21世纪高等学校规划教材 | 计算机科学与技术

C++语言程序设计

(MOOC版)(第2版)

阚道宏 编著

清华大学出版社

C++语言程序设计

（MOOC版）（第2版）

阚道宏　编著

清华大学出版社
北京

内 容 简 介

本书是在《C++语言程序设计(MOOC 版)》的基础上进一步总结爱课程网“中国大学 MOOC”(http://www.icourse163.org/)的网络教学实践修订而成。本书按照实际编程应用来梳理和组织C++语言的知识点，按章节顺序可分为程序设计基础、结构化程序设计方法和面向对象程序设计方法三大部分。内容编排由易到难，循序渐进。每个小节都设计了适合在线评判的单选练习题，每章则设计了适合课堂讨论的程序阅读题、改错题和编程题。

凡开设“C++语言程序设计”课程的教师可将本书作为授课教材使用，联系作者可免费获得配套教学课件和视频。参加慕课(MOOC)或其他网络课程学习的学生可将本书作为线下阅读教材使用。

图书在版编目(CIP)数据

C++语言程序设计：MOOC 版/阚道宏编著. —2 版. —北京：清华大学出版社，2017
（21 世纪高等学校规划教材 • 计算机科学与技术）
ISBN 978-7-302-47562-0

Ⅰ. ①C…　Ⅱ. ①阚…　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（2017）第 140800 号

责任编辑：郑寅堃
封面设计：傅瑞学
责任校对：梁　毅
责任印制：李红英

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 装 者：清华大学印刷厂
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：29.75　　**字　　数**：721 千字
版　　次：2016 年 2 月第 1 版　2017 年 10 月第 2 版　　**印　　次**：2017 年 10 月第1 次印刷
印　　数：1～2000
定　　价：59.00 元

产品编号：074877-01

出版说明

随着我国改革开放的进一步深化，高等教育也得到了快速发展，各地高校紧密结合地方经济建设发展需要，科学运用市场调节机制，加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度，通过教育改革合理调整和配置了教育资源，优化了传统学科专业，积极为地方经济建设输送人才，为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是，高等教育质量还需要进一步提高以适应经济社会发展的需要，不少高校的专业设置和结构不尽合理，教师队伍整体素质亟待提高，人才培养模式、教学内容和方法需要进一步转变，学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007 年 1 月，教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》，计划实施“高等学校本科教学质量与教学改革工程（简称‘质量工程’）”，通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容，进一步深化高等学校教学改革，提高人才培养的能力和水平，更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中，各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势，对其特色专业及特色课程（群）加以规划、整理和总结，更新教学内容、改革课程体系，建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上，经教育部相关教学指导委员会专家的指导和建议，清华大学出版社在多个领域精选各高校的特色课程，分别规划出版系列教材，以配合“质量工程”的实施，满足各高校教学质量和教学改革的需要。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作，提高教学质量的若干意见》精神，紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”，在有关专家、教授的倡议和有关部门的大力支持下，我们组织并成立了“清华大学出版社教材编审委员会”（以下简称“编委会”），旨在配合教育部制定精品课程教材的出版规划，讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师，其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求，“编委会”一致认为，精品课程的建设工作从开始就要坚持高标准、严要求，处于一个比较高的起点上；精品课程教材应该能够反映各高校教学改革与课程建设的需要，要有特色风格、有创新性（新体系、新内容、新手段、新思路，教材的内容体系有较高的科学创新、技术创新和理念创新的含量）、先进性（对原有的学科体系有实质性的改革和发展，顺应并符合 21 世纪教学发展的规律，代表并引领课程发展的趋势和方向）、示范性（教材所体现的课程体系具有较广泛的辐射性和示范性）和一定的前瞻性。教材由个人申报或各校推荐（通过所在高校的“编委会”成员推荐），经“编委会”认真评审，最

后由清华大学出版社审定出版。目前，针对计算机类和电子信息类相关专业成立了两个“编委会”，即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。推出的特色精品教材包括：

