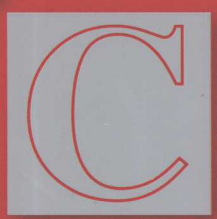
21世纪高等学校计算机基础实用规划教材

计算机程序设计基础教程
——C++语言

刘卫国 周欣然 编著



清华大学出版社

21世纪高等学校计算机基础实用规划教材

计算机程序设计基础教程
——C++语言

刘卫国 周欣然 编著

清华大学出版社
北京

内 容 简 介

本书遵循以计算思维能力培养为切入点的教学改革思路，以C++语言作为实现工具，介绍程序设计的基础知识与基本方法。全书的主要内容有程序设计概述、基本数据及运算、流程控制、函数、批量数据的组织、复杂数据及运算、类与对象、类的继承与派生、多态性与虚函数、模板与STL简介、输入输出流、异常处理。

在本书编写过程中，考虑到初学者的认知特点以及培养程序设计能力的教学要求，对C++语言本身的语法规则做了适当处理和组织编排，突出C++语言的重要概念和本质特点。全书以实际问题的求解过程为向导，突出从问题到算法、再到程序的一种思维过程，强调计算机求解问题的思路引导与程序设计思维方式的训练，既介绍C++语言在面向过程程序设计中的应用，又介绍C++语言在面向对象程序设计中的应用，重点放在程序设计的思想与方法上。

本书可作为高等学校计算机程序设计课程的教材，也可供参加各类计算机等级考试的读者以及社会各类计算机应用人员阅读参考。

图书在版编目(CIP)数据

计算机程序设计基础教程：C++语言/刘卫国等编著. —北京：清华大学出版社，2015
21世纪高等学校计算机基础实用规划教材
ISBN 978-7-302-40051-6

Ⅰ. ①计… Ⅱ. ①刘… Ⅲ. ①C语言—程序设计—教材 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2015)第091150号

责任编辑：魏江江　赵晓宁
封面设计：何凤霞
责任校对：梁　毅
责任印制：沈　露

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：北京鑫海金澳胶印有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：24.25　　**字　　数**：604千字
版　　次：2015年8月第1版　　**印　　次**：2015年8月第1次印刷
印　　数：1～2000
定　　价：44.50元

产品编号：029498-01

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。

本系列教材立足于计算机公共课程领域，以公共基础课为主、专业基础课为辅，横向满足高校多层次教学的需要。在规划过程中体现了如下一些基本原则和特点。

（1）面向多层次、多学科专业，强调计算机在各专业中的应用。教材内容坚持基本理论适度，反映各层次对基本理论和原理的需求，同时加强实践和应用环节。

（2）反映教学需要，促进教学发展。教材要适应多样化的教学需要，正确把握教学内容和课程体系的改革方向，在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养，为学生的知识、能力、素质协调发展创造条件。

（3）实施精品战略，突出重点，保证质量。规划教材把重点放在公共基础课和专业基础课的教材建设上；特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版，逐步形成精品教材；提倡并鼓励编写体现教学质量和教学改革成果的教材。

（4）主张一纲多本，合理配套。基础课和专业基础课教材配套，同一门课程可以有针对不同层次、面向不同专业的多本具有各自内容特点的教材。处理好教材统一性与多样化，基本教材与辅助教材、教学参考书，文字教材与软件教材的关系，实现教材系列资源配套。

(5) 依靠专家，择优选用。在制定教材规划时依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时，要引入竞争机制，通过申报、评审确定主题。书稿完成后要认真实行审稿程序，确保出书质量。

繁荣教材出版事业，提高教材质量的关键是教师。建立一支高水平教材编写梯队才能保证教材的编写质量和建设力度，希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21世纪高等学校计算机基础实用规划教材

联系人：魏江江 weijj@tup.tsinghua.edu.cn

前言

计算机程序设计基础是一门非常重要的计算机课程，这门课通过介绍程序设计的基础知识，使学生掌握高级语言程序设计的基本思想、方法和技术，理解利用计算机解决实际问题的基本过程和思维规律，从而具备创新能力，为未来应用计算机进行科学研究与实际应用奠定坚实的基础。

