高等学校计算机专业教材精选·算法与程序设计

C语言程序设计与实训

余久久　编著

清华大学出版社

高等学校计算机专业教材精选·算法与程序设计

C语言程序设计与实训

余久久　编著

清华大学出版社
北京

内 容 简 介

本书根据应用型本科高校计算机类、信息类等相关工科专业开设的C语言程序设计课程的教学要求和特点编写，内容共分为10章，包括C语言概述、数据类型与运算、数据输入与输出、程序结构设计与应用、数组、函数、指针、结构体、文件、结构化程序设计与实训。全书以C语言的基本概念与基本知识为引领，从实际问题出发，以应用为基础，本着“理论适度，突出实训，增强职业素养”的原则，通过实训任务，由浅入深、循序渐进地引导读者学习与掌握C语言程序设计方法，激发学生学习兴趣，提高动手实践能力。

本书内容通俗易懂，理论适度，实践性强，适用面广。每章最后配有习题，作为对本章学习知识点的巩固，以方便学生复习与自学。

本书适合作为应用型本科高校、高职高专院校计算机及其相关专业的课程教材，也可以作为软件企业的职业培训类书籍以及各类软件技术人员的参考书。

图书在版编目(CIP)数据

C语言程序设计与实训/余久久编著. —北京：清华大学出版社，2016
(高等学校计算机专业教材精选·算法与程序设计)
ISBN 978-7-302-45299-7

Ⅰ. ①C… Ⅱ. ①余… 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(2016)第260847号

责任编辑：张　玥
封面设计：傅瑞学
责任校对：徐俊伟
责任印制：杨　艳

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
课件下载：http://www.tup.com.cn,010-62795954

印 刷 者：北京富博印刷有限公司
装 订 者：北京市密云县京文制本装订厂
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：20　　**字　　数**：490千字
版　　次：2016年12月第1版　　**印　　次**：2016年12月第1次印刷
印　　数：1～2000
定　　价：39.50元

产品编号：067407-01

前　言

C语言是目前广泛使用的一种高级程序设计语言，也是国内外高校讲述程序设计方法的首选语言。“C语言程序设计”已成为计算机类课程体系中的第一门重要的基础课程。该课程实践性较强，需要进行大量的上机操作与实训。在实践中发现问题、分析问题、解决问题，才能更好地掌握C语言，并最终学会利用C语言解决实际问题。

目前市面上出版的C语言程序设计类教材比较多，所介绍的理论知识及其应用案例也很全面。很多教材提倡项目(案例)教学思路，把一个或多个真实的软件项目(案例)及其运作流程从头至尾融入C语言课程教学中，通过项目驱动逐一介绍C语言的基本知识内容，以培养学生的工程实践能力，提高学生的动手技能。这种教学思路具有新颖性，但前提是要对所教授学生的实际认知状况进行一个合理的评估。对于一些地方性的应用型本科高校或高职院校层次的学生而言，学习C语言之前大都缺乏一定深度的计算机与软件方面的专业基础知识，加之数学知识较薄弱，又没有实际项目开发经验，因此会觉得课程内容空洞乏味，理解起来有一定难度。编者长期在应用型本科高校一线从事C语言课程的教学及指导实践工作，所在高校的学生在大一阶段学习C语言课程之前，绝大多数都是“程序设计零基础”学习，更不用说具备软件项目实践经历。以各种实际软件项目(案例)为驱动，去讲解C语言的各种语法知识，反而使大多数学生感到课程知识枯燥无味，因为课后还要专门查阅一系列后续计算机类专业课程的资料辅助学习，从而加重了学习负担。在多年的教学实践中，编者能亲身感受到“零基础”学生在学习C语言过程中产生的困惑。很多学生学习时会出现“课堂上老师一讲就懂，上机调试程序总是一调就错，自己又找不出原因”的窘况，反而降低了学习积极性。例如，上机调试程序时，很多学生分不清C语言中=与==、1/3与1/3.0的区别，会把C语言表达式if(x>2&&x<3)错写成if(2<x<3)，把p=p*n;错写成p=pn;等情况，导致程序运行异常。归根结底，还是课堂上对C语言的基本语法知识没有掌握扎实，课后又未能及时复习，也没有通过上机实训环节对所学知识进行巩固造成的。所以，编者认为，针对应用型本科学生，C语言课程的教学应定位于“理论知识适度，强调上机实训以强化基本概念，增强学生的职业素养”的教学思想，而对于项目驱动为主导的教学方式，则适合作为对日常教学活动的拓展。同样，仅通过校内几十个学时的学习，试图使“程序设计零基础”的大一学生具备良好的软件项目工程实践能力是不现实的，也是不科学的。因为良好的工程实践能力是需要学生今后通过专门的(校内、外)实习实训，以及在未来的工作岗位中逐渐培养与建立起来的。

