普通高等教育“十一五”国家级规划教材

21世纪大学本科
计算机专业系列教材

杨 波 刘明军 主编
李晓明 主审

程序设计基础（C语言）

http://www.tup.com.cn

- 国家精品课程主讲教材
- 根据教育部“高等学校计算机科学与技术专业规范”组织编写
- 与美国 ACM 和 IEEE CS *Computing Curricula* 最新进展同步

清华大学出版社

普通高等教育“十一五”国家级规划教材

21世纪大学本科计算机专业系列教材

国家精品课程主讲教材

程序设计基础(C语言)

杨波 刘明军 主编

杨波 刘明军 潘玉奇 蔺永政 袁宁 编著

李晓明 主审

清华大学出版社

北京

内 容 简 介

本书以培养编程能力为出发点，以实用性为目标，全面地介绍了C语言程序设计的基本知识和程序设计的基本方法。全书分为9章，内容涵盖了C语言的全部知识点。首先介绍了程序设计的基本概念、C语言基础与程序结构，给学习者一个全面的程序概念；然后逐步介绍了C语言基本内容和程序设计方法；最后给出了程序设计实例。

本书是作者多年来从事C语言教学的经验积累，可作为高等学校大学本科、高职高专“C语言程序设计”课程的教材，也可作为IT领域C语言程序设计者的自学用书。

图书在版编目（CIP）数据

程序设计基础(C语言)/杨波，刘明军主编. —北京：清华大学出版社，2010.9
(21世纪大学本科计算机专业系列教材)
ISBN 978-7-302-23496-8

Ⅰ. ①程… Ⅱ. ①杨… ②刘… Ⅲ. ①C语言—程序设计 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2010)第155561号

责任编辑：张瑞庆　柴文强
责任校对：李建庄
责任印制：何　芊

出版发行：清华大学出版社　　**地　　址**：北京清华大学学研大厦A座
http://www.tup.com.cn　　**邮　　编**：100084
社　总　机：010-62770175　　**邮　购**：010-62786544
投稿与读者服务：010-62795954，jsjjc@tup.tsinghua.edu.cn
质　量　反　馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
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：185×260　　**印　张**：23.25　　**字　数**：576千字
版　　次：2010年9月第1版　　**印　次**：2010年9月第1次印刷
印　　数：1～4000
定　　价：32.80元

产品编号：037627-01

前言

FOREWORD

C语言是计算机程序设计语言的主流语种。30多年来，C语言经历了不断的发展和完善，逐步成为国内外公认的一种优秀程序设计语言，有着其他语言不可比拟的优点。

目前C语言教材主要分为两类。一类是以C语法为中心，在介绍语法的基础上，结合程序设计巩固C语言的语法知识。强调的是语法教学，C语言知识的掌握，而不是C语言程序设计能力。另一类是案例教材，通过案例学习，兼顾语法教学，通过模仿学习程序设计。相对前一类，语法教学系统性相对欠缺。本教材是在总结我们建设“C语言程序设计”国家精品课程过程中的经验，认真研究该课程的特点，分析当前出版的C语言程序设计教材的基础上编写完成的。

我们认为，作为程序设计教材应该重点培养学生的编程能力，同时应该掌握扎实的语法知识。学生创新能力的培养是潜移默化的，作为教材应该在学生创新能力方面加以引导，培养学生发现问题、分析问题、解决问题的能力。

本教材的主要特点如下：

(1) 强化程序设计能力培养。

本教材从实际问题需求出发引出理论，从个体到一般，以点带面。根据程序设计的需要，引出相关的知识点，将知识学习和使用密切结合，加深了理解，也避免了枯燥的学用分离的语法学习，使学习者明确为什么引出这些知识点，强化了知识点在程序设计中的应用。

(2) 注重学生创新思维的培养。

教材贯穿了提出需要解决的问题、分析问题、引出概念、讲解知识点、程序实现的编写思路。通过给出实际问题，分析问题的特点，引导学生思考，然后给出解决问题的思路。通过潜移默化的作用，培养学生的创新思维和分析问题解决问题的能力。

(3) 突出实用性和趣味性。

在例题的选择上力求实用性和趣味性，以此提高应用程序设计的能力和学习兴趣。内容的组织编排强化实践教学，突出编程能力培养。所有例题不是简单地给出程序，而是首先分析问题，提出解题思路，再给出解决方案。将算法和数据结构结合起来，培养学生编程能力。

