

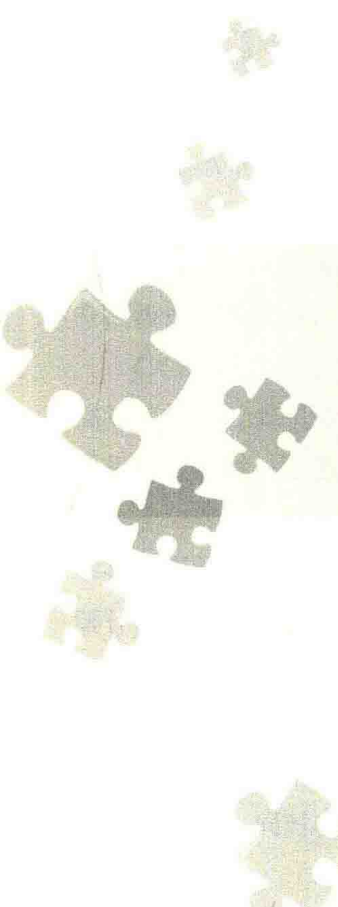
高等学校通识教育系列教材

C/C++案例教程

王朝晖 凌云 周克兰 张志强 编著



清华大学出版社



高等学校通识教育系列教材

C/C++案例教程

王朝晖 凌云 周克兰 张志强 编著

清华大学出版社
北 京

内 容 简 介

本书以 C/C++ 语言程序设计为蓝本阐述计算机程序设计的方法。全书内容由浅入深, 例题经典、丰富。全书共分 17 章, 每章内容包括知识要点、例题分析与解答、测试题和实验案例四部分。本书附录给出测试题的参考答案。

本书既可作为普通本科院校、普通高等专科学校的计算机程序设计实验教材, 也可作为相关教师 and 学生的参考用书。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目 (CIP) 数据

C/C++ 案例教程/王朝晖等编著. —北京: 清华大学出版社, 2019 (2019.8重印)
(高等学校通识教育系列教材)

ISBN 978-7-302-52438-0

I. ①C… II. ①王… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2019) 第 039284 号

责任编辑: 刘向威

封面设计: 文 静

责任校对: 焦丽丽

责任印制: 丛怀宇

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 北京九州迅驰传媒文化有限公司

经 销: 全国新华书店

开 本: 185mm×260mm

印 张: 24

字 数: 588 千字

版 次: 2019 年 4 月第 1 版

印 次: 2019 年 8 月第 2 次印刷

印 数: 2501~2700

定 价: 59.00 元

产品编号: 082998-01

前 言

C/C++语言是国内外广泛使用的计算机程序设计语言，其功能强，可移植性好，既具有高级语言的优点，又具有低级语言的特点，特别适合编写系统软件。

C/C++语言不仅受到计算机专业人士的喜爱，也受到非计算机专业人士的青睐。许多高等院校在计算机专业和非计算机专业都开设了“C/C++语言程序设计”课程。全国的计算机等级考试、江苏省的计算机等级考试以及其他各省的计算机等级考试都把C/C++语言列入了二级考试范围。为了帮助学生更快更好地掌握C/C++语言程序设计的特点，理解和掌握常用的程序设计的算法和思想，本书作者结合二十多年一线教学的实践经验，参照《全国计算机等级考试二级C/C++语言程序设计大纲》和《江苏省高等学校非计算机专业学生计算机知识与应用能力等级考试大纲》规定的二级C/C++语言考试要求编写了本书。

本书的最大特点是由易到难、循序渐进，列举了大量的典型题目，同时给出了详细的分析和解答。为了使读者能进一步自主进行强化训练，根据C语言的每个知识点给出相应的练习题目，同时在附录中也给出了正确的参考答案，方便读者判断自己解题正确与否，提高学习效率。

