



21st CENTURY

实用规划教材

21世纪全国高职高专
计算机系列实用规划教材

C++ 程序设计

教程与实训

主 编 严仲兴
副主编 韩最蛟 魏米亚

内容特点:

- 全书系统地介绍了C++面向对象程序设计的基本知识以及基本编程思想，语法结构深入浅出，重点介绍了抽象、继承、封装、多态及模板等面向对象程序的重要特征。
- 本书选材适当，通俗易懂，实例丰富，重点突出，书中实例全部在Visual C++ 6.0集成开发环境下调试运行并通过。
- 本书特别适合作为高职高专院校的计算机职业技术类学生学习C++语言的教材，也可作为其他理工科专业学生学习C++语言的教材，同时也适合作为相关技术人员的参考书。



北京大学出版社
PEKING UNIVERSITY PRESS

21 世纪全国高职高专计算机系列实用规划教材

C++程序设计教程与实训

主 编	严仲兴	
副主编	韩最蛟	魏米亚
参 编	王广胜	贺玉珍
	邢培振	冯国华



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

全书系统地介绍了 C++面向对象程序设计的基本知识以及基本编程思想,语法结构深入浅出,重点介绍了抽象、继承、封装、多态及模板等面向对象程序的重要特征。另外,介绍了基于 MFC 的 C++可视化程序设计方法。全书共分 8 章,主要内容包括: C++语言的基本概念和面向对象程序设计的基本思想;数据类型、语句结构、指针、函数等使用方法;类与对象;多态性和模板;输入输出流;基于 MFC 的可视化程序设计方法。书中实例全部在 Visual C++ 6.0 集成开发环境下调试运行并通过。

本书选材适当,通俗易懂,实例丰富,重点突出,特别适合作为高职高专院校的计算机职业技术类学生学习 C++语言的教材,也可作为其他理工科专业学生学习 C++语言的教材,同时也适合作为相关技术人员的参考书。

图书在版编目(CIP)数据

C++程序设计教程与实训 / 严仲兴主编. —北京: 北京大学出版社, 2006.1

(21 世纪全国高职高专计算机系列实用规划教材)

ISBN 7-301-10265-8

I. C… II. 严… III. C 语言—程序设计—高等学校: 技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2005) 第 142934 号

书 名: C++程序设计教程与实训

著作责任者: 严仲兴 主编

责任编辑: 刘 丽

标准书号: ISBN 7-301-10265-8/TP · 0849

出 版 者: 北京大学出版社

地 址: 北京市海淀区成府路 205 号 100871

网 址: <http://cbs.pku.edu.cn> <http://www.pup6.com>

电 话: 邮购部 62752015 发行部 62750672 编辑部 62750667

电子信箱: pup_6@163.com

排 版 者: 北京东方人华北大彩印中心 电话: 62754190

印 刷 者: 河北涿县鑫华书刊印刷厂

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 毫米×1092 毫米 16 开本 16.75 印张 384 千字

2006 年 1 月第 1 版 2006 年 1 月第 1 次印刷

定 价: 22.00 元

《21 世纪全国高取高专计算机系列实用规划教材》

专家编审委员会

主 任 刘瑞挺

副主任 (按拼音顺序排名)

陈玉国 崔锁镇 高文志 韩希义

黄晓敏 魏 峥 谢一风 张文学

委 员 (按拼音顺序排名)

安志远 丁亚明 杜兆将 高爱国 高春玲 郭鲜凤

韩最蛟 郝金镇 黄贻彬 季昌武 姜 力 李晓桓

连卫民 刘德军 刘德仁 栾昌海 罗 毅 慕东周

彭 勇 齐彦力 沈凤池 陶 洪 王春红 闻红军

武凤翔 武俊生 徐 红 徐洪祥 徐受容 许文宪

严仲兴 杨 武 于巧娥 袁体芳 张 昕 赵 敬

赵润林 周朋红 訾 波

信息技术的职业化教育

(代丛书序)

刘瑞挺/文

北京大学出版社第六事业部组编了一套《21 世纪全国高职高专计算机系列实用规划教材》。为此，制订了详细的编写目的、丛书特色、内容要求和风格规范。在内容上强调面向职业、项目驱动、注重实例、培养能力；在风格上力求文字精练、图表丰富、脉络清晰、版式明快。

