

Chengxushejijiaocheng

C 和 C++ 程序设计教程

21世纪文化教材

赵建周 杨庆祥 / 主编



航空工业出版社

7.23.12
ZJZ.1/1

C 和 C++ 程序设计教程

主 编 赵建周 杨庆祥

副主编 郭新志 靳瑞萍

李卫平 李安伏

编 委 齐来金 孔 娟

王艳萍 王学军

王卫华 张立志



航空工业出版社

0056557

内 容 提 要

本书针对计算机应用能力等级考试(二级C语言)并严格按照考试大纲而编写。考虑到C终将被C++所取代以及可视化编程技术的发展,本书又用相当篇幅在C的基础上详细地介绍了C++程序设计语言,并对Visual C++ 6.0作了简介。

本书作为教材,主要供高校学生参加计算机应用能力等级考试学习使用,也可供本科少学时和专科计算机专业学生以及工程技术人员和广大计算机爱好者使用。

图书在版编目(CIP)数据

C和C++程序设计教程 / 赵建周, 杨庆祥主编. — 北京:
航空工业出版社, 2000.3
ISBN 7-80134-624-6

I.C… II. ①赵… ②杨… III.C语言-程序设计-
教材 IV.TP312

中国版本图书馆CIP数据核字(2000)第03732号

航空工业出版社出版发行

(北京市安定门外小关东里14号 100029)

北京云浩印刷厂印刷

全国各地新华书店经售

2000年3月第1版

2000年3月第1次印刷

开本: 787×1092 1/16

印张: 20.5

字数: 498千字

印数: 1-12000

定价: 26.80元

本社图书如有缺页、倒页、脱页、残页等情况, 请与本社发行部联系调换。联系电话: 010-82627676 或 64941995

前 言

C 语言是目前国际上流行的一种结构化、模块化、可编译的计算机高级程序设计语言。它既适合于系统程序的开发,也适合于应用程序的开发。由于它具有丰富的数据类型、简洁的表达式、极强的输入输出功能、灵活的数据构造能力、强有力的控制流程以及数值计算和符号处理能力、运行效率高、可移植性强等许多优点,受到了软件专家们的赞赏。目前我国各理工科院校绝大多数专业的教学以及工程技术人员的编程中得到广泛的应用,并且已被国家教委指定为计算机应用能力等级考试二级考试语言。

C++ 是一种多范型的程序设计语言,既支持面向过程的程序设计,又支持面向对象的程序设计,是学习 Visual C++ 的基础。C++ 作为 C 的超集,以 C 为支撑,对 C 进行了改进和扩充,添加了一些新的机制,是向面向对象领域进行的延伸。学了 C 再转向 C++ 会节省较多的时间,达到事半功倍的效果。

随着计算机技术的不断发展和我国计算机应用水平的逐渐提高,社会需求的人才所具备的知识结构也在不断变化,对大学毕业生原来所要求的“计算机 1 级、英语 4 级”正在被“计算机 2 级、英语 6 级”取代。本书正是在此背景下编写的。

本书共分三部分:

第一部分, C 程序设计(本书的前 10 章内容)。该部分涵盖了二级考试大纲规定的全部内容,包括 C 语言结构、数据类型及其运算、基本语句及流程控制、编译预处理、函数、数组、指针、位运算、结构体、联合和文件基本操作等内容。

第二部分, C++ 程序设计(本书的第 11 章至第 14 章内容)。随着计算机技术的快速发展和应用水平的不断提高, C 终将为 C++ 所淘汰。不久就可能开考 C++ 程序设计,故我们编写了该部分内容。在内容组织安排上,采用第一部分自成一体,再由第一部分自然过渡到第二部分的方案。经多次教学实践证明和学生普遍反映,这种安排更为合理,读者(学生)更易接受 C++ 面向对象的程序设计思想。

第三部分, Visual C++ 6.0 基础(本书的第 15 章至第 18 章内容)。目前,可视化编程技术非常流行,发展很快,其中 Visual C++ 已逐步成为主要的编程工具。为了使读者(学生)在有了 C 和 C++ 程序设计基础后,能更快地掌握这一流行技术,特安排了这部分内容。

本书是在充分总结教学经验的基础上,遵照普及实用的原则进行编写的。书中所含内容极其丰富,同时在内容的组织上立意创新,遵循一般读者的常规思维编排章序,简明而又精细。此外,在本书中所出现和使用的符号,如 +、-、*、<、=、\ 等大部分采用中文形式下编排,但是读者在编程时,均应使用西文形式下的符号(注释除外),特此说明。