全书共分17章。每章知识要点部分对相应章节的重点内容进行了归纳和总结；在例题分析与解答部分列举了一些容易出错、具有一定难度的选择题和填空题，对其给予详尽的分析和解答；之后，为了强化和掌握本章知识，给出相关的测试题目和参考答案。在每章实验里，针对每个实验题目，都提出实验要求、给出算法提示，要求学生给出完整的代码；根据问题需要，提出了相关的思考问题，帮助学生更加深刻透彻地理解该实验的知识要点。如果初学者能够认真做好本书提供的每个题目，就能够掌握C/C++语言程序设计的基本要领和技巧，进而也就掌握了计算机程序设计的基本思想，通过国家和各省C/C++语言程序设计二级考试也就更加顺利了。

本书在编写过程中得到了苏州大学计算机公共教学部所有老师的大力支持和参与，他们提出了宝贵建议，在此表示衷心的感谢！

本书编写是由王朝晖、凌云、周克兰和张志强合作完成的，由张志强负责全书统稿。第1~12章由王朝晖编写；第13~15章由张志强编写；第16章由周克兰编写，第16章案例由张玉华提供；第17章由凌云编写。

本书在编写过程中参考了许多同行的著作，作者在此一并表示感谢。

本书的编写得到了江苏省高等教育教改研究课题（2017JSJG532）资助。感谢为本书提供直接或间接帮助的每一位朋友，你们的帮助和鼓励促成了本书的顺利完成。

尽管作者试图把本书写得更加完美，但由于时间和能力所限，书中难免会有错误和不当之处，恳请读者批评指正，以便下次再版或印刷时修订。

编 者

2018年11月

目 录

第 1 章 C 语言导论	1
1.1 知识要点	1
1.1.1 程序设计语言概述	1
1.1.2 C 语言的特点	2
1.1.3 C 语言程序的构成	2
1.1.4 C 源程序的编辑、编译、链接与执行	2
1.2 例题分析与解答	2
1.3 测试题	5
第 2 章 基本数据类型、运算符与表达式	7
2.1 知识要点	7
2.1.1 C 语言的数据类型	7
2.1.2 常量与变量	7
2.1.3 C 语言运算符	7
2.1.4 C 语言运算符的结合性和优先级	8
2.1.5 C 语言表达式	8
2.2 例题分析与解答	8
2.3 测试题	13
2.4 实验案例	15
第 3 章 顺序结构程序设计	18
3.1 知识要点	18
3.1.1 C 语句	18
3.1.2 C 语句分类	18
3.1.3 输入输出的实现	19
3.2 例题分析与解答	19
3.3 测试题	21
3.4 实验案例	23
第 4 章 选择结构程序设计	25
4.1 知识要点	25

4.1.1	关系运算符和关系表达式	25
4.1.2	逻辑运算符和逻辑表达式	25
4.1.3	if 语句	25
4.1.4	if 语句的嵌套	26
4.1.5	由条件运算符构成的选择结构	26
4.1.6	switch 语句和 goto 语句	26
4.2	例题分析与解答	27
4.3	测试题	31
4.4	实验案例	32
第 5 章	循环结构程序设计	35
5.1	知识要点	35
5.1.1	循环结构的 3 种形式	35
5.1.2	continue 语句和 break 语句	36
5.2	例题分析与解答	36
5.3	测试题	39
5.4	实验案例	45
第 6 章	数组	48
6.1	知识要点	48
6.1.1	数组的概念	48
6.1.2	一维数组的定义和引用	48
6.1.3	二维数组的定义和引用	48
6.1.4	字符数组的定义和引用	49
6.2	例题分析与解答	50
6.3	测试题	52
6.4	实验案例	60
第 7 章	函数	63
7.1	知识要点	63
7.1.1	函数的概念	63
7.1.2	函数的种类	63
7.1.3	函数定义的一般形式	63
7.1.4	函数的调用	64
7.1.5	C 语言中数据传递的方式	64
7.1.6	函数的嵌套调用和递归调用	64
7.1.7	全局变量和局部变量	64
7.1.8	变量的存储类别	65
7.1.9	内部函数和外部函数	65

