287
习题 9	291
第 10 章 文件	295
10.1 文件的概述	295
10.1.1 文件的分类	295
10.1.2 文件的缓冲区	296

10.1.3 文件的存取方式	297
10.1.4 文件类型的指针	297
10.2 文件的常用操作	297
10.2.1 文件的打开与关闭	297
10.2.2 文件的读写	300
10.2.3 文件的定位	310
习题 10	312
附录 I 常用字符与 ASCII 码对照表	313
附录 II C 语言运算符的优先级和结合方向	314
附录 III 常用 C 库函数	315
参考文献	319

第1章 C语言基础

【学习目标】

1. 了解程序设计语言的发展和 C 语言的特点。
2. 掌握 C 语言程序的基本结构。
3. 能正确运用 C 语言的标识符及关键字。
4. 能熟练运用 Visual C++集成开发环境创建、编译、连接和运行 C 语言程序。

1.1 程序设计基本概念

在当今这个信息化社会中,计算机这个信息设备是人们赖以工作、学习、生活的主要工具,它之所以能处理大量复杂的问题,则全依靠硬件和软件的协同工作。计算机之所以比其他的诸如电视机、计算器等电子设备的功能更为强大,就是因为计算机软件的可编程的特点。换言之,同样的硬件配置,加载不同的软件,就可以完成不同的工作。

当用户使用计算机来完成某项工作时,会有两种可能的情况:一种是可以借助于现有的应用软件来完成任务,例如,一般的文字处理工作可以使用 Office 组件,科学计算可以选择 MATLAB,工程制图可以选用 AUTOCAD,但更多的情况却是没有完全合乎要求的软件可以选用,这时就需要使用计算机程序设计语言来编制程序,以完成特定的工作,这就是程序设计。

1.1.1 程序和程序设计

1. 程序

一般来说,当要完成一项复杂任务时,需要进行一系列的具体工作。这些按照一定顺序安排的工作就是操作序列,或称为(工作)程序。

例如,一个专题报告会议的程序如下:

- ① 宣布会议开始。
- ② 介绍出席会议领导及专家。
- ③ 主持人致辞。
- ④ 报告人作专题报告。
- ⑤ 与会代表发言。
- ⑥ 宣布会议结束。

对这个会议的(工作)程序作一个简要归纳,该程序主要用于描述完成召开专题报告会议这项任务所涉及的对象和动作规则。这里,对象就是上述程序中的领导、专家、主持人、报告人、代表等,而宣布、介绍、致辞、报告、发言等都是动作。这些动作的

先后以及动作所作用的对象都要符合一定的规则。比如，不能先做报告、发言，再致辞，否则，这项专题报告会议就不成功。

由此可见，程序的概念是普遍的，将这个概念运用到计算机上，则程序就是为完成特定任务而编写的一系列指令的集合。这里所谓指令是指计算机能够直接识别和执行的命令，它在计算机内部以二进制码表示。

2. 程序设计

为了让计算机能够按人的意图处理上述问题，人们必须借助计算机能够理解的语言即程序设计语言，告诉计算机要处理什么以及如何处理，这个过程便是程序设计，即用程序设计语言编写出一些指令来驱动计算机完成特定的工作。

例如，用计算机来完成这样一项任务——统计一个班级学生的考试成绩，并按成绩排序。

这是许多学校教学管理的一项基础工作，如果让专门的工作人员去处理，仅按照上述要求，即可以很快完成。但如果交给计算机来处理，必须要对上述任务进行明确的定义，并且要规定完成任务的确切步骤。

一般来讲，程序设计过程包括以下几个步骤：

(1) 分析问题，建立数学模型

要用计算机解决实际问题，首先应对要解决的问题进行详细的分析，弄清需要计算机“做什么”，包括需要输入什么数据，要得到什么结果，最后输出什么等。然后需要把实际问题简化，用数据语言来描述它，这称为建立数学模型。主要包括以下三个部分。

输入：就是说为完成以上任务，需要什么条件，例如，需要学生姓名、学号、相关课程的考试成绩。

处理：就是要让计算机如何对输入信息加工，例如，需要对各科成绩求和并计算平均值。

输出：就是用户所需要的结果，例如，在屏幕上显示或打印出一个班按各科成绩平均值由高到低排列顺序的学生名册。

(2) 设计算法

弄清楚要计算机“做什么”后，就需要设计算法，明确让计算机“怎么做”，而算法则是对“怎么做”的步骤的描述。解决一个问题可能有多种算法，这时应该通过分析、比较，选择一种最优的算法。算法设计后，往往需要选择适当的形式将算法描述出来。例如，对上述问题的一种可能的算法描述为：