本书由赵建周、杨庆祥主编。参编作者有:齐来金(第 1 章、第 3 章及附录 C 部分内容,约 3.7 万字);赵建周(第 2 章,约 4.2 万字);王学军(第 4 章、第 5 章及附录 A 部分内容,约 3.5 万字);郭新志(第 6 章、第 7 章,约 4 万字);张立志(第 8 章及附录 A 部分内容,约 3.6 万字);王艳萍(第 9 章及附录 C 部分内容,约 3.7 万字);王卫华(第 10 章、第 15 章、第 16 章及附录 B 部分内容,约 3.5 万字);靳瑞萍(第 11 章,约 4 万字);

李安伏（第12章及第11章习题，约4万字）；李卫平（第13章，约4万字）；孔娟（第14章及附录B部分内容，约3.6万字）；杨庆祥（第17章、第18章，约4.2万字）。

本书由北京卓越文化艺术有限公司总策划，在编写过程中得到了申斌教授和刘翔副教授的关心和大力帮助，在此谨向他们表示衷心的感谢。

鉴于编者的水平有限，加之编写时间仓促，书中难免有不妥之处，恳请读者批评指正。

编 者

2000年1月

目 录

第 1 章 C 语言概述	1
1.1 C 语言的发展历史简介	1
1.2 C 语言的特点	1
1.3 C 程序入门与一般形式	2
1.3.1 C 程序入门	2
1.3.2 C 源程序的结构和一般形式	5
1.4 C 程序的上机编辑、编译、连接和执行过程	7
1.4.1 编译型语言源程序在机器上运行的步骤	7
1.4.2 用 Turbo C 来运行 C 程序的步骤	7
习 题	10
第 2 章 数据类型、运算符与表达式	11
2.1 数据类型	11
2.1.1 C 的基本数据类型	11
2.1.2 类型修饰词	11
2.1.3 C 的其他数据类型	12
2.2 常量	12
2.2.1 数值常量	13
2.2.2 字符常量	13
2.2.3 字符串常量	14
2.2.4 符号常量	15
2.3 变量	15
2.3.1 整型变量	16
2.3.2 字符型变量	17
2.3.3 浮点型和双精度型变量	17
2.3.4 字符串变量	17
2.3.5 变量赋初值	19
2.4 标识符	19
2.5 运算符与表达式	20
2.5.1 算术运算符与算术表达式	20
2.5.2 关系运算符与关系表达式	21
2.5.3 逻辑运算符与逻辑表达式	22
2.5.4 增 1 和减 1 运算符	22

2.5.5	位运算符	23
2.5.6	赋值运算符和赋值表达式	24
2.5.7	逗号运算符和逗号表达式	25
2.5.8	条件运算符	25
2.6	混合运算及类型转换	26
2.6.1	对二元算术运算	26
2.6.2	对赋值运算	26
2.7	枚举类型及其说明	27
2.7.1	枚举类型及变量定义	27
2.7.2	枚举类型变量可参与的操作	28
2.8	输入/输出方式	29
2.8.1	字符输入库函数 <code>getchar</code> 和输出库函数 <code>putchar</code>	29
2.8.2	字符串的输入函数 <code>gets</code> 和输出函数 <code>puts</code>	29
2.8.3	有格式的输入函数 <code>scanf</code> 和输出函数 <code>printf</code>	30
2.9	编程风格	36
习 题		37
第 3 章	变量的存储类别	39
3.1	变量的存储类别概述	39
3.2	局部变量	40
3.3	全局变量	41
3.3.1	全局变量及其定义和作用范围	41
3.3.2	全局变量的有关说明	42
3.4	寄存器变量	46
3.5	静态存储变量	47
3.5.1	局部静态存储变量	47
3.5.2	全局静态变量	48
3.6	动态存储变量	49
3.7	变量存储类别小结	49
习 题		51
第 4 章	语句和流程控制	52
4.1	最简单的语句	52
4.2	复合语句	52
4.3	if 语句 (条件语句)	54
4.3.1	if 语句的一般形式	54
4.3.2	用 if 语句实现多分支判定结构	55
4.4	循环语句	56
4.4.1	while 循环语句	56