7.2	例题分析与解答	65
7.3	测试题	71
7.4	实验案例	79
第8章	指针	87
8.1	知识要点	87
8.1.1	指针变量的定义	87
8.1.2	变量的指针和指向变量的指针变量	87
8.1.3	数组的指针和指向数组的指针变量	87
8.1.4	字符串的指针和指向字符串的指针变量	88
8.1.5	指针数组	88
8.1.6	函数的指针和指向函数的指针变量	88
8.1.7	用指针作函数参数	88
8.1.8	指向指针的指针	88
8.2	例题分析与解答	89
8.3	测试题	93
8.4	实验案例	99
第9章	结构体、共用体和枚举类型	107
9.1	知识要点	107
9.1.1	结构体的概念	107
9.1.2	结构体类型的定义	107
9.1.3	结构体类型成员的引用	107
9.1.4	结构体变量的指针和结构体指针变量	108
9.1.5	指向结构体数组的指针	108
9.1.6	共用体	108
9.1.7	typedef 的用法	109
9.1.8	枚举类型	109
9.2	例题分析与解答	110
9.3	测试题	112
9.4	实验案例	116
第10章	编译预处理	121
10.1	知识要点	121
10.1.1	编译预处理概述	121
10.1.2	宏定义	121
10.1.3	文件包含	121
10.1.4	条件编译	121
10.2	例题分析与解答	122

10.3	测试题	125
10.4	实验案例	128
第 11 章	内存的使用	131
11.1	知识要点	131
11.1.1	动态使用内存	131
11.1.2	链表的概念	131
11.1.3	链表的建立	132
11.1.4	链表的查找与输出	132
11.1.5	释放链表	132
11.2	例题分析与解答	133
11.3	测试题	134
11.4	实验案例	138
第 12 章	文件	143
12.1	知识要点	143
12.1.1	文件的概念	143
12.1.2	文件类型指针	143
12.1.3	文件操作	143
12.2	例题分析与解答	144
12.3	测试题	147
12.4	实验案例	148
第 13 章	C++对 C 的扩充	153
13.1	知识要点	153
13.1.1	C++语言概述	153
13.1.2	C++的数据类型	154
13.1.3	C++的运算符	154
13.1.4	C++的名字空间	154
13.1.5	C++的输入输出	154
13.1.6	C++的函数与模板	155
13.1.7	C++的异常处理	155
13.2	例题分析与解答	155
13.3	测试题	164
13.3.1	测试题 1	164
13.3.2	测试题 2	169
13.4	实验案例	175
13.4.1	案例 1——函数重载	175
13.4.2	案例 2——函数模板	176

第 14 章	基于 C++ 的面向对象编程	179
14.1	知识要点	179
14.1.1	类和对象	179
14.1.2	类的继承与派生	180
14.1.3	类的多态性	181
14.1.4	类模板与泛型编程	182
14.2	例题分析与解答	182
14.3	测试题	194
14.3.1	测试题 1	194
14.3.2	测试题 2	199
14.4	实验案例	204
14.4.1	案例 1——类和对象	204
14.4.2	案例 2——继承和派生	207
14.4.3	案例 3——类的多态性	210
14.4.4	案例 4——类模板与泛型编程	213
第 15 章	基于 MFC 的 Windows 编程	216
15.1	知识要点	216
15.1.1	MFC 基础	216
15.1.2	MFC 控件	217
15.1.3	MFC 绘图	217
15.1.4	常用消息	218
15.2	例题分析与解答	219
15.3	测试题	226
15.3.1	测试题 1	226
15.3.2	测试题 2	231
15.4	实验案例	236
15.4.1	案例 1——MFC 控件	236
15.4.2	案例 2——MFC 绘图	240
15.5	综合案例	245
15.5.1	综合案例——游戏软件制作	245
15.5.2	综合案例——绘图软件制作	256
第 16 章	编程技术基础	271
16.1	知识要点	271
16.1.1	基本数据结构与算法	271
16.1.2	程序设计基础	275
16.1.3	软件工程基础	276