（1）21 世纪高等学校规划教材·计算机应用——高等学校各类专业，特别是非计算机专业的计算机应用类教材。

（2）21 世纪高等学校规划教材·计算机科学与技术——高等学校计算机相关专业的教材。

（3）21 世纪高等学校规划教材·电子信息——高等学校电子信息相关专业的教材。

（4）21 世纪高等学校规划教材·软件工程——高等学校软件工程相关专业的教材。

（5）21 世纪高等学校规划教材·信息管理与信息系统。

（6）21 世纪高等学校规划教材·财经管理与应用。

（7）21 世纪高等学校规划教材·电子商务。

（8）21 世纪高等学校规划教材·物联网。

清华大学出版社经过三十多年的努力，在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌，为我国的高等教育事业做出了重要贡献。清华版教材形成了技术准确、内容严谨的独特风格，这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会

联系人：魏江江

E-mail:weijj@tup.tsinghua.edu.cn

第2版前言

本书是在《C++语言程序设计(MOOC 版)》的基础上进一步总结爱课程网“中国大学MOOC”(http://www.icourse163.org/)的网络教学实践修订而成。依据本教材开设的“C++语言程序设计”MOOC(慕课)课程已开设四个学期，累计选课人数超过十万人。

MOOC 学员能够积极参与课堂讨论，提出各种问题，并对如何开展 C++语言程序设计教学提出了很多宝贵的意见和建议。这里分享几个 MOOC 课堂的精华贴。

【空空 mooc369】：这门课是学软件的，为什么第 1 章很多内容是关于硬件的呀？

【教师回复】：初学者学习程序设计，首先应树立以下两个观念。

（1） 程序员向计算机硬件下达指令，然后由硬件执行指令。程序员应首先了解硬件的基本结构和工作原理，这样才能知道如何向硬件下达指令。

（2） 程序是一组下达给计算机硬件的指令序列（或称为语句序列）。仅从语法角度去理解，语句是抽象的。初学者要学会从有形的硬件去理解抽象的语句和语法。

程序设计课程只在一开始简单介绍一下硬件，后续章节在讲解 C++语法时会提及内存或 CPU 等硬件。

【luckymooc360】：C++语言为什么提出引用和指针的概念？

【教师回复】：程序员通过定义变量来申请内存，再用变量名访问所分配的内存单元。大型程序需要多个程序员协作开发，共同完成。如果其他程序员想访问上述变量的内存单元，例如读取其中的数据，可以吗？答案是肯定的，可以，但只能通过引用或指针来访问（即间接访问）。

【小二上盘 ID】：老师，我们学校先学 C 语言，然后学的是 C#和 Java。怎样才能学好程序设计？以后从事工作用 Java 好，还是 C/C++好？

【教师回复】：如何学习程序设计，这是很多初学者经常提出的问题。经过一段时间的学习，很多同学会产生新的困惑。例如，理论知识看了好几遍了，但是怎样提高编程能力？自己对 MOOC 课程的学习效果很满意，下一步该做什么？程序设计的学习过程是什么样的，最终能学到什么程度？针对这些问题我谈一下个人体会，供同学们参考。

学习程序设计大致可以分为三个阶段：初级、中级和高级。

（1）初级。初级阶段的目标是学习程序设计原理，其中包括计算机硬件基本结构及其工作原理，程序如何管理内存来存储数据（例如变量的定义与访问，数据类型，引用与指针等）、程序如何控制 CPU 来处理数据（例如各种不同的运算符，或通过控制语句来控制指令的执行顺序）等。程序设计原理还要学习如何设计大型、复杂的程序，这就需要学习程序设计方法。程序设计方法有两种，分别是结构化程序设计和面向对象程序设计。初级阶段学习结束后，同学们可以参加计算机等级考试（二级）或各种程序设计大赛等活动，并在应试过程中进一步提高自己的水平。