- ①输入全部学生姓名、学号、英语成绩、数学成绩、计算机基础成绩；
- ②对每个学生各门课程成绩计算平均值；
- ③按平均值对学生成绩进行排序；
- ④按平均成绩从高到低的顺序输出或打印。

(3) 编程

当算法设计完后，就可以用某种程序设计语言将算法编写出程序。

(4) 调试程序

编写出的程序还必须在计算机上运行，排除程序中可能的错误，即找出在程序编写

过程中的语法错误和逻辑错误，直到得到正确的结果为止，这个过程称为程序调试。

不难看出，编程只是程序设计的一个方面，在实际编程之前必须经过认真的分析与算法设计。

1.1.2 程序设计语言

1. 程序设计语言

人与计算机之间进行信息交换通常使用程序设计语言。人们把自己的意图用某种程序设计语言编写程序，并将其输入计算机，告知完成什么任务以及如何完成，达到计算机为人做事的目的。程序设计语言经历了机器语言、汇编语言和高级语言三个阶段。

(1) 机器语言

机器语言是机器的指令序列。机器指令是用一串 0 和 1 的二进制编码表示的，可以直接被计算机识别并执行。机器语言是面向机器的语言，与计算机硬件密切相关，针对某一类型计算机编写的机器语言程序不能在其他类型计算机中运行。机器语言的缺点是编写程序很困难，而且程序难改、难读。但机器语言编写的程序执行速度快，占用内存空间少。

(2) 汇编语言

汇编语言是指用一些有特定含义的符号替代机器的指令作为编程用的语言，其中使用了很多英文单词的缩写，这些字母和符号称为助记符。这些助记符易编程、可读性好、修改方便，但机器并不认识，所以需把它翻译成相对应的机器语言程序。这种翻译的过程就叫编译，将汇编语言程序翻译成相应的机器语言程序是由汇编程序完成的。汇编语言的每一条语句和机器语言指令一一对应，故仍是一种面向机器的语言。

(3) 高级语言

高级语言是用英文单词、数学表达式等易于理解的形式书写的，并按严格的语法规则和一定的逻辑关系组合的一种计算机语言。高级语言编写的程序独立于计算机硬件，可读性好、易于维护，提高了程序设计的效率。

2. C 语言的背景

程序设计语言有很多，在众多的程序设计语言中，C 语言的强大功能和各方面的优点逐渐为人们所认识，并很快在各类大、中、小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 语言作为计算机编程语言，具有功能强、语句表达简练、控制 and 数据结构丰富灵活、程序运行速度快、所需内存空间少等特点。它既具有高级语言的特点，又具有汇编语言中的位、地址、寄存器等概念，拥有其他许多高级语言所没有的操作能力，所以有人说 C 语言是介于高级语言和低级语言之间的一种语言。C 语言既适合编写系统软件，又可用于编写应用软件。总的来说 C 语言有如下特点：

(1) C 语言的语言成分简洁、紧凑、书写形式自由；

(2) C 语言拥有丰富的数据类型；

(3) C 语言的运算符丰富、功能强大；

- (4) C 语言是结构化程序设计语言;
- (5) C 语言对语法限制不严格, 程序设计灵活;
- (6) C 语言编写的程序具有良好的可移植性;
- (7) C 语言可以实现汇编语言的大部分功能;
- (8) C 语言编译后生成的目标代码小, 质量高, 程序的执行效率高。

程序设计语言的发展与演变体现出继承与创新。C 语言渊源于 B 语言, B 语言又渊源于 BCPL 语言, BCPL 语言又渊源于 CPL 语言, CPL 语言又渊源于 ALGOL 60 语言。语言之间的这种继承与丰富、发展与完善的关系可以由图 1-1 来描述。