16.1.4	数据库设计基础	277
16.2	例题分析与解答	278
16.3	本章测试	284
16.3.1	测试题 1	284
16.3.2	测试题 2	288
16.4	综合案例	291
16.4.1	综合案例 1——顺序表的实现和测试	291
16.4.2	综合案例 2——单链表的实现和测试	296
16.4.3	综合案例 3——栈的顺序实现和应用	302
16.4.4	综合案例 4——循环队列的实现和测试	305
第 17 章	全国二级考试模拟	309
17.1	C 语言程序设计考试大纲 (2018 年版)	309
17.1.1	基本要求	309
17.1.2	考试内容	309
17.1.3	考试方式	311
17.2	第一套试题	311
17.3	第二套试题	319
17.4	第三套试题	330
17.5	第四套试题	341
17.6	第五套试题	351
附录 A	测试题参考答案	360
参考文献		374

1.1 知 识 要 点

1.1.1 程序设计语言概述

1. 程序设计语言的发展

程序设计语言的发展经历了以下几个阶段。

1) 机器语言

机器语言是直接由二进制代码指令表达的计算机语言，指令是由 0 和 1 组成的一串代码。用机器语言编写的程序可以被机器直接执行，但不直观，且难记、难理解、不易掌握。

2) 汇编语言

汇编语言是用一些助记符号来代替机器语言中由 0 和 1 所组成的操作码，如 ADD、SUB 分别代表加、减等。用汇编语言编写的程序不能被机器直接执行，要翻译成机器语言程序才能执行。

汇编语言和机器语言都依 CPU 的不同而异，统称为面向机器的语言。

3) 高级语言

高级语言接近自然语言和数学语言，是不依赖任何机器的一种容易理解和掌握的语言。

用高级语言编写的程序称为“源程序”。源程序不能在计算机上直接运行，必须将其翻译成由 0 和 1 组成的二进制程序才能执行。翻译过程有两种方式：一种是翻译一句执行一句，称为“解释执行”方式，完成翻译工作的程序称为“解释程序”；另一种是全部翻译成二进制程序后再执行，称为“编译执行”，完成翻译工作的程序称为“编译程序”，编译后的二进制程序称为“目标程序”。

2. 结构化的程序设计方法

结构化的程序设计方法强调程序结构的规范化，一般采用顺序结构、分支结构和循环结构 3 种基本结构。结构化的程序设计可以总结为“自顶向下、逐步细化”和“模块化”的设计方法。

“自顶向下，逐步细化”是指先整体后局部的设计方法。即先求解问题的轮廓，然后再逐步求精，是先整体后细节，先抽象后具体的过程。“模块化”是将一个大任务分成若干较小任务，即复杂问题简单化。每个小任务完成一定的功能，称为“功能模块”。各个功能

模块组合在一起就解决了一个复杂的大问题。

1.1.2 C 语言的特点

C 语言是一种结构紧凑、使用方便、程序执行效率高的编程语言，它有 9 种控制语句、32 个关键字和 34 种运算符。C 语言的主要特点如下：

- (1) 语言表达能力强。
- (2) 语言简洁、紧凑，使用灵活，易于学习和使用。
- (3) 数据类型丰富，具有很强的结构化控制性。
- (4) 语言生成的代码质量高。
- (5) 语法限制不严格，程序设计自由度大。
- (6) 可移植性好。

C 语言的 32 个关键字如下：auto, break, case, char, const, continue, default, double, else, enum, extern, float, for, goto, int, long, register, return, short, signed, sizeof, do, if, static, struct, switch, typedef, union, unsigned, void, volatile 和 while。

1.1.3 C 语言程序的构成

- (1) C 语言的源程序是由函数构成的，每一个函数完成相对独立的功能，其中至少必须包括一个 main() 函数。
- (2) C 程序总是从 main() 函数开始执行。
- (3) C 语言规定每个语句以分号结束，分号是语句组成不可缺少的部分。
- (4) 程序的注释部分应括在 /* 与 */ 之间，注释部分可以出现在程序的任何位置。