（2）中级。中级阶段的目标是学习程序应用开发（学会就能有好工作了哦）。程序应用开发需要基于别人的程序来开发，从零开始是不可能的。结构化程序设计方法规定：其他人给你函数库，你要会调用别人的函数。面向对象程序设计方法规定：其他人给你类库，你要知道如何使用别人的类库。因此我们在学习应用开发之前必须要掌握结构化程序设计方法或面向对象程序设计方法。目前面向对象程序设计是主流，已经很少有人继续给程序员提供函数库了，所提供的都是类库。掌握了面向对象程序设计方法，你只要拿到微软公司提供的 MFC 类库（随 Visual C 6.0 或 Visual Studio 提供），就可以用 C++语言开发 Windows 图形界面的程序（或者你拿到苹果公司的类库、谷歌公司的类库，就可以使用这些类库开发 iPhone 或 Android 系统的 App 了）。不同的操作系统是由不同厂家开发的，它们对计算机语言的支持程度有所不同。例如，Windows 操作系统是由微软开发的，开发 Windows 软件主要使用 C++和 C#语言；MacOS/iOS 操作系统是由苹果公司开发的，开发 MacOS/iOS 软件主要使用 Objective-C（C++的变种）；Android 操作系统（从 Linux 演变而来）由谷歌公司主导，开发 Android 软件主要使用 Java 语言。可以看出，程序应用开发可能会用到不同的计算机语言，但程序设计原理是共同的。

（3）高级。高级阶段的目标是提高自己的理论水平，不光要知其然，还要知其所以然。计算机专业的同学需要先学习程序设计原理，然后再学习计算机组成原理、数据结构、操作系统、编译原理、数据库原理、计算机网络、计算机图形学、数字图像处理、算法设计、离散数学和人工智能等课程。在学习完这些专业课程之后，你的理论水平会得到很大提高，将会成为一名真正的高手。

【LingDash】：老师，总地来说讲得挺好的，但是我有些想法。比如在讲类的继承与派生时，应先让学生体会到，每个类都从底层开发十分麻烦，那么怎么解决呢？这时引出“类的继承与派生”，将用与不用“类的继承与派生”进行比较，这样的学习会更加符合人的认知规律。

【教师回复】：有道理，接受你的建议，谢谢！

【cc76965】很高兴能够遇到这门公开课。这门公开课的条理十分清晰，让我对 C++的语法有了一个清晰的脉络。我是一名计算机专业的学生，已经学了数据结构，但是缺少开发项目的经验。不知道如何去寻找开发项目，也不知道如何下手，希望老师能够给我一些建议。

【网友 Crazy 峰少回复】：下面这个网站里有大量的编程题目可以做：https://www.patest.cn/。

这一版就是在总结广大 MOOC 学员所提出的难点、问题和建议的基础上对上一版所做的修订。重点完善了结构化程序设计和面向对象程序设计这两种程序设计方法的内容，使之更加系统化。

在此谨向中国大学 MOOC 和所有提出宝贵建议的 MOOC 学员们表示感谢！

作 者

2017 年 3 月于北京

前言

1. 关于 MOOC

MOOC（Massive Open Online Course），即大规模开放在线课程，中文译为“慕课”，是近几年兴起的一种基于互联网的新型教学模式。2012 年被称为“MOOC 元年”。MOOC 教学模式实现了两个转变，即由以教师为中心向以学习者为中心转变，学习者则由被动学习向主动学习转变。与普通网络教学视频所不同的是，MOOC 实现了从授课到习题、讨论、答疑、测验，直到最终学习评价等环节的完整教学过程。与传统课堂授课不同的是，开设 MOOC 课程需重新梳理和组织知识点，并分别提供适合线上使用的练习题以及线下使用的讨论或实验题。

可以将线上 MOOC 与线下课堂这两种教学模式结合起来。线上 MOOC 就是先由学生自主完成知识学习，例如观看视频、做线上习题等。线下课堂则是由教师组织课堂讨论、实验、测验，或讲解重点疑难问题。“线上 MOOC，线下课堂”是对现有“课上听课，课下作业”教学模式的翻转。虽然 MOOC 教学模式尚处于起步试验阶段，但大多数网络学习者十分喜欢 MOOC。目前已涌现出很多知名的 MOOC 网站，例如国外的 Coursera、Udacity 和 Edx，国内的中国大学 MOOC、学堂在线和雨虹学网等。中国农业大学也正在基于雨虹学网积极开展校内课堂教学改革方面的尝试与探索。