近年来，计算机教育界提出，应将计算思维能力培养作为计算机教育的重要任务。计算思维(computational thinking)是指运用计算机科学的基础概念进行问题求解、系统设计以及人类行为理解的一系列思维活动。计算思维不仅反映了计算的原理，更重要的是体现了基于计算机的问题求解思路与方法。就课程性质而言，计算机程序设计基础课程最能够体现问题求解方法，是理解计算机工作过程的有效途径，也是计算思维能力培养的重要载体。因此，计算机程序设计基础课程的重要性不仅体现在一般意义上的程序设计能力的培养，而且体现在引导学生实现问题求解思维方式的转换，即学生计算思维能力的培养。当然，要实现计算思维能力的培养不是一件容易的事，这也是程序设计教学改革的重要切入点。本教材正是按照这种改革理念，以实际问题的求解过程为向导，介绍程序设计的基础知识与基本方法，教材内容强调计算机求解问题的思路引导与程序设计思维方式的训练，重点放在程序设计的思想与方法上。

C++语言由C语言发展而来，它保留了C语言原有的优点，与C兼容，用C语言写的程序基本上可以不加修改地用于C++语言。同时，C++语言又在C的基础上得到发展，一是基于面向过程机制对C语言的功能做了不少扩充；二是增加了面向对象机制，支持面向对象程序设计方法。在当今软件开发中，C++语言有着广泛的应用，也是高等学校最常用的程序设计教学语言之一。本书针对程序设计初学者以及准备参加C++语言二级考试的读者，本着让初学者学习C++语言更容易的目的，合理组织内容，突出C++语言的本质特点与教学要求，既介绍C++语言在面向过程程序设计中的应用，又介绍C++语言在面向对象程序设计中的应用，强调程序设计基本思想和思维训练，力求体现以下4个方面的特点。

(1) 全书强调计算机问题求解的思路引导，突出从问题到算法、再到程序的一种思维过程。不是罗列现成的程序，而是讲清楚程序是怎么来的，怎样才能得到程序。在语言编译系统的选择上，本书使用Visual Studio 2010作为上机环境，目的是让教材内容更接近软件开发的实际需要，为读者进一步学习和应用C++语言打下基础。

(2) 恰当取舍，突出C++语言的本质特点和教学要求。全书用通俗易懂的叙述讲清C++语言的重要概念，不求面面俱到，对于初学者不常用到的内容作了简化处理。教材也不过分死抠语言细节，引导读者在实践中去掌握语法规则。

(3) 全书的组织编排遵循循序渐进原则。第1章介绍程序设计的基础知识，建立起对

C++语言的初步认识。第2章介绍基本数据类型，在这一章中并未罗列C++语言的全部运算，而是将相关运算分散到各章去介绍，一方面让读者尽早接触到程序，另一方面也避免了教学过程中的单调乏味。第3章分别介绍程序的3种基本结构，体现了最基本的程序设计方法。第4章介绍函数，体现了模块化程序设计的需要。前4章只涉及C++语言的基本数据类型，重点放在程序的3种基本结构的实现方法和程序设计能力培养上。第5和第6章是数组和C++语言的构造数据类型，涉及更复杂数据的表达方法。第7～10章是面向对象程序设计的内容，先介绍类与对象的操作，再介绍类的基本特性，这是C++语言的特色内容。第11章是文件操作，这是程序设计语言的传统内容。第12章介绍异常处理，这是软件开发中经常使用的方法。全书内容编排符合初学者的认知特点，有利于总体上把握课程内容，帮助读者逐步深入理解和掌握课程知识。各章小结中总结了本章主要的知识点，帮助读者总结归纳课程内容，达到巩固提高的目的。

(4)本书有配套的教学参考书、教学课件与相关教学资源。为了方便教学和读者上机操作练习，笔者还编写了《计算机程序设计实践教程——C++语言》(清华大学出版社出版)一书，作为与本书配套的教学参考书。实践教程既与本教材相互配套，又是本教材很好的补充。另外，还有与本书配套的教学课件、各章习题答案、例题源程序等教学资源，可从清华大学出版社网站(http://www.tup.com.cn)下载使用，也可发邮件到weijj@tup.tsinghua.edu.cn咨询。

本书第1～第7章由刘卫国编写，第8～第12章及附录由周欣然编写。参与讨论与部分编写工作的还有蔡旭晖、童键、刘胤宏、文碧望、石玉、欧鹏杰、胡勇刚、刘苏洲、孙士闯、周克涛等。清华大学出版社的编辑对本书的出版做了大量工作，在此表示衷心的感谢。