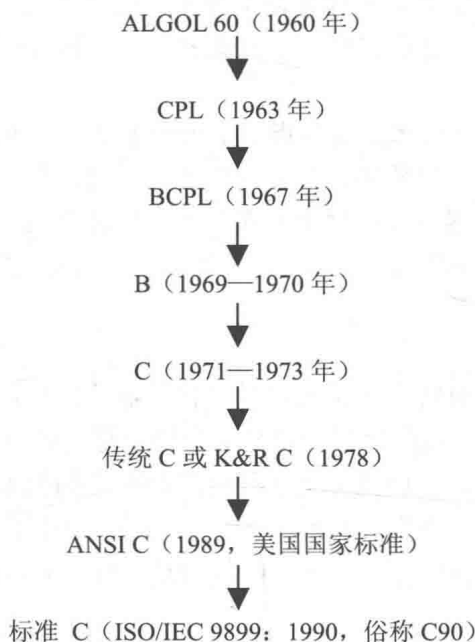


图 1-1 C 语言的产生与发展历程

ALGOL, 也被称为国际代数语言, 是计算机发展史上首批产生的高级语言, 当时还是晶体管计算机流行的时代, 由于 ALGOL 语句和普通语言表达式接近, 更适于数值计算, 所以 ALGOL 多用于科学计算机。

1963 年英国剑桥大学推出了 CPL (Combined Programming Language) 语言。CPL 语言在 ALGOL 60 的基础上接近硬件一些, 但规模比较大, 难以实现。1967 年英国剑桥大学的 Martin Richards 对 CPL 语言做了简化, 推出了 BCPL (Base Combined Programming Language) 语言。1970 年美国贝尔实验室的 Ken Thompson 以 BCPL 语言为基础, 又做了进一步的简化, 设计出了很简单的而且很接近硬件的 B 语言 (取 BCPL 的第一个字母), 并用 B 语言写出了第一个 UNIX 操作系统。但 B 语言过于简单, 功能有限, 1971 年至 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 (取 BCPL 的第二个字母)。C 语言即保持 BCPL 语言和 B 语言的优点 (精练、接近硬件), 又克服了它们的缺点 (过于简单, 数据无类型等)。

1978 年, Brain W.Kerninghan 和 Dennis Ritchie 合著了《The C Programming Language》一书, 被称为传统的 C 语言。1983 年美国国家标准化协会 (American National Standard

Institute)制定了C语言标准,称为ANSI C,并在1989年通过了认证,俗称C89。次年,即1990年,国际标准化组织(International Standard Organization, ISO)接受了ANSI C标准,称为ISO/IEC 9899:1990,是C语言的第一个国际标准,也称为标准C,俗称C90,本书使用ANSI C标准。

1.2 C语言程序开发过程

1. C语言程序开发过程

把一个写在纸上的C语言源程序代码转换为能在计算机操作系统平台上执行的程序需要经过以下几个过程:

(1) 编辑:利用编辑程序,将源程序代码输入计算机中,最终生成一个源程序文件(.c)的过程。这期间必须严格遵守C语言的语法规则,要特别注意必须在英文状态下输入源程序的所有字符和标点符号(中文文字除外)。

(2) 编译:使用编译程序对源程序文件进行编译,编译程序会自动分析、检查源程序文件的语法错误,并按“错误”和“警告”(error、warning)两类错误报告出错行和原因。用户根据报告信息修改源程序,再编译,直到程序编译通过,生成中间目标程序文件(.obj)。

(3) 连接:使用连接程序将中间目标文件与所指定的库文件和其他的中间目标文件进行连接,连接程序会自动分析、检查,并给出连接出错的原因。用户根据错误报告信息再修改源程序,再编译,再连接,直到连接通过,生成可执行文件(.exe)。

(4) 运行:连接通过后,就可以运行可执行文件,得到运行结果。当然,也可能由于解决问题的算法错误,而使源程序具有逻辑错误,导致得到错误的运行结果,或者由于语义上的错误,出现运行时错误。这就需要检查算法,修改源程序,再编译,再连接,直到运行结果正确。

2. 使用 Visual C++ 6.0 环境开发 C 语言程序过程

C语言程序的编译系统有许多种,早期非常流行的编译系统有Turbo C,它是美国Borland公司生产的一套DOS平台上的C语言编译系统。随着面向对象技术的飞速发展,面向对象技术C++、C#陆续面世,在Windows程序和大型软件中得到广泛的使用。目前被广泛使用的C语言程序编译系统是Visual C++ 6.0,它是美国微软公司生产的一套Windows平台上的C语言编译系统。在Visual C++ 6.0环境下开发C语言程序一般需要经过以下几个过程:

(1) 编辑 C 源程序文件

所谓C源程序文件,就是存放C源程序的文件,C源程序文件可以随意命名,但扩展名必须是.c(如tt.c)。

编辑C源程序文件有两种含义:建立或修改C源程序文件,如果磁盘中没有相应的文件,则编辑的目的就是建立此文件,将程序输入到文件中。如果相应的文件已经存在,则编辑的目的就是要修改此文件。

①建立 C 源程序文件

第一步：打开 Visual C++ 6.0 运行环境界面，用鼠标依次单击【开始】→【所有程序】→【Microsoft Visual Studio 6.0】→【Microsoft Visual C++ 6.0】，将弹出 Visual C++ 6.0 运行环境界面，如图 1-2 所示。（已关闭“Did you know...”对话框）

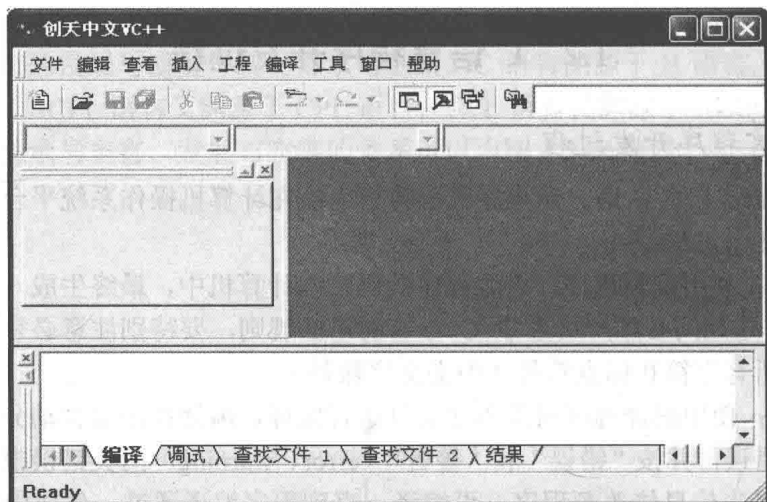


图 1-2 Visual C++ 6.0 运行环境界面图

第二步：建立 C 源程序文件，在图 1-2 中，单击【文件】菜单下的【新建...】命令弹出“新建”对话框，如图 1-3 所示。

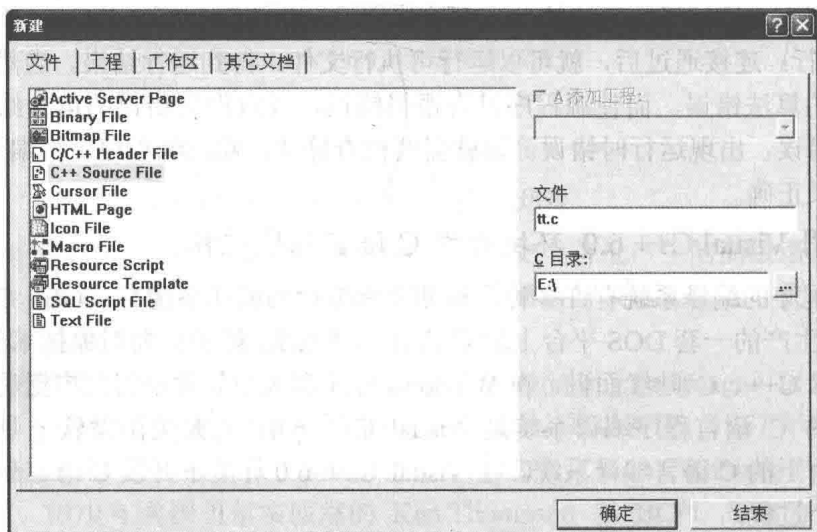


图 1-3 新建对话框

在图 1-3 对话框中，选择“文件”选项卡，在“文件”选项卡的列表框中，选择“C++ Source File”文件类型，在文件文本框中输入文件名，注意一定不要漏掉 C 源程序文件名的后缀“.c”，例如在图 1-3 中输入的文件名为“tt.c”，在目录文本框中，单击旁边“...”按钮后，从弹出的对话框中选择保存 C 源程序文件所在路径，例如在图 1-3 中选择了“E:\”作为存储 C 源程序的路径。最后单击“确定”按钮，将弹出一个输入源程序代码的编辑