学习 C++语言程序设计需要边阅读、边思考、边消化吸收。虽然有了大量的网上资源，但纸质教材仍是初学者线下学习的首选方式，这也是作者编写出版本书的目的。

2. 本书特色

1）适用于 MOOC 在线教育课程

本书按应用需求来梳理和组织 C++语言的知识点，其中包括结构化程序设计方法和面向对象程序设计方法。内容编排由易到难，循序渐进。每个小节都设计了适合在线评判的单选练习题，每章则设计了适合课堂讨论的程序阅读题、改错题和编程题。

2）采用案例教学

每个知识点都从精心设计的任务需求开始导入，然后提出对应的实现方法，最后系统地阐述其语法细则，既保证了知识体系的完整性，又能让读者直观理解抽象的概念和原理。

3）创新教学方法

本书从三个方面对 C++语言教学进行了探索。一是强化初学者对“程序由计算机硬件执行”这一基本概念的认知，从有形的硬件来理解相对抽象的程序，这样各种语法概念就不再那么抽象了；二是明确提出程序设计过程中存在不同的程序员角色，并充分利用角色

来引导读者理解语法的应用语境；三是明确提出必须从代码分类管理、数据类型、归纳抽象和代码重用等多个维度才能准确理解面向对象程序设计方法。教学实践表明，上述教学方法可降低学习难度。

3. 内容摘要

本书内容按章节顺序可分为三部分，分别是程序设计基础（第 1~4 章）、结构化程序设计方法（第 5~6 章）和面向对象程序设计方法（第 7~10 章）。

第 1 章　程序设计导论。从初学者对计算机已有的认知开始，将初学者逐步引导到计算机程序的世界。本章首先介绍计算机硬件、指令及机器语言、程序等基本概念，然后描述程序与计算机硬件、程序员、用户之间的关系，让读者在一开始就能明确程序员的职责，实现从用户到程序员的角色转换。本章要点：一是让读者从有形的硬件来理解相对抽象的软件；二是让读者认识到计算机中的数据是有类型的，类型决定了数据在计算机中的存储位数和存储格式；三是让读者知道，学习程序设计和学习编程语言不是一回事。和 C 语言、Java 语言相比，C++语言的知识体系更加系统、全面。本书选用 C++语言作为程序设计初学者的入门语言。

第 2 章　数值计算。本章从最简单的数值计算问题开始，以案例教学的方式让读者领会程序设计中一些最基础的概念，其中包括程序中的变量和常量、表达式与运算符、数据的输入和输出等。最后介绍了 C++程序访问内存的三种方式，它们分别是变量名、引用和指针。本章要点：一是让读者将程序中的数据与内存联系起来，这样就很容易理解数据类型、引用和指针等初学者难以掌握的概念；二是让读者重点关注运算符的运算规则、优先级和结合性等语法细节；三是让读者初步体会到计算机语言与人类语言的不同之处，即计算机语言的语法规则非常严格，甚至到了机械的程度，稍有不慎就会出现语法错误。

第 3 章　算法与控制结构。本章讲解程序中的算法及三种算法基本结构，并通过选择结构和循环结构中的条件引出布尔类型。C++语言通过选择语句来描述选择结构算法，通过循环语句来描述循环结构算法。最后通过案例简单讲解算法的设计与评价方法。本章要点：一是让读者了解绝大部分复杂算法都可以由三种基本的算法结构来完成；二是让读者掌握布尔类型的作用及其相关的运算符；三是让读者了解编程能力实际上是一个人计算思维能力的反映，阅读程序和模仿编程是初学者培养计算思维能力的两个重要途径；四是让读者根据案例认真体会如何根据算法合理选用不同的控制语句。

第 4 章　数组与文字处理。本章学习如何在程序中存储和处理大量数据。数组可以存储大量具有相同类型的数据集合。计算机只能存储和处理数值数据，而文字处理程序所处理的对象是字符数据，为此，C++语言引入了字符类型。读者需深入了解字符编码和字符类型。文字处理必须使用数组，即字符型数组。本章最后用一节的篇幅简单介绍了中文处理及 Unicode 编码。本章要点：一是让读者重点掌握数组定义及访问的语法规则；二是让读者认识到计算机内部对数组的管理和访问是通过指针（即内存地址）来实现的；三是让读者通过具体案例初步了解数组的常用处理算法。