一、组编过程

2004 年 10 月，第六事业部林章波主任、葛昊晗副主任开始策划这套丛书，分派编辑深入各地职业院校，了解教学第一线的情况，物色经验丰富的作者。2005 年 1 月 15 日在济南召开了“北大出版社高职高专计算机规划教材研讨会”。来自 13 个省、41 所院校的 70 多位教师汇聚一堂，共同商讨未来高职高专计算机教材建设的思路和方法，并对规划教材进行了讨论与分工。2005 年 6 月 13 日在苏州又召开了“高职高专计算机教材大纲和初稿审定会”。编审委员会委员和 45 个选题的主、参编，共 52 位教师参加了会议。审稿会分为公共基础课、计算机软件技术专业、计算机网络技术专业、计算机应用技术专业 4 个小组对稿件逐一进行审核。力争编写出一套高质量的、符合职业教育特点的精品教材。

二、知识结构

职业生涯的成功与人们的知识结构有关。以著名侦探福尔摩斯为例，作家柯南道尔在“血字的研究”中，对其知识结构描述如下：

- ◆ 文学知识——无；
- ◆ 哲学知识——无；
- ◆ 政治学知识——浅薄；
- ◆ 植物学知识——不全面。对于药物制剂和鸦片却知之甚详。对毒剂有一般了解，而对于实用园艺却一无所知；
- ◆ 化学知识——精深；
- ◆ 地质学知识——偏于应用，但也有限。他一眼就能分辨出不同的土质。根据裤子上泥点的颜色和坚实程度就能说明是在伦敦什么地方溅上的；
- ◆ 解剖学知识——准确，却不系统；
- ◆ 惊险小说知识——很渊博。似乎对近一个世纪发生的一切恐怖事件都深知底细；
- ◆ 法律知识——熟悉英国法律，并能充分实用；
- ◆ 其他——提琴拉得很好，精于拳术、剑术。

事实上，我国唐朝名臣狄仁杰，大宋提刑官宋慈，都有类似的知识结构。审视我们自己，每个人的知识结构都是按自己的职业而建构的。因此，我们必须面向职场需要来设计教材。

三、职业门类

我国的职业门类分为 18 个大类：农林牧渔、交通运输、生化与制药、地矿与测绘、材料与能源、土建水利、制造、电气信息、环保与安全、轻纺与食品、财经、医药卫生、旅游、公共事业、文化教育、艺术设计传媒、公安、法律。

每个职业大类又分为二级类，例如电气信息大类又分为 5 个二级类：计算机、电子信息、通信、智能控制、电气技术。因此，18 个大类共有 75 个二级类。

在二级类的下面，又有不同的专业。75 个二级类共有 590 种专业。俗话说：“三百六十行，行行出状元”，现代职业仍在不断涌现。

四、IT 能力领域

通常信息技术分为 11 个能力领域：规划的能力、分析与设计 IT 解决方案的能力、构建 IT 方案的能力、测试 IT 方案的能力、实施 IT 方案的能力、支持 IT 方案的能力、应用 IT 方案的能力、团队合作能力、文档编写能力、项目管理能力以及其他能力。

每个能力领域下面又包含若干个能力单元，11 个能力领域共有 328 个能力单元。例如，应用 IT 方案能力领域就包括 12 个能力单元。它们是操作计算机硬件的能力、操作计算机软件包的能力、维护设备与耗材的能力、使用计算机软件包设计机构文档的能力、集成商务计算机软件包的能力、操作文字处理软件的能力、操作电子表格应用软件的能力、操作数据库应用软件的能力、连接到互联网的能力、制作多媒体网页的能力、应用基本的计算机技术处理数据的能力、使用特定的企业系统以满足用户需求的能力。

显然，不同的职业对 IT 能力有不同的要求。

五、规划梦想

于是我们建立了一个职业门类与信息技术的平面图，以职业门类为横坐标、以信息技术为纵坐标。每个点都是一个函数，即 $IT(Professional)$ ，而不是 $IT+Professional$ 单纯的相加。针对不同的职业，编写它所需要的信息技术教材，这是我们永恒的主题。

这样组合起来，就会有 $IT((328)*(Pro(590)))$ ，这将是一个非常庞大的数字。组织这么多的特色教材，真的只能是一个梦想，而且过犹不及。能做到 $IT((11)*(Pro(75)))$ 也就很不容易了。