基于此，针对应用型本科大一学生的认知特点，结合实际教学环境，编者通过对C语言的了解与感悟以及多年的教学实践，在对自己的备课讲义进行认真而系统的梳理后编写了本书。本书定位于“程序设计零基础”的读者，着眼于理论适度，突出实用。书中对C语言的主要知识内容进行了通俗易懂的讲解，每一个C程序实例则采用了较为简单的方法来实现，注重对C语言语法知识的掌握，而淡化程序算法设计思想，以便学生更容易理解程序的组织结构与功能。针对一些难理解、易混淆的知识点，书中为初学者指出了一些学习上的注

意事项，而这些大都来自于近几届学生在课堂学习、课后作业、上机操作等方面所出现的共性问题，希望读者学习时要引起足够重视。全书内容在编排上循序渐进，在一些重要章节的后面都安排了相应的实训任务，实训程序大都由相应章节中的例题所改写，并做适当拓展，目的是要求学生通过上机实训加强对本节所学内容知识的强化与巩固，从而做到举一反三，融会贯通。实训任务既可以安排在上机课内完成，也可以让学生课后自行上机完成。编者建议，学生一定要亲自调试书中的每一个 C 程序（包括例题程序与实训程序），认真观察与分析程序运行结果，而不能仅局限于“字面上看懂了程序”。最后，本书第 10 章通过一个简易的“万年历”程序案例，介绍如何使用 C 语言代码完成程序设计及项目实践过程。任课老师可以根据学时安排，把该案例作为选学内容或 C 语言课程设计的参考内容。

本书以 2014 年安徽省职业与成人教育学会教育科研规划项目、2015 年安徽省省级质量工程项目、2015 年安徽三联学院校级质量工程项目为依托，系项目研究成果之一。成书过程中，编者得到了安徽三联学院校领导的大力支持。此外，合肥工业大学张佑生教授、张正武教授与杜习英教授，安徽三联学院计算机工程学院操晓峰主任也为该书内容的编写提出了宝贵的建议，在此表示衷心的感谢。

本着学习与借鉴的目的，本书在编写过程中参考了大量同类 C 语言书籍及文献，在此谨向原作者表示诚挚的谢意。由于编者水平有限，加之时间仓促，书中的疏漏和不当之处在所难免，还望各位同行批评指正。

编　者

2016 年 8 月

目　录

第 1 章　C 语言概述

本章学习目标

- 计算机程序设计语言及其分类
- 程序设计的主要过程
- C 程序的组成结构及特征
- 良好的代码书写习惯
- Visual C++ 6.0 环境下运行 C 程序的方法

在当今世界的各个领域，计算机都得到了广泛的应用。从简单的上网冲浪到复杂的网络游戏开发，从个人网上购物到我国庞大的“嫦娥三号”探月工程，都少不了计算机的身影。计算机类专业的大学生不能只满足于会使用办公软件，而要学习计算机程序设计的知识，能够熟练编写出本专业领域中的相关应用程序，掌握运用计算机处理现实问题的方法，培养分析与解决问题的能力。

1.1　计算机程序设计

程序是用来解决某个问题的方法及步骤的具体描述。程序设计是以某种程序设计语言为工具，利用这种语言设计出程序。程序设计过程通常包括分析、设计、编码、测试、维护等一系列阶段。专业的程序设计人员也称为程序员。

1.1.1　计算机程序

从计算机的角度来看，程序是用计算机语言来描述解决问题的方法及步骤，也称为计算机程序。计算机程序通常用某一种程序设计语言来编写，编写出一组基本操作（指令）的组合，来对数据进行处理，并运行于某种应用平台上，以达到问题求解的目的。比如可以把一个计算机程序形象地比喻成用中文（一种计算机程序设计语言）撰写出的一个做酸菜鱼的菜谱（程序），用来指导厨师（程序员）使用炊具及调味作料（应用平台）来烹饪这一道菜。可见，计算机程序的执行过程也就是某一具体问题的求解过程。程序的执行是有始有终的，并且每一个步骤都能够操作，当所有步骤执行完后，程序对应的问题也就迎刃而解了。所以，要想解决问题，首先要设计出解决问题的正确方法与具体实施步骤。

1.1.2　计算机程序设计语言及其分类

计算机程序设计语言简称计算机语言，用来编写相应的计算机程序，是人与计算机交流的工具。根据计算机程序设计语言的发展阶段，其可以分为三类，即机器语言、汇编语言与高级语言。