经过前 4 章的学习，读者已掌握了程序设计原理基础部分的内容。那么该如何编写更大型的计算机程序呢？这就需要进一步学习程序设计原理的高级部分，即程序设计方法。

程序设计方法的基本思想是：将大型程序中的数据和算法分解成程序零件，将不同零件的设计任务交由不同的程序员完成，这样就能以团队的形式来共同开发。更进一步，如果所分解出的程序零件在以前项目中曾经开发过，或者可以从市场上购买到，那么就可以直接使用这些零件来组装软件，实现快速开发。使用已有的程序零件，实际上是重用其程序代码，这就是程序设计中的代码重用。从第 5 章开始，本书对程序设计方法进行系统介绍。

第 5 章　结构化程序设计之一。本章学习如何将一个复杂的数据处理算法分解成多个简单模块，分而治之，这被称为是结构化程序设计方法。C++语言支持结构化程序设计方法，以函数的语法形式来描述和组装模块，即函数的定义和调用。函数是结构化程序设计方法的基础，它为代码重用提供了有效的手段。函数之间需要共享数据才能完成规定的数据处理任务。C++语言提供了集中管理和分散管理两种不同的数据管理策略。本章要点：一是读者要准确领会结构化程序设计的思想内涵，并熟练掌握 C++语言中函数相关的语法知识；二是让读者深入计算机内部，了解程序执行时其代码和变量在内存中的存储原理，这样可以更容易理解变量作用域和生存期等抽象的概念；三是读者要准确把握函数间传递数据的三种方式；四是读者要分别站在两种不同的程序员角度，即定义函数的程序员和调用函数的程序员，才能更容易地理解函数相关的各种语法知识。

第 6 章　结构化程序设计之二。本章学习如何以多文件结构的形式来组织和管理源代码，并介绍几种常用的编译预处理指令；然后再介绍几种特殊形式的函数，其中包括带默认形参值的函数、重载函数、内联函数、带形参和返回值的主函数以及递归函数等。本章还会介绍与 C 语言相关的系统函数和自定义数据类型。本章最后以微软公司开发的 Win32 API 函数库为例介绍如何开发一个 Windows 图形用户界面程序，并对结构化程序设计方法进行简单的回顾和总结。本章要点：一是学习掌握与多文件结构相关的语法知识，其中包括外部函数和全局变量的声明、头文件等；二是重点掌握带默认形参值的函数、重载函数和内联函数这三种常用的特殊函数形式；三是牢固树立重用代码的思想，学会通过调用他人编写的函数来提高开发效率。

第 7 章　面向对象程序设计之一。面向对象程序设计方法将程序中的数据元素和算法元素根据其内在关联关系进行分类管理，这就形成了“类”的概念。分类可以更好地管理。类相当于是一种自定义的数据类型，用类所定义的变量称为“对象”。本章通过具体案例演示了结构化程序设计是如何演变到面向对象程序设计的，然后再系统地介绍面向对象程序设计方法。本章内容包括类的定义、对象的定义与访问、对象的构造与析构、类中的常成员与静态成员以及类的友元等。本章要点：一是读者必须从代码分类管理、数据类型、归纳抽象和代码重用等多个维度才能准确理解类与对象的概念；二是读者需认真学习类与对象编程的具体语法规则；三是深入领会面向对象程序设计通过设置访问权限来实现类封装的基本原理；四是深入了解对象的构造与析构过程，程序员通过编写构造与析构函数来参与对象的构造与析构过程；五是读者要懂得从两个不同的角度，即定义类的程序员和使用类定义对象的程序员，才能更容易地理解类与对象相关的各种语法知识。

第 8 章　面向对象程序设计之二。重用类代码有三种方式，分别是用类定义对象、类的组合和类的继承。本章讲解类的组合与继承。程序员可以基于已有的零件类来定义新的整体类，这就是类的组合。程序员可以继承已有的基类来定义新的派生类，这就是类的继