因此，我们既要在宏观上把握职业门类的大而全，也要在微观上选择信息技术的少而精。

六、精选内容

在计算机科学中，有一个统计规律，称为 90/10 局部性原理(Locality Rule)：即程序执行的 90% 代码，只用了 10% 的指令。这就是说，频繁使用的指令只有 10%，它们足以完成 90% 的日常任务。

事实上，我们经常使用的语言文字也只有总量的 10%，却可以完成 90% 的交流任务。同理，我们只要掌握了信息技术中 10% 频繁使用的内容，就能处理 90% 的职业化任务。

有人把它改为 80/20 局部性原理，似乎适应的范围更广些。这个规律为编写符合职业教育需要的精品教材指明了方向：坚持少而精，反对多而杂。

七、职业本领

以计算机为核心、贴近职场需要的信息技术已经成为大多数人就业的关键本领。职业教育的目标之一就是培养学生过硬的 IT 从业本领，而且这个本领必须上升到职业化的高度。

职场需要的信息技术不仅是会使用键盘、录入汉字，而且还要提高效率、改善质量、降低成本。例如，两位学生都会用 Office 软件，但他们的工作效率、完成质量、消耗成本可能有天壤之别。领导喜欢谁？这是不言而喻的。因此，除了道德品质、工作态度外，必须通过严格的行业规范和个人行为规范，进行职业化训练才能养成正确的职业习惯。

我们肩负着艰巨的历史使命。我国人口众多，劳动力供大于求的矛盾将长期存在。发展和改革职业教育，是我国全面建设小康社会进程中一项艰巨而光荣的任务，关系到千家万户人民群众的切身利益。职业教育和高技能人才在社会主义现代化建设中有特殊的作用。我们一定要兢兢业业、不辱使命，把这套高职高专教材编写好，为我国职业教育的发展贡献一份力量。

刘瑞挺教授 曾任中国计算机学会教育培训委员会副主任、教育部理科计算机科学教学指导委员会委员、全国计算机等级考试委员会委员。目前担任的社会职务有：全国高等院校计算机基础教育研究会副会长、全国计算机应用技术证书考试委员会副主任、北京市计算机教育培训中心副理事长。

本系列教材编写目的和教学服务

本系列教材在遍布全国的各位编写老师的共同辛勤努力下，在编委会主任刘瑞挺教授和其他编审委员会成员的指导下，在北京大学出版社第六事业部的各位编辑刻苦努力下，本系列教材终于与广大师生们见面了。

教材编写目的

近几年来，职业技术教育事业得以蓬勃的发展，全国各地的高等职业院校以及高等专科学校无论是从招生人数还是学校的软、硬件设施上都达到了相当规模。随着我国经济的高速发展，尽快提高职业技术教育的水平显得越来越重要。教育部提出：职业教育就是就业教育，也就是说教学要直接面对就业，强调实践。不但要介绍技术，更要介绍具体应用，注重技术与应用的结合。本套教材的主要编写思想如下。

1. 与发达国家相比，我国职业技术教育教材的发展比较缓慢并且滞后，远远跟不上职业技术教育发展的需求。我们常常提倡职业教育的实用性，但在课堂教学中仍然使用理论性和技术性教材进行职业实践教学。针对这种现状，急需推出一系列切合当前教育改革需要的高质量的优秀职业技术实训型教材。

2. 本套教材总结了目前优秀计算机职业教育专家的教学思想与经验，与广大职业教育一线老师共同探讨，最终落实到本套教材中，开发出一套适合于我国职业教育教学目标和教学要求的教材，它是一套能切实提高学生专业动手实践能力和职业技术素质的教材。

3. 社会对学生的职业能力的要求不断提高，从而催化出了许多新型的课程结构和教学模式。新型教学模式是必须以工作为基础的模仿学习，它是将学生置于一种逼真的模拟环境中，呈现给学生的是具有挑战性、真实性和复杂性的问题，使学生得到较真实的锻炼。

4. 教材的结构必须按照职业能力的要求创建并组织实施新的教学模式。教学以专项能力的培养展开，以综合能力的形成为目标。能力的培养既是教学目标，又是评估的依据和标准。