1. 机器语言

机器语言是由计算机二进制代码 0 与 1 表示的一组机器指令的集合，具有灵活、直接执

行和执行速度快等特点，能够被计算机直接识别。但是，机器语言的阅读性差，理解起来困难，且编写效率很低，极易出错。下面举一个例子，如例 1.1 所示。

例 1.1 已知 a=2，b=3，利用机器语言编写出一个 a=a+b 的程序，也就是说最终 a 的数值等于 5。

利用机器语言编写“a=a+b”的代码，其中 a=2，b=3。

```
#01: 00000000 00000000 00000000 00000010
#02: 00000000 00000000 00000000 00000011
#03: 00000000 00000000 00000000 00000101
```

代码解释：

#01：令 a=2。

#02：令 b=3。

#03：将 a 和 b 的值相加，并将结果放在 a 中，即 a=5。

可见，机器语言与人们习惯用的语言差别很大，难学、难记、难理解，编写效率低，难以推广，容易出错。只有计算机专家或资深专业人员才使用机器语言直接设计程序。

2. 汇编语言

汇编语言对机器语言进行了符号化处理，增加了一些由英文字母或数字表示的指令、助记符等表示对机器语言某些功能上的操作，以方便记忆。例如，用符号 ADD 表示“相加”，SUB 表示“相减”，MOV 表示“传送”等。使用汇编语言设计出上述同样的程序，如例 1.2 所示。

例 1.2 利用汇编语言编写 a=a+b 的程序，其中 a=2，b=3。

```
#01: MOV R1,#2
#02: MOV R2,#3
#03: ADD R1,R1,R2
```

代码解释：

#01：把数值 2 放入寄存器 R1 中。

#02：把数值 3 放入寄存器 R2 中。

#03：把寄存器 R1 中的数值 2 与寄存器 R2 中的数值 3 相加，得到数值 5，再把数值 5 放入寄存器 R1 中。

可见，尽管汇编语言比机器语言可读性好一些，但是也难以普及，仅限于专业人员使用。

汇编语言不能直接被计算机识别，需要使用一种称为“汇编程序”的软件，把汇编语言程序转化成能够被计算机直接识别的机器语言，这个识别过程称为“汇编”。最后，在不同型号计算机上设计出的汇编语言程序也不具有通用性，即在型号甲的计算机上编写出的汇编语言不一定能在型号乙的计算机上使用。所以，汇编语言的层次很低，仅面向具体的计算机（机器），离计算机硬件很贴近，也被称之为“低级语言”。

3. 高级语言

高级语言是一种接近自然语言、阅读起来符合人类思维习惯的程序设计语言，类似于数学语言，具有良好的通用性和可移植性，不依赖于具体的计算机（机器）类型。常见的高级语言有 FORTRAN、Pascal、C、C++ 、Java、C# 等，现代应用程序的设计大多使用高级语言。

例 1.3 利用高级语言编写 a=a+b 的程序，其中 a=2，b=3。

```
main()
{
    int a,b;                    //语句 1;
    a=2;                        //语句 2;
    b=3;                        //语句 3;
    a=a+b;                      //语句 4;
    printf("a=%d\n",a);         //语句 5;
}
```

注：例 1.3 是用高级语言中的 C 语言编写出的程序。

语句 1：定义两个整数型的变量，取名为 a 和 b。

语句 2：把整数 2 赋给 a。

语句 3：把整数 3 赋给 b。

语句 4：计算 a+b 的值，并且把得到的结果再赋给变量 a。

语句 5：以十进制整数的形式输出 a 的值。

可见，高级语言读写起来更接近人们的思维习惯，理解性强，通用性良好。当然，使用高级语言编写出的程序不能被计算机直接识别并执行，需要进行“翻译”。大多数高级语言使用一种称为“编译程序”的软件，把用高级语言事先编写好的程序（源程序）转化为所对应的机器指令程序（目标程序），让计算机执行机器指令程序，得到最终结果。

按照人们设计程序时所采用的思维方式，高级语言又分为以下两类：

（1）面向过程的高级语言

面向过程就是以拟解决的问题为思考核心，使用计算机程序设计逻辑，描述需要解决的问题及其解决方法。其注重高质量的数据结构和算法，研究如何采用相应的数据结构来描述问题，以及采用什么样的算法去高效地解决问题。自 20 世纪 80 年代起，大多数流行的高级语言都是面向过程的高级语言，如 FORTRAN、Pascal 、C 等。由于面向过程的高级语言要求在编写程序之前严格设计好解决问题的每一个过程细节，高度强调过程化的程序分析思想，所以其适合设计规模较小、复杂度不太高的程序，但对于设计规模较大的程序时，就显得力不从心了。