4.4.2 do-while 语句.....	56
4.4.3 for 循环语句.....	57
4.4.4 goto 语句和带标号的语句.....	58
4.5 break 语句(中止语句).....	59
4.6 switch 语句(开关语句).....	60
4.7 continue 语句(继续语句).....	62
4.8 return 语句(返回语句).....	62
4.9 循环的退出及死循环的避免.....	63
4.9.1 循环的退出.....	63
4.9.2 死循环问题.....	63
4.10 程序举例.....	64
习 题.....	65
第 5 章 预处理	67
5.1 宏替换.....	67
5.1.1 不带参数的宏.....	67
5.1.2 带参的宏.....	69
5.2 文件包含.....	70
5.3 条件编译.....	70
习 题.....	71
第 6 章 函数	73
6.1 函数定义.....	73
6.1.1 函数定义的一般形式.....	73
6.1.2 函数参数的使用及说明.....	74
6.1.3 函数的返回值.....	75
6.2 函数调用.....	76
6.2.1 函数调用的一般形式.....	76
6.2.2 对被调用函数的说明.....	77
6.3 函数的嵌套调用及递归.....	78
6.4 内部函数和外部函数.....	84
6.4.1 内部函数.....	84
6.4.2 外部函数.....	84
6.5 函数的应用.....	84
习 题.....	85
第 7 章 数组	86
7.1 一维数组的定义及引用.....	86

7.1.1	一维数组的定义	86
7.1.2	关于一维数组的几点说明	86
7.1.3	一维数组元素的引用	86
7.1.4	一维数组的初始化	87
7.1.5	一维数组的应用举例	87
7.2	二维数组的定义和引用	89
7.2.1	二维数组的定义	89
7.2.2	二维数组定义中的几点说明	89
7.2.3	二维数组的引用	90
7.2.4	二维数组初始化	90
7.2.5	二维数组应用举例	91
7.3	字符数组	92
7.3.1	字符数组的定义	92
7.3.2	字符数组的初始化	92
7.3.3	字符数组、字符串及字符三者间的关系	92
7.3.4	字符数组的输入、输出	93
7.3.5	字符串处理函数	93
7.3.6	字符数组的应用	95
习 题		96

第 8 章 指针

8.1	指针的概念	98
8.2	变量与指针	100
8.2.1	指针变量的定义	101
8.2.2	指针变量的引用	101
8.3	指针与数组	103
8.3.1	指向数组(元素)的指针变量的定义与赋值	103
8.3.2	通过指针引用数组元素	104
8.3.3	指向多维数组的指针	106
8.3.4	字符指针	108
8.3.5	指针数组	108
8.4	指针与函数	110
8.4.1	指向函数的指针变量的定义及使用	110
8.4.2	用指针类型数据作函数参数	112
8.4.3	带参的主函数	114
8.4.4	返回指针的函数	115
8.5	指向指针的指针	116
习 题		118

第9章 位运算、结构体和联合 119

9.1	位运算.....	119
9.1.1	按位取反运算符.....	119
9.1.2	按位与运算符.....	120
9.1.3	按位或运算符.....	121
9.1.4	位异或运算符.....	121
9.1.5	左移运算符.....	122
9.1.6	右移运算符.....	123
9.1.7	计算机中的几种码制.....	123
9.2	结构体.....	124
9.2.1	结构体的概念.....	124
9.2.2	结构体变量的定义.....	125
9.2.3	结构体变量的引用.....	126
9.2.4	结构体变量的初始化.....	127
9.2.5	结构体数组.....	128
9.2.6	指向结构体的指针.....	129
9.3	链表.....	131
9.3.1	链表的概述.....	131
9.3.2	链表操作.....	132
9.4	联合.....	138
习 题		139

第10章 文件 140

10.1	文件概述.....	140
10.2	文件的打开与关闭.....	141
10.2.1	打开文件函数.....	141
10.2.2	关闭文件函数.....	142
10.3	文件的读写.....	143
10.3.1	字符的输入输出函数.....	143
10.3.2	字符串的输入输出函数.....	144
10.3.3	数据块的输入输出函数.....	145
10.3.4	指定文件的输入输出函数.....	145
10.3.5	文件定位函数.....	146
10.4	出错检测函数.....	147
10.4.1	ferror 函数.....	147
10.4.2	clearerr 函数.....	147
习 题		147