(4) 强调学用结合和规范化编程。

学习的目的是为了使用。因此，知识点的学习紧密结合使用，知识点基本上采用了学了即用的原则。一方面加强了知识点的理解和巩固，另一方面知道这些知识点在什么地方用和如何用。避免为了学习而学习，学而不用的问题。努力引导学生养成良好的编程习惯，编

写风格优美、可读性强、易于维护的程序代码。

编写一本精品教材绝非易事，尽管我们力图贯彻突出程序设计能力的培养和启迪创新思维的思想，但是由于水平有限，还有许多不尽如人意的地方。另外，在编写过程中，由于时间紧迫，难免存在问题和不足，敬请同行和广大读者提出宝贵意见，以便我们在以后的版本中改进。

本教材由济南大学C语言课程组组织编写，参加编写的有杨波、刘明军、潘玉奇、蔺永政、袁宁等。董吉文教授、周劲副教授及课程组的其他老师在教材的编写工作中提出了一些很好的建议，在此表示感谢。

北京大学李晓明教授审阅了全书，提出了非常中肯和宝贵的意见。对全书的定稿给予了很大帮助。

本书配备完整的教学课件和案例源代码，有需要的老师可登录课程网站下载，课程网站网址为：http://c.ujn.edu.cn。也可与作者E-mail联系，联系邮箱为：ise_panyq@ujn.edu.cn。

作者

2010年7月于济南

目 录

CONTENTS

* 本书带 * 部分内容为选学内容。

第 1 章 程序设计概述

程序(Program)是计算任务的处理对象和处理规则的描述。任何以计算机为处理工具的任务都是计算任务。处理对象是数据(如数字、文字、图形、图像、声音等,它们只是表示,而无含义)或信息(数据及有关的含义)。处理规则一般指处理动作和步骤。程序设计(Programming)是设计、编制和调试程序的方法与过程。程序设计是软件构造活动中的重要组成部分,程序设计往往以某种程序设计语言为工具,给出这种语言下的程序,程序设计是一项创造性的工作,是伴随着计算机应用和程序设计语言的发展而发展起来的一门学科。对于不同的计算机语言,其程序设计的基本方法是大致相同的。本书以 C 语言程序设计为主线,介绍程序设计的基本概念和基本方法。

本章主要内容:简单介绍了计算机软件、程序设计语言的分类,重点讨论了程序设计的方法步骤、数据结构的概念、算法及其描述;介绍了程序设计的一般方法,介绍了结构化程序设计、面向对象程序设计以及面向服务程序设计的特点。通过本章的学习对程序设计的基本概念和方法有基本了解,了解算法和数据结构的概念,掌握算法的表示方法。

1.1 计算机软件

计算机软件(Computer Software)是计算机系统中的程序及其文档。程序是计算任务的处理对象和处理规则的描述;文档是为了便于了解程序所需要的简明资料。

1.1.1 程序设计语言

软件语言(Software Language)是用于书写计算机软件的语言。它主要包括需求定义语言、功能性语言、设计性语言、实现性语言以及文档语言等。

程序设计语言(Programming Language)是用于书写计算机程序(习惯上指实现级语言程序)的语言。语言的基础是一组记号和一组规则。根据规则由记号构成的记号串的总体就是语言。在程序设计中,这些记号串就是程序。程序设计语言的基本成分有:

① 数据成分:用于描述程序所涉及的数据。

② 运算成分:用于描述程序中所包含的运算。

③ 控制成分:用于表达程序中的控制构造。

④ 传输成分:用于表达程序中数据的传输。

按照语言与硬件的关联程度不同,有低级语言和高级语言之分。计算机程序设计语言的发展,经历了从机器语言、汇编语言到高级语言的历程,大量的应用程序开发通常采用高级语言,机器语言和汇编语言通常用于与机器硬件关联密切的程序设计。

1. 机器语言

机器语言(Machine Language)是用二进制代码表示的计算机能直接识别和执行的一种机器指令的集合。它是计算机的设计者通过计算机的硬件结构赋予计算机的操作功能。机器语言具有灵活、直接执行和速度快等特点。不同型号的计算机其机器语言是不相通的,按着一种计算机的机器指令编制的程序,一般情况下不能在另一种计算机上执行。