（2）面向对象的高级语言

面向对象的思维方式即把世界上的万事万物都看成一个个实际对象，每个对象都有自己的特点，并以自己的方式做事，不同对象之间会存在着某种形式的联系（通信），以此构成世界的运转。从计算机专业术语来看，对象的特点就是它们的自身属性，所做的事情就是这个对象能够实现的方法或操作。采用面向对象的分析方法，在一定程度上会提高程序的重用性，降低程序的复杂度，使得计算机程序设计能够适应较复杂的应用需求。常见的支持面向对象分析思想的程序设计语言主要有 C++ 、Java、C＃等，适合作为设计大型复杂程序所采用的高级语言。目前，很多高校计算机类专业把一些面向对象的高级语言作为 C 语言的后续课程开设，本书在此不作介绍。

1.1.3 程序设计过程

“程序设计”即人们通常说的“编写程序”，简称编程。程序是计算机的主宰，控制着计算

机该去做什么事。程序设计就是分析、解决问题的方法和步骤，并将其记录下来的过程。通常程序设计过程主要历经以下几个阶段。

1. 分析

分析也称为问题域分析或需求分析。就是在设计某个计算机程序之前一定要搞清楚这个程序完成的是什么功能，能为我们做什么事情。这个阶段貌似很简单，甚至很多人对此不屑一顾。但是，对问题分析错误的结果就像写作文跑题，即使文字再工整、行文再通顺也得不到高分，最后还得从头返工。

2. 设计

设计就是明确拟编写的程序应该如何实现相应的功能。设计的内容主要是设计算法，即规划出拟编写程序的解题方法和具体步骤。例如，要编写一个求解三角形面积的程序，设计阶段就是要明确选用什么数学方法或思路来求解。比如，是选用通常的“底×高/2”的方法，还是选用“海伦公式”方法求解等。求解的每一个步骤都应清晰无误地用语言文字或流程图等描述出来。当然，如果只是编写一个很简单的小程序，设计阶段是可以省略的。但是，对于结构复杂的程序，必须对程序解题方法的过程及实施步骤进行严格的规划，这样才能防止在设计阶段出现严重的逻辑性设计错误。

3. 编码

编码阶段才是真正的“编写程序”，即按照设计好的算法选用一种计算机程序设计语言编写程序，通过特定的软件工具输入到计算机中。像 C 语言就是一种很好的高级程序设计语言。

4. 运行结果、分析与测试

运行程序，并分析程序执行后得到的结果。注意，能得到运行结果并不代表当前程序就一定正确，还要对其结果进行认真分析，看其是否合理，与我们预期推测的结果是否一致。

例如，用 C 语言编写一个关于求解正数 x 倒数(1/x)的程序(x>0)，y 表示 x 的倒数，即 y=1/x。当 x=2.0，求出 y 的值是 0.5，没有问题。但是，当 x=2，求出 y 的值却是 0，与实际不符。这是因为在 C 语言的语法中，两个整数做除法操作，当分子与分母同为整数时，得到的结果却是两者的整数商。所以，只有当 x 取值为正小数时，y 的值才是真正的 x 的倒数。这也说明了一个问题，这个程序对于某些数据(x 取正小数)能得到正确结果，而对另外一些数据(x 取正整数)，却得不到正确结果。这也正说明程序还有漏洞，需要修改。因此，编码结束后，要对程序进行测试。所谓测试，就是设计多组不同类型的输入数据，检查程序对不同数据的运行情况，从中尽量发现程序是否存在错误(漏洞)，并修改程序，使之能适用于各种数据的输入情况。所以，我们把程序“y=1/x”修改成“y=1.0/x”，就解决了对正数 x 求倒数的问题。实际操作中，尤其是作为商业用途使用的计算机程序，投入市场前必须经过严格测试。

5. 编写程序文档

在现实生活中，由于许多计算机程序编写后是通过市场提供给别人使用的，所以必须向使用者提供程序说明书，也称为用户文档，其内容应包括程序名称、程序功能、运行环境、程序的装入和启动、需要输入的数据以及使用的注意事项等。

最后强调的是，程序文档也是计算机程序的一个重要组成部分。正如现在很多软件产品的配套光盘既包括程序，也包括程序的操作说明书、使用说明书等一样。

1.2 为什么要学习C语言

C语言是一种通用的编程语言，具有功能丰富、使用灵活、运行速度快、运用范围广泛等优点，是当今最流行的计算机高级程序设计语言之一。C语言程序设计同样是国内高校计算机类专业的一门重要专业基础课程，也是这些专业学生入校后所学习的第一门程序设计类课程。掌握C语言已成为衡量软件开发人员程序设计能力的一项基本功。所以，学习C语言的重要性不言而喻。