本书在排版时，与程序相关的变量用正体，这样程序中的变量和文字叙述中的变量保持形式一致，方便读者阅读。

由于编者水平有限，书中难免存在不足之处，恳请广大读者批评指正。

编者

2015年4月

目 录

第1章 程序设计概述

计算机是在程序(program)控制下进行自动工作的,它解决任何实际问题都依赖于解决问题的程序。要编写程序就要熟悉一种程序设计语言,掌握程序设计的基本方法,理解用计算机解决问题的基本过程和思维规律,为应用计算机进行科学研究与实际应用奠定坚实基础。C++语言由C语言发展而来。与C语言一样,C++语言具有程序简洁、数据类型丰富、表达能力强、使用灵活、实用高效等优点,而且支持面向对象程序设计方法,在当今软件开发中有着广泛的应用。本书以C++语言作为实现工具,介绍程序设计的基本思想和方法。

本章介绍程序设计的基本知识、C++语言的发展与特点、C++语言程序的基本结构以及C++语言程序的执行步骤。通过本章的学习,使读者对程序设计和C++语言有一个概要认识,从而为以后各章的学习打下基础。

1.1 程序设计基础知识

在学习C++语言程序设计之前,需要了解一些程序设计的基础知识,包括程序设计的过程、算法的概念、算法的描述方法以及流行的程序设计方法。

1.1.1 程序与程序设计

从一般意义来说,程序是对解决某个实际问题的方法和步骤的描述,而从计算机角度来说,程序是用某种计算机能理解并执行的语言所描述的解决问题的方法和步骤。计算机执行程序所描述的方法和步骤,并完成指定的功能。所以,程序就是供计算机执行后能完成特定功能的指令序列。

一个计算机程序主要描述两部分内容:一是描述问题的每个对象和对象之间的关系;二是描述对这些对象作处理的处理规则。其中关于对象及对象之间的关系是数据结构(data structure)的内容,而处理规则是求解的算法(algorithm)。针对问题所涉及的对象和要完成的处理,设计合理的数据结构可有效地简化算法,数据结构和算法是程序最主要的两个方面。著名的瑞士计算机科学家N. Wirth教授曾提出:

算法 + 数据结构 = 程序

程序设计的任务就是设计解决问题的方法和步骤(即设计算法),并将解决问题的方法和步骤用程序设计语言来描述。对于初学者来说,往往把程序设计简单地理解为只是编写一个程序,这是不全面的。程序设计反映了利用计算机解决问题的全过程,包含多方面的内容,而编写程序只是其中的一个方面。使用计算机解决实际问题,通常是先要对问题进行分

析并建立数学模型，然后考虑数据的组织方式和算法，并用某一种程序设计语言编写程序，最后调试程序，使之运行后能产生预期的结果。这个过程称为程序设计(programming)。具体要经过以下4个基本步骤。

1. 分析问题，确定数学模型或方法

要用计算机解决实际问题，首先要对待解决的问题进行详细分析，弄清问题的需求，包括需要输入什么数据，要得到什么结果，最后应输出什么，即弄清要计算机“做什么”。然后把实际问题简化，用数学语言来描述它，这称为建立数学模型。建立数学模型后，需选择计算方法，即选择用计算机求解该数学模型的近似方法。不同的数学模型，往往要进行一定的近似处理。对于非数值计算则要考虑数据结构等问题。

2. 设计算法，画出流程图

弄清楚要计算机“做什么”后，就要设计算法，明确要计算机“怎么做”。解决一个问题，可能有多种算法。这时，应该通过分析、比较，挑选一种最优的算法。算法设计后，要用流程图把算法形象地表示出来。

3. 选择编程工具，按算法编写程序

当为解决一个问题确定了算法后，还必须用程序设计语言将该算法编成程序，这个过程称为编码(coding)。

4. 调试程序，分析输出结果

编写完成的程序，还必须在计算机上运行，排除程序可能的错误，直到得到正确结果为止。这个过程称为程序调试(debugging)。即使是经过调试的程序，在使用一段时间后，仍然会被发现尚有错误或不足之处。这就需要对程序做进一步的修改，使之更加完善。