用机器语言编写程序,编程人员要首先熟记所用计算机的全部指令代码和代码的含义,需要自己处理每条指令和每一数据的存储分配和输入输出,需要记住编程过程中每步所使用的工作单元处在何种状态。机器语言程序全是0和1的指令代码,程序的编写、调试和修改非常复杂。

2. 汇编语言

汇编语言(Assembly Language)是面向机器的程序设计语言。在汇编语言中,用助记符代替操作码,用地址符号或标号代替地址码,方便记忆和使用。使用汇编语言编写的程序,机器不能直接识别,需要将汇编语言翻译成机器语言,这种起翻译作用的程序叫编译程序,把汇编程序翻译成机器语言的过程称为编译。

汇编语言是面向机器的低级语言,保持了机器语言的优点,具有直接和简捷的特点,可有效地访问、控制计算机的各种硬件设备,目标代码简短,占用内存少,执行速度快。

3. 高级语言

高级语言(High-level Language)是一种更接近于自然语言的计算机程序设计语言,具有很强的描述能力,高级语言不直接依赖于具体的计算机硬件。使用高级语言编写的程序不能直接运行,需要将其转换为机器语言才能运行,通常的转换方式有解释和编译两种。

在计算机发展历史上,先后出现过几百种高级语言。其中,影响较大、使用较普遍的有FORTRAN,ALGOL,COBOL,BASIC,LISP,PL/1,PASCAL,C,PROLOG,Ada,C++、Visual C++,Visual Basic,Java等。

1.1.2 程序设计

程序设计是目标明确的智力活动,是根据计算机要完成的任务,提出相应的要求,在此基础上设计数据结构和算法,然后再编写相应的程序代码并测试该代码运行的正确性,直到能够得到正确的运行结果为止。

由于程序是软件的本体,软件的质量主要是通过程序的质量来体现,在软件研究中,程序设计的工作非常重要,内容涉及有关的基本概念、工具、方法以及方法学等。程序设计的具体步骤如下:

① 方案确定:程序将以数据处理的方式解决客观世界中的问题,因此进行程序设计首先应该将实际问题用数学语言描述出来,形成一个抽象的,具有一般性的数学问题。也就是建立问题的抽象数学模型,然后制定解决该模型所代表的数学问题的算法。数学模型精确地阐述了模型本身所涉及的各种概念、已知条件、所求结果,以及已知条件与所求结果之间的联系等各方面的信息。数学模型是进一步确定解决所代表的数学问题算法的基础。利用

数学模型和算法的结合得到问题的解决方案。

② 算法描述：具体的解决方案确定后，需要对所采用的算法进行描述，算法的初步描述可以采用自然语言方式、程序流程图、N-S 图、伪代码等。算法描述应简单明确，能够比较明显地展示程序设计思想，是进行程序调试的重要参考。

③ 数据描述：根据程序设计的目标及对数据的处理要求，确定所处理数据的表示方式，即数据结构。算法和数据结构密切相关，两者应相互结合。

④ 编写程序：使用计算机系统提供的某种程序设计语言，根据上述算法描述和数据结构，将已设计好的算法表达出来。使得非形式化的算法转变为形式化的由程序设计语言表达的算法，这个过程称为程序编制(编码)。程序的编写过程需要反复调试才能得到可以运行且结果“正确”的程序。

⑤ 程序测试：程序编写完成后必须经过科学的、严格的测试，才能最大限度地保证程序的正确性。同时，通过测试可以对程序的性能作出评估。

程序设计是很讲究方法的，一个良好的设计思想方法能够大大提高程序的高效性、合理性。程序设计是软件开发工作的重要部分，而软件开发是工程性的工作，所以要有规范。

1.1.3 数据结构

数据结构(Data Structure)是计算机存储、组织数据的方式。数据结构是指相互之间存在一种或多种特定关系的数据元素的集合。通常情况下，精心选择的数据结构可以带来更高的运行或者存储效率。数据结构往往同高效的检索算法和索引技术有关。数据结构一般包括以下三方面内容：

① 数据元素之间的逻辑关系，也称数据的逻辑结构(Logical Structure)。数据的逻辑结构是从逻辑关系上描述数据，与数据的存储无关，是独立于计算机的。数据的逻辑结构可以看作是从具体问题抽象出来的数学模型。