1.1.4 C 源程序的编辑、编译、链接与执行

C 语言的源程序必须先由源文件 (.c) 经编译生成目标文件 (.obj)，再经过链接方可生成可执行的文件 (.exe)。

1.2 例题分析与解答

一、选择题

1. 以下叙述中正确的是_____。
- A. 程序设计的任务就是编写程序代码并上机调试
- B. 程序设计的任务就是确定所用数据结构
- C. 程序设计的任务就是确定所用算法
- D. 以上 3 种说法都不完整

分析：程序设计的任务是根据实际需求，设计解决问题的算法和所用的数据结构，然后编写程序代码并上机调试，最终完成解决实际问题的计算机程序。

答案：D

2. C 语言源程序文件名的扩展名是_____。

A. .exe

B. .c

C. .obj

D. .cpp

分析: C 语言源程序的扩展名是.c 或.C; 扩展名为.exe 的文件是可执行文件; 扩展名为.obj 的文件是目标文件; C++源程序扩展名为.cpp。

答案: B

3. 以下叙述错误的是_____。

A. C 语言源程序文件经编译后生成扩展名为.obj 的目标程序文件

B. C 语言源程序经过编译、链接步骤之后才能生成一个真正可执行的二进制机器指令文件

C. 用 C 语言编写的程序称为源程序, 它以 ASCII 码形式存放在一个文本文件中

D. C 语言中的每条可执行语句和非执行语句最终都将被转换成二进制的机器指令

分析: C 语言源程序经过编译后生成.obj 目标程序; C 程序经过编译、链接后才能形成一个可执行的二进制机器指令文件; 用 C 语言编写的程序称为源程序, 以 ASCII 码形式存放在一个文本文件中, 如.c 文件; C 语言中的每条可执行语句将被转换成二进制的机器指令; 非执行语句不能被转换成二进制的机器指令。

答案: D

4. 一个 C 程序的执行是从_____。

A. 本程序的 main 函数开始, 本程序的 main 函数结束

B. 本程序的第一个函数开始, 本程序的最后一个函数结束

C. 本程序的 main 函数开始, 本程序的最后一个函数结束

D. 本程序的第一个函数开始, 本程序的 main 函数结束

分析: 一个 C 程序总是从 main 函数开始执行的, 而不论 main 函数在整个程序中的位置如何, main 函数可以放在程序的最前头, 也可以放在程序最后, 或在一些函数之前, 在另一些函数之后。一个 C 程序的结束也是在本程序的 main 函数中结束。

答案: A

5. 以下叙述不正确的是_____。

A. 一个 C 源程序可由一个或多个函数组成

B. 一个 C 源程序必须包含一个 main 函数

C. C 程序的基本组成单位是函数

D. 在 C 程序中, 注释说明只能位于一条语句的后面

分析: 在 C 语言中, /*...*/表示注释部分, 为便于理解, 常用汉字表示注释, 当然也可以用英语或拼音作为注释。注释是给人看的, 对编译和运行不起作用。注释可以加在程序中的任何位置。

答案: D

6. C 语言规定, 在一个源程序中, main 函数的位置_____。

A. 必须在最开始

B. 必须在系统调用的库函数的后面

C. 可以在任意位置

D. 必须在最后

分析：一个 C 程序至少包含一个 `main()` 函数，也可以包含一个 `main` 函数和若干其他函数。`main` 函数可以在整个程序中的任意位置，可以放在程序的最前头，也可以放在程序最后；或在一些函数之前，在另一些函数之后。

答案：C

7. 一个 C 语言程序由_____。

- A. 一个主程序和若干子程序组成
- B. 函数组成
- C. 若干过程组成
- D. 若干子程序组成

分析：C 程序是由函数构成的。一个 C 程序至少包含一个 `main` 函数，也可以包含一个 `main` 函数和若干其他函数。因此，函数是 C 程序的基本单位。被调用的函数可以是系统提供的库函数（如 `printf` 和 `scanf` 函数），也可以是用户根据需要自己编写的函数（自定义函数）。C 的函数相当于其他语言中的子程序。