解决实际问题时，应对问题的性质与要求进行深入分析，从而确定求解问题的数学模型或方法，接下来进行算法设计，并画出流程图。有了算法流程图，再来编写程序就容易多了。

1.1.2 算法及其描述

计算机是通过执行人们所编制的程序来完成预定的任务。在广义上说，计算机按照程序所描述的算法对某种结构的数据进行加工处理。

算法是对数据运算的描述，而数据结构是指数据的组织存储方式，包括数据的逻辑结构和存储结构。程序设计的实质是对实际问题选择一种好的数据结构，并设计一个好的算法，而好的算法在很大程度上取决于描述实际问题的数据结构。

1. 算法的概念

在日常生活中，人们做任何一件事情，都是按照一定规则、一步一步地进行，这些解决问题的方法和步骤称为算法。例如，工厂生产一部机器，先把零件按一道道工序进行加工，然后，把各种零件按一定法则组装起来，生产机器的工艺流程就是算法。

计算机解决问题的方法和步骤，就是计算机解题的算法。计算机用于解决数值计算，如科学计算中的数值积分、解线性方程组等的计算方法，就是数值计算的算法；用于解决非数值计算，如用于数据处理的排序、查找等方法，就是非数值计算的算法。要编写解决问题的程序，首先应设计算法，任何一个程序都依赖于特定的算法，有了算法，再来编写程序是容易的事情。

下面举两个简单例子，以说明计算机解题的算法。

【例 1-1】 求 $u=\dfrac{x-y}{x+y}$，其中 $x=\begin{cases}a^2+b^2 & a<b\\ a^2-b^2 & a\geqslant b\end{cases}$，$y=\begin{cases}\dfrac{a+b}{a-b} & a<b\\ \dfrac{4}{a+b} & a\geqslant b\end{cases}$。

这一题的算法并不难，可写成：

(1) 从键盘输入 a，b 的值。

(2) 如果 $a<b$，则 $x=a^2+b^2$，$y=\dfrac{a+b}{a-b}$，否则 $x=a^2-b^2$，$y=\dfrac{4}{a+b}$。

(3) 计算 u 的值：$\dfrac{x-y}{x+y}$。

(4) 输出 u 的值。

【例 1-2】 输入 10 个数，要求找出其中最大的数。

设 max 单元用于存放最大数，先将输入的第 1 个数放在 max 中，再将输入的第 2 个数与 max 相比较，较大者放在 max 中，然后将第 3 个数与 max 相比，较大者放在 max 中……一直到比完 9 次为止。

算法要在计算机上实现，还需要把它描述为更适合程序设计的形式，对算法中的量要抽象化、符号化，对算法的实施过程要条理化。上述算法可写成如下形式：

(1) 输入一个数，存放在 max 中。

(2) 用 i 来统计比较的次数，其初值置 1。

(3) 若 $i\leqslant 9$，执行第(4)步，否则执行第(8)步。

(4) 输入一个数，放在 x 中。

(5) 比较 max 和 x 中的数，若 $x>$max，则将 x 的值送给 max；否则，max 值不变。

(6) i 增加 1。

(7) 返回到第(3)步。

(8) 输出 max 中的数，此时 max 中的数就是 10 个数中最大的数。

从上述算法示例可以看出，算法是解决问题的方法和步骤的精确描述。算法并不给出问题的精确解，只是说明怎样才能得到解。每一个算法都是由一系列基本的操作组成的。这些操作包括加、减、乘、除、判断、置数等。所以，研究算法的目的就是要研究怎样把问题的求解过程分解成一些基本的操作。

算法设计好后，要检查其正确性和完整性，再根据它用某种高级语言编写相应的程序。程序设计的关键就在于设计一个好的算法。所以，算法是程序设计的核心。

2. 算法的特性

从上面的例子中，可以概括出算法的 5 个特性。

(1) 有穷性。算法中执行的步骤总是有限次数的，不能无止境地执行下去。例如，计算圆周率 π 的值，可用如下公式：

$$\frac{\pi}{4}=1-\frac{1}{3}+\frac{1}{5}-\frac{1}{7}+\cdots$$

这个多项式的项数是无穷的，因此，它是一个计算方法，而不是算法。要计算 π 的值，只能取有限项。例如，计算结果精确到第 5 位，那么，这个计算就是有限次的，因而才能称得上算法。