② 数据元素及其关系在计算机存储器内的表示，称为数据的存储结构(Storage Structure)。数据的存储结构是逻辑结构用计算机语言的实现(也称为映像)，它依赖于计算机语言。对机器语言而言，存储结构是具体的。一般，只在高级语言的层次上讨论存储结构。

③ 数据的运算，即对数据施加的操作。数据的运算定义在数据的逻辑结构上，每种逻辑结构都有一个运算的集合。最常用的检索、插入、删除、更新、排序等运算实际上只是在抽象的数据上所施加的一系列抽象的操作。所谓抽象的操作，是指我们只知道这些操作是“做什么”，而无须考虑“如何做”。只有确定了存储结构之后，才考虑如何具体实现这些运算。

1.1.4 算法

在客观世界中，我们做任何事情都有一定的方法和步骤。比如，要得到某门课程的学分，就包括选课、听课、完成作业、参加考试等环节。如果考试不及格，还要按规定补考或重修。这就是获得课程学分的方法步骤。用计算机解决问题也是按照相应的步骤一步一步完成的。这些方法步骤都可以称其为算法(Algorithm)。广义地说，算法是为解决问题而采取的方法和步骤。

1. 算法的概念

在程序设计中,算法是一系列解决问题的清晰指令。也就是说,能够对一定规范的输入,在有限时间内获得所要求的输出。算法可以理解为有基本运算及规定的运算顺序所构成的完整的解题步骤。或者看成按照要求设计好的有限的确切的计算序列,并且这样的步骤和序列可以解决一类问题。如果一个算法有缺陷,或不适合于某个问题,执行这个算法将不会解决这个问题。不同的问题有不同的算法,不同的算法可能用不同的时间、空间或效率来完成同样的任务。一个算法的优劣可以用空间复杂度与时间复杂度来衡量。

2. 算法的特征

一个算法应该具有以下五个重要的特征:

① 有穷性:一个算法必须保证执行有限步之后结束。

② 确切性:算法的每一步骤必须有确切的定义。

③ 可行性:算法原则上能够精确地运行,而且人们用笔和纸做有限次运算后即可完成。

④ 输入:一个算法有0个或多个输入,以刻画运算对象的初始情况,所谓0个输入是指算法本身定出了初始条件。

⑤ 输出:一个算法有一个或多个输出,以反映对输入数据加工后的结果。没有输出的算法是毫无意义的。

计算机科学家尼克劳斯·沃思曾著过一本著名的书《数据结构+算法=程序》,可见算法在计算机科学界与计算机应用界的地位。

【例1.1】 对给定有序数列{3,5,11,17,21,23,28,30,32,50},查找关键字值为30的数据元素。

算法一:将30按顺序与给定数列逐一比较直到找到为止。这种算法在最坏的情况下可能需要比较整个序列。

算法二:查找过程中采用跳跃式方式查找,即先以有序数列的中点位置为比较对象,如果要找的元素值小于该中点元素,则将待查序列缩小为左半部分,否则为右半部分。通过一次比较,将查找区间缩小一半。经$\log 2^n$次比较就可以完成查找过程。但是其缺点也很明显,要求查找数列必须有序,而对所有数据元素按大小排序是非常费时的操作。

可见,不同的算法其效率有很大的差别,不同的算法适合的场合也不同。

3. 算法设计与分析的常用方法

① 递推法:递推法把问题分成若干步,找出相邻几步的关系,从而达到目的的方法称为递推法。它是利用问题本身所具有的一种递推关系求解问题的一种方法,是迭代算法的最基本形式。

② 递归法:递归指的是一个过程,函数不断引用自身,直到引用的对象已知。

③ 穷举搜索法:穷举搜索法是对可能是解的众多候选解,按某种顺序进行逐一枚举和检验,并从中找出那些符合要求的候选解作为问题的解。

④ 贪婪法:贪婪法是一种不追求最优解,只希望得到较为满意解的方法。贪婪法一般可以快速得到满意的解,因为它省去了为找最优解要穷尽所有可能而必须耗费的大量时间。贪婪法常以当前情况为基础作最优选择,而不考虑各种可能的整体情况,所以贪婪法不要回溯。