答案：B

二、填空题

1. 用高级语言编写的源程序必须通过_____程序翻译成二进制程序才能执行，这个二进制程序称为程序。

分析：用高级语言编写的源程序有两种执行方式：一是利用“解释程序”，翻译一条语句，执行一条语句，这种方式不会产生可以执行的二进制程序，例如 BASIC 语言；二是利用“编译程序”一次翻译形成可以执行的二进制程序，例如 C 语言。凡是编译后生成的可执行二进制程序都称为“目标程序”。

答案：编译目标

2. C 源程序的基本单位是_____。

分析：C 程序是由函数构成的。一个 C 程序至少包含一个 `main` 函数，也可以包含一个 `main` 函数和若干其他函数。因此，函数是 C 程序的基本单位。

答案：函数

3. 一个 C 源程序中至少应包括一个_____。

分析：一个 C 程序至少包含一个 `main` 函数，也可以包含一个 `main` 函数和若干其他函数。

答案：`main` 函数

4. 在一个 C 源程序中，注释部分两侧的分界符分别为_____和_____。

分析：在 C 语言中，用 `/*` 和 `*/` 括起来的内容表示注释内容，为便于理解，我们常用汉字表示注释，当然也可以用英语或拼音作注释。注释是给人看的，对编译和运行不起作用。

答案：`/*` `*/`

5. 在 C 语言中，输入操作是由库函数_____完成的，输出操作是由库函数_____完成的。

分析：在 C 语言中输入源数据用格式输入函数 `scanf` 来完成，而输出数据由 `printf` 来负责。语法格式见主教材。

答案：`scanf` `printf`

1.3 测试题

一、选择题

1. 以下叙述正确的是_____。
 - A. 用 C 程序实现的算法必须要有输入和输出操作
 - B. 用 C 程序实现的算法可以没有输出但必须要有输入
 - C. 用 C 程序实现的算法可以没有输入但必须要有输出
 - D. 用 C 程序实现的算法可以既没有输入也没有输出
2. 以下叙述错误的是_____。
 - A. 算法正确的程序最终一定会结束
 - B. 算法正确的程序可以有 0 个输出
 - C. 算法正确的程序可以有 0 个输入
 - D. 算法正确的程序对于相同的输入一定有相同的结果
3. 以下叙述正确的是_____。
 - A. C 程序是由函数构成的
 - B. C 程序是由过程构成的
 - C. C 程序由函数和过程构成
 - D. 一个 C 程序可以有多个 main 函数
4. 以下不是算法特点的是_____。
 - A. 有穷性
 - B. 确定性
 - C. 有效性
 - D. 有一个输入或多个输入
5. 表示一个算法, 可以用不同的方法, 不常用的有_____。
 - A. 自然语言
 - B. 传统流程图
 - C. 结构化流程图
 - D. ASCII 码
6. 以下不属于结构化程序设计特点的是_____。
 - A. 自顶向下
 - B. 逐步细化
 - C. 模块化设计
 - D. 使用无条件 GOTO 语句

二、填空题

1. 一个 C 程序是由一个主函数和若干_____组成的。
2. C 语言提供的合法关键字有_____个。
3. 在 C 语言程序中, 主函数的名字是_____。
4. C 程序的编译过程一般分成 5 个步骤: 编译预处理、_____, 优化、汇编和链接。
5. 在 C 语言程序中, 经常使用_____函数输入数据。
6. 在 C 语言程序中, 经常使用_____函数输出结果。
7. 把高级语言源程序翻译成等价的机器语言程序的软件被称为翻译程序或_____。
8. 用户编写的程序可能存在的错误有 3 大类, 分别是_____, 逻辑错误和运行错误。

9. _____ 是用户编写的程序违背了 C 语言的语法规则, 这些错误通常在程序编译、链接过程中可以发现。

三、编程题

1. 参照主教材例题, 编写一个 C 程序, 输出信息: Very good!。
2. 编写程序, 输入两个数, 计算这两个数的和, 并输出。