承与派生。利用派生类和基类之间的特殊关系可以进一步提高程序代码的可重用性，这就是面向对象程序设计中的对象替换与多态技术。本章将具体讲解与多态相关的运算符重载、虚函数和抽象类等概念。最后本章将简单讨论一下类的多继承。本章要点：一是让读者学会使用组合和继承的方法来定义新类，这样可以提高类代码的开发效率；二是读者应理解，类在组合或继承时可以进行二次封装；三是从提高程序代码重用性的角度可以更容易地理解对象多态性；四是多继承会导致语法陷阱，新的面向对象程序设计语言（例如 Java 和 C#）已不再支持类的多继承，而只支持接口的多继承，读者只需要了解多继承的基本原理即可。

第 9 章　流类库与文件 I/O。C 语言通过输入/输出函数（例如 scanf 和 printf）实现了数据的输入和输出。C++语言则是通过输入/输出流类为程序员提供了输入/输出的功能。这些输入/输出流类都是从类 ios 派生出来的，组成了一个以 ios 为基类的类族，这个类族被称为 C++语言的流类库。本章将介绍流类库中三组不同功能的输入/输出流类，它们分别是通用输入/输出流类、文件输入/输出流类和字符串输入/输出流类。本章要点：一是读者应理解之前所用的 cin、cout 指令实际上分别是通用输入/输出流类的对象；二是通过本章学习，读者可以从侧面了解全球顶尖的 C++程序员是如何来设计和编写类的，这样可以帮助读者进一步深入体会前面所学习的各种面向对象程序设计知识；三是重点学习如何进行文件读写操作，大部分程序都需要使用文件来保存数据。

第 10 章　C++标准库。C++语言全盘继承了 C 语言的标准 C 库，另外又增加了一些新的库。新库中包含一些新增的系统函数，但更多的是为面向对象程序设计方法所提供的系统类库，这些新库被统称为 C++标准库。为了更好地凝练源代码，C++语言引入了模板技术，其中包括函数模板和类模板。模板技术是一种代码重用技术，C++标准库在编写时就采用了模板技术，因此标准库能以较少的代码量来提供很强大的功能。本章内容重点介绍模板技术、C++语言的异常处理机制以及 C++标准库所提供的数据集合存储及处理功能。本章最后以微软公司开发的 MFC 类库为例介绍如何开发一个 Windows 图形用户界面程序。本章要点：一是让读者了解如何使用模板技术来提高函数和类代码的可重用性；二是重点学习 C++语言的异常处理机制；三是初步掌握如何使用 C++标准库中的向量类、列表类、集合类和映射类来存储和处理数据集合。

学习完 C++面向对象程序设计之后，程序员在拿到一个具体的程序设计任务时，首先应当考虑有哪些现成的类库可以使用。使用现成的类库开发程序，开发周期将大大缩短。基于已有的类库开发程序，相当于是用别人已经做好的零件来组装产品。程序的应用开发，通常就是用已有的程序零件来组装自己的软件产品。只要掌握了面向对象程序设计方法和 C++语言，相信每位读者都能够借助各种第三方类库，发挥出无限的开发潜能。

4. 使用建议

凡希望开设 C++语言程序设计在线教育课程的教师，可将本书作为授课教材。联系作者可免费获得配套教学课件和视频。参加在线课程学习的学生可将本书作为线下阅读教材。

如将本书作为课堂教学用书，则建议讲课学时和实验学时各为 32 学时，合计 64 学时。每学时 50 分钟。作者本人按如下方式安排讲课学时：第 1、3、4、9、10 章各 2 学时，第

2、5、6、7 章各 4 学时，第 8 章 6 学时。

联系作者：kandaohong@cau.edu.cn

5. 致谢

作者编写本书的想法源于中国农业大学“雨虹学网：面向主动学习的教学云平台建设”项目。在参与相关系统开发和教学实践的过程中，作者积累了一些 MOOC 在线课程教学的经验。

本书编写过程中，得到了中国农业大学信电学院高万林院长的热情鼓励和大力支持。本书部分素材来自于雨虹学网的教学实践活动，这得益于张晓东教授、孙瑞志教授等领导的关心和指导。另外，本书在编写过程中还得到了郑立华、吕春利、冀荣华、刘云玲、陈瑛、周绪宏、胡慧、段晶洁、李鑫、李静等同事和同学的热心帮助。在此一并致以衷心的感谢！

最后，感谢家人对我的理解和支持。

作 者

2015 年 9 月于北京

目 录