第 11 章 转向 C++	148
11.1 C++ 的特点	148
11.2 从 C 到 C++ 的转变	149
11.2.1 标识符、关键字	149
11.2.2 声明语句	149
11.2.3 运算符	150
11.2.4 特有的输入输出语句	152
11.2.5 函数原型与函数声明	153
11.2.6 存储类	154
11.2.7 结构体	154
11.2.8 Sizeof 运算符	155
11.3 C++ 新特征	155
11.3.1 注释	156
11.3.2 常量类型	156
11.3.3 显式类型转换	157
11.3.4 作用域分辨符	157
11.3.5 内联函数和宏	159
11.3.6 参数缺省	160
11.3.7 引用类型	162
11.3.8 指针	166
11.3.9 函数重载	168
11.3.10 类型参数化	170
11.3.11 常量与函数	173
11.3.12 动态内存分配	174
习 题	176
第 12 章 类与对象	181
12.1 定义一个类	181
12.1.1 类的定义	181
12.1.2 定义成员函数	183
12.2 对象	184
12.2.1 对象声明	184
12.2.2 构造函数与析构函数	185
12.2.3 类、对象的内存分配	190
12.2.4 静态类数据	191
12.2.5 源文件的组织	193
12.3 对象行为与组织	193
12.3.1 友元	193

12.3.2 对象组织	196
习 题	199
第13章 继承	201
13.1 派生类	201
13.1.1 派生类声明	201
13.1.2 友元与继承	204
13.1.3 派生类的构造函数与析构函数	205
13.2 多基派生	209
13.3 虚基类	210
13.4 虚函数	211
13.4.1 方法的多态性与虚函数	211
13.4.2 虚函数的访问	215
13.4.3 纯虚函数与抽象类	219
13.4.4 多基派生中虚函数的二义性	221
习 题	224
第14章 重载与转换	226
14.1 概述	226
14.2 特殊运算符的重载	230
14.2.1 重载复合运算符	230
14.2.2 重载“=”运算符	232
14.2.3 重载“++”和“--”运算符	234
14.2.4 重载比较运算符	236
14.2.5 重载插入符和提取符	237
14.2.6 重载运算符[]	240
14.2.7 重载运算符()	241
14.3 转换	243
14.3.1 不同数据类型之间的转换	243
14.3.2 不同类类型之间的转换	245
习 题	247
第15章 Visual C++ 概述	249
15.1 Visual C++的发展史	249
15.2 Visual C++ 6.0 的安装	249
第16章 Visual C++6.0 基本操作	252
16.1 Visual C++6.0 集成开发环境的外观和基本操作	252

16.1.1	Visual C++6.0 集成开发环境的组成元素	252
16.1.2	正文窗口及其操作	253
16.1.3	项目工作台窗口及其操作	254
16.1.4	输出窗口及其操作	256
16.2	利用 Visual C++ 6.0 编写应用程序的一般过程	256
16.2.1	新建项目	256
16.2.2	添加代码	258
16.2.3	添加资源	260
16.2.4	编译、连接和调试	262
第 17 章	资源及资源编辑器简介	263
17.1	资源	263
17.1.1	资源的基本概念	263
17.1.2	资源符号	263
17.2	资源编辑器	264
17.2.1	对话框编辑器	265
17.2.2	菜单编辑器	267
17.2.3	图形编辑器	270
17.2.4	工具栏编辑器	271
17.2.5	加速键表编辑器	273
17.2.6	字符串表编辑器	274
17.2.7	版本信息编辑器	275
第 18 章	Windows 编程基础	276
18.1	创建 Windows 应用程序的基本框架	276
18.2	Windows 编程基础	281
18.2.1	Windows 的工作机制	281
18.2.2	程序的控制流程	281
18.2.3	基本的数据类型	284
18.3	Class Wizard 和 WizardBar 的使用	285
18.3.1	WizardBar 简介	285
18.3.2	Class Wizard 的使用	286
附录 A	Turbo C 的上机过程	290
A.1	运行 Turbo C 时的硬件配置及安装	290
A.1.1	配置	290
A.1.2	安装	290
A.2	Turbo C 的使用	290