1.2.1 C语言发展历程简介

C语言最早是由美国贝尔(Bell)实验室的Dennis M. Ritchie等人于20世纪70年代末发明设计出的一种高级计算机程序设计语言。1989年，美国国家标准协会(American National Standards Institute，ANSI)制定出一套完整的C语言语法标准，称为ANSI C。1990年，国际标准化组织(International Standard Organization，ISO)又在ANSI C标准的基础上进行少量的修改，发布了C语言ISO C的标准，即ISO 9899-1990，也称为C90。20世纪90年代中后期，ISO组织又对C90进行了某些技术上的完善，在C90之上推出了C99标准。目前，市面上大多数C语言书籍介绍C语言语法结构都遵循C90或C99标准。本书叙述C语言就是以C99为标准的。

1.2.2 C语言语法结构的特点

C语言是一种面向过程的高级程序设计语言，易学、易读、易懂、易编程、易维护，还具有直接访问计算机硬件等功能。C语言语法结构主要具有以下特点。

1. 语言简洁、紧凑

C语言自身一共只有32个关键字、9种控制语句，省略了一些不必要的成分，使用方便、灵活。

2. 运算符丰富

C语言的运算符范围很广泛。把赋值运算、关系比较、强制类型转换、括号等都作为运算符处理，从而使C的运算类型极其丰富，表达式类型多样化。在C语言中灵活使用各种运算符，可以实现在其他高级语言中难以实现的运算。

3. 数据结构丰富

C语言的数据结构很丰富，数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等。支持各种复杂的数据结构(如链表、树、栈等)的运算。

4. 便于实现程序的模块化

C语言具有结构化的控制语句(如if…else语句、switch语句、for语句等)。可以使用函数作为程序的模块单位，便于实现程序的模块化，符合现代软件企业编程风格的要求。

5. 程序设计的自由度大

C语言对语法不作太严格限制，程序设计的自由度大。例如，允许把一个整数赋给一个实型变量，可以把多条C语句写在同一行上，整型数据与字符型数据可以兼容等。因此，C语言允许程序编写者有较大的设计自由度。

1.3 简单的C程序

人们把事先用C语言编写好的程序称为C语言的源程序，简称C程序。为了更好地学习C语言，先通过几个简单的例子了解一下C程序的结构特点。这几个C程序的难度由浅入深，在功能上只是输出现成的文字信息，或是用来解决一些简单的数学问题。虽然有关C语言的语法内容还未介绍，但是可以从这几个例子中大致了解一下C程序的组成结构及其书写格式。由于C程序执行过程中所产生的C目标(中间)程序与最终的C可执行程序不是本书深入讨论的内容，所以本书把C语言的源程序统称为C程序，以方便初学者理解。

例1.4 在计算机屏幕上输出“欢迎学习C语言!”的文字信息。

C程序代码如下：

```
#include <stdio.h>                  //编译预处理指令;
main()                              //定义C程序的主函数;
{                                   //主函数起始的标识;
    printf("欢迎学习C语言!");      //输出"欢迎学习C语言!"的文字信息;
}                                   //主函数结束的标识;
```

程序运行结果如下：

```
欢迎学习C语言!Press any key to continue
```

程序分析：

本程序是在Visual C++ 6.0环境下运行的，有关在Visual C++ 6.0环境下运行C程序的方法将在本章1.5节中介绍。其中，“欢迎学习C语言!”的文字部分是程序运行后得到的输出结果，而Press any key to continue则是系统运行完一个程序后自动给出的提示信息，提示用户可以按下计算机键盘上任意键后重新返回程序窗口，等待下一步操作(例如，保存程序、修改程序等)。

例1.5 从键盘上任意输入一个小数a，表示某正方形的边长，计算出正方形的周长c与面积s的值。

C程序代码如下：

```
#include <stdio.h>          //编译预处理指令;
main()                      //定义C程序的主函数;
{
    float a,c,s;            //定义a,c,s三个表示小数数值的实型变量;
    scanf("%f",&a);         //通过计算机键盘任意输入变量a的值,作为正方形的边长值;
    c=4*a;                  //计算4*a的值,作为正方形的周长值,把结果存放在变量c中;
    s=a*a;                  //计算a*a的值,作为正方形的面积值,把结果存放在变量s中;
    printf("c=%f,s=%f\n",c,s);      //在屏幕上输出c、s的值;
}
```

假设输入a的值为1.5，即：

```
1.5↙
```