A.2.1	进入 Turbo C.....	290
A.2.2	选择工作目录.....	291
A.2.3	建立工作环境.....	292
A.2.4	编辑源文件.....	293
A.2.5	编译和连接.....	294
A.2.6	运行.....	296
附录 B	Borland C++ 集成开发环境.....	297
B.1	安装与启动.....	297
B.1.1	Borland C++ 的安装.....	297
B.1.2	Borland C++ 的启动与退出.....	298
B.2	窗口操作.....	298
附录 C	文本、窗口及绘图.....	302
C.1	字符显示模式及控制.....	302
C.2	字符显示属性及控制.....	303
C.3	开窗口和定位.....	303
C.4	与窗口输入/输出相关的一组函数.....	304
C.5	Turbo C 绘图.....	304
C.5.1	图形显示模式.....	304
C.5.2	图形系统控制.....	305
C.5.3	色彩控制.....	307
C.5.4	绘图和着色.....	307
C.5.5	图形模式下的文本输出.....	309
C.5.6	图形模式下的定位.....	309

第1章 C语言概述

C语言是目前国际上流行的一种结构化、模块化、可编译的计算机高级程序设计语言。它既适合于系统程序的开发，也适合于应用程序的开发。例如，著名的 Unix 操作系统、微机上流行的 C-DBASEIII 等都是用 C 语言开发的。很多图形软件的接口应用程序，以及目前被人们广泛研究的神经网络应用软件等大都采用 C 语言来编写。数据处理（如企业管理）以及数值计算等都可以很方便的使用 C 语言。C 语言几乎无所不能。

1.1 C语言的发展历史简介

C语言最初是在 1972 年由 Dennis Ritchie 为 DEC 公司 PDP-11 计算机配置 Unix 操作系统而设计的，这个操作系统以及所有的基本 Unix 应用程序都用 C 语言编写而成。虽然 C 语言是在 Unix 系统上发展起来的，但它并不局限于某一种操作系统或机器。用 C 语言编写的程序可以很方便地移植到任何支持 C 语言的机器上。

C 语言既具有高级语言的所有特点，也具有实现汇编语言的某些功能（如对内存地址的操作、位操作、端口访问等直接对硬件进行操作的功能），因而既克服了用汇编语言编写的程序可读性差和可移植性差的缺点，同时又增强了高级语言的处理能力。

随着微型计算机的流行和发展，C 语言也逐渐成为微机上颇有影响的一种程序设计语言。目前在微机上流行使用的有 Microsoft C、Turbo C 和 Quick C 等。各种版本的 C 语言编译系统的基本部分是相同的，但也有一些细微的差异，因此，读者在选用了自己的 C 语言编译系统后，应了解自己所选用的计算机系统的特点和规定（可以参考有关的手册）。本书 C 语言部分主要以 Turbo C 为蓝本来介绍 C 语言的使用。

1.2 C语言的特点

C 语言能在短短的二三十年内得到广泛的发展和流行应用，显示出其强大的生命力，主要是由于它具有以下独到的特点：

(1) 以英文小写字母为基础。这符合人们日常阅读和书写的习惯。同时在不使用小写字母的微型机上允许使用大写字母，使用的范围不受影响。

(2) 程序语言表达简洁、紧凑，书写格式自由，使用方便灵活，因而程序短小精悍，让人易写易懂。

(3) 程序由函数组合而成，因此程序功能结构比较清楚。而且每个函数都是独立的，可以单独编译，对设计一个大的程序来说，有利于分工编译和调试。

(4) 运算符丰富，有利于编写程序。它把括号、赋值、强制类型转换等都作为运算符来处理，并增加了增 1（++）、减 1（--）和条件运算等其他高级语言没有的运算符，因而使运算符多达 34 种以上。同时也使 C 语言的运算类型极其丰富，表达式类型多样化。我

们能够通过这些运算符的灵活使用, 缩短程序代码, 完成其他高级语言难以实现的运算。

(5) 数据类型多。C 语言的数据类型比 Fortran、Algol 等多, 与 Pascal 语言相当。具有整型、实型、字符型、数组类型、指针类型、结构体类型和联合类型等现代化语言所应有的各种数据类型。能够实现各种复杂的数据结构(如链表、栈、队列和树等)的运算。

(6) 把指针作为数据来处理, 且可用指针指向各类数据结构。

(7) 具有结构化程序设计所应有的控制语句(如 if-else 语句、while 语句、do-while 语句、switch 语句、for 语句), 是理想的结构化程序设计语言, 符合现代编程的风格和要求。

(8) 语法的限制不死板, 程序设计比较自由。例如, 整型、字符型以及逻辑型数据可以通用, 变量的类型限制不严格, 数组的下标越界不作检查等。因而要求程序员自己仔细检查程序, 保证其正确, 不要过分依赖 C 编译程序。允许设计者充分使用 C 语言的自由度。

(9) 输入输出功能通过函数程序来实现。

(10) 允许直接访问物理地址, 进行位(bit)操作, 可实现汇编语言的大部分功能, 可以直接对硬件进行操作。C 语言体现了高级语言的很多优越性。

(11) 可移植性好(与汇编语言相比)。用 C 语言编写的程序基本上不作修改就能用于各种型号的计算机和各种操作系统中。

(12) 通过预处理可以进行宏调用。使大型程序的设计可以分工编写, 因而是一种模块化的程序设计语言。

(13) 没有作为语言组成部分的文件类型, 但能通过操作系统完成对文件的操作, 等等。

C 语言的上述特点, 读者目前可能还不能深刻理解, 待学完了 C 语言的全部主要内容后, 再回过头来重温一下, 就会有比较深刻的体会。当然, 这里只是介绍 C 语言的一些主要特点。C 语言内部的其他特点, 将会在以后的各章中进一步介绍。

正是由于上述特点, 目前 C 语言愈来愈得到人们的广泛重视, 受到世界上软件专家们的高度赞赏。目前我国各理工科院校几乎所有专业的教学中都开设有 C 语言课程, 同时在工程技术人员编程中, C 语言愈来愈广泛地被使用, 并被专家们认为是软件的三种基本技能(高级语言 C、数据库程序设计语言和图形程序设计语言)之一。现在, C 语言已被国家教委指定为计算机应用能力等级考试二级考试语言。

当然, C 语言也有不足之处。例如, 运算符优先级太多、不便记忆、有些与常规不同、类型检验较弱及转换比较随便等, 所以不太安全。尽管如此, C 语言还不失为一个实用的通用程序设计语言。使用者更可以发挥 C 语言格式自由、限制较少、语言简洁等特点, 编制自己的各种应用程序。

1.3 C 程序入门与一般形式

1.3.1 C 程序入门

用 C 语言编写的程序称为 C 语言源程序, 或简称 C 程序。

C 语言的源程序是什么形式呢? 下面通过几个简单的例程来分析一下 C 源程序的特点。

例 1.1 打印一串字符的 C 程序。

```
main()
```

```
{
    printf("Hello, world!");
}
```

C 程序一般是由一个或多个函数组成，而这些函数可以保存在一个或多个源程序文件中。这些文件都以 .C 为扩展名。在一个 C 程序中必须有且只能有一个名为 main 的函数，称为主函数。在运行 C 程序时总是从 main 函数开始执行。函数名后的一对圆括号是函数的标志。其中 printf("Hello, world!") 语句将在显示屏幕上打印输出括号中双引号内的内容：

Hello, world!

例 1.2 通过打印字符串形成一幅图案的 C 程序。

```
main()
{
    printf("*****\n");
    printf("Welcome to C! \n");
    printf("*****\n");
}
```

本程序的作用是在终端屏幕上输出三行字符组成的图案：

```
*****
Welcome to C!
*****
```

其中，main() 是主函数。大括号 “{” 和 “}” 括起来的部分是函数体。本例中主函数体由三个输出语句组成，其中 printf 是 C 语言中输出函数（也称库函数），双引号内的字符串原样输出。而 '\n' 是换行符，当遇到 '\n' 时就换行。每条语句最后都有一个 “;” 号，不可缺少。我们还可将例 1.2 的程序改写成：

例 1.3

```
main()      /*打印星号的主函数*/
{
    void stars(void); /* void 是不返回任何值的函数说明*/
    stars();
    printf(" Welcome to C!");
    stars();
}

void stars() /*打印星号的函数*/
{printf("*****\n"); }
```

该程序的运行结果与例 1.2 相同。但是，它打印一行星号是由函数 stars() 来实现的。其中 /*...*/ 表示注释部分。注释部分仅供人阅读，在编译和运行时都不加理睬。注释可以放在程序中的任何位置。为了便于理解，我们用汉语进行注释，在不支持中文的纯英文版本下则只可以用英文或汉语拼音进行注释。

例 1.4 求两个整数之和。

```
main() /* 求两整数的和 */
```