5. 本套的重点是先让学生实践，从实践中领悟、总结理论，然后再学习必要的理论，用理论指导实践。从这一个循环的教学过程中，学生的职业能力将得到极大的提高。

教学服务

1. 提供电子教案

本系列教材绝大多数都是教程与实训二合一，每一本书都有配套的电子教案，以降低任课老师的备课强度，此课件可以在我们网站上随时下载。

2. 提供教学资源下载

本系列教材中涉及到的实例(习题)的原始图片和其他素材或者是源代码、原始数据等文件，都可以在我们网站上下载。

3. 提供多媒体课件和教师培训

针对某些重点课程，我们配套有相应的多媒体课件。对大批量使用本套教材的学校，我们会免费提供多媒体课件，另外还将免费提供教师培训名额，组织使用本套教材的教师进行相应的培训。

前 言

面向对象程序设计(OOP)是一种围绕现实世界的真实模型来组织的模型化的程序设计方法,将数据与数据的处理抽象成对象的概念,然后以对象为中心描述和规划系统。在编程中,将对象视为一个数据集合,数据的处理具有很好的稳定性,这样,更符合人们的思维习惯。面向对象的程序设计方法,具有很好的可重用性和可维护性,且易于修改与扩充。

C++语言完全支持面向对象的程序设计方法,与C语言高度兼容并且功能强大、高效灵活,得到计算机编程人员的广泛重视和推崇。C++语言支持传统的结构化程序设计方法,程序员必须从结构化程序设计方法的基本思路出发,充分理解过程与函数的概念,然后迅速过渡到对象化编程的思想方法上来,将过程与函数的概念抽象到类,使用类进行编程,直至面向对象的抽象描述。无论是面向过程,还是面向对象,都必须理解程序运行的计算机内存布局和语言的编译及设计。

本书引用大量实例程序代码,许多理论知识都是以实例的演示说明和归纳总结来进行讲解,从程序运行结果中,得出抽象的理论,使读者易于理解与掌握。书中的程序实例许多都选自作者多年的C++程序设计教学实践,选材适当,深入浅出,重点突出,具有很强的实用性和可读性。

本书尤其适合作为高职高专院校的计算机软件及其相关专业的学生学习C++面向对象的程序设计的教材,当然也可作为其他理工科专业C++程序设计的教材。

书中所有程序实例全部经过调试运行,程序运行环境为 Visual C++ 6.0 集成开发环境。

Microsoft Visual C++ 6.0 是微软公司推出的功能强大的开发工具包,在此环境中开发C++程序,可以大大地提高软件设计能力和开发速度。本书从实用的角度出发,介绍 Visual C++ 6.0 软件包的使用方法和编程技巧。通过具体的程序示例,详细介绍了菜单、对话框、常用工具栏等界面设计方法。MFC 库是基于 Windows 可视化程序设计基础,利用 MFC 向导可以开发设计出菜单、对话框、常用控件等图形化的界面。

全书共分 8 章,建议授课课时为 80 学时。第 1 章讲述了 C++语言的基础知识、面向对象编程的基本概念,以及 Visual C++的集成开发环境;第 2 章讲述了 C++语言的基本内容,包括数据类型、运算、各种语句结构、数组、指针、函数等,这些内容是构成 C++程序的基础;第 3、4、5 章重点介绍了 C++语言中面向对象的主要特征和编程思想,包括封装、继承、多态等;第 6 章主要介绍以流类库形式工作的 I/O 流系统;第 7、8 章重点介绍了可视化编程的基本方法,主要介绍 Windows 编程基础知识、使用 MFC 进行可视化编程的基本方法。

本书主编为苏州工业园区职业技术学院的严仲兴,副主编为中共四川省委党校四川行政学院的韩最蛟和华北水利水电学院职业技术学院的魏米亚,山西运城学院的贺玉珍、湖北生态工程职业技术学院的王广胜、华北水利水电学院职业技术学院的邢培振和冯国华参

编。各章的编写分工如下：第1章由王广胜编写，第2章由韩最蛟编写，第3章由严仲兴编写，第4、5章由贺玉珍编写，第6章由邢培振编写，第7章由冯国华编写，第8章由魏米亚编写。

由于编者水平有限，时间仓促，不妥之处在所难免，衷心希望广大读者批评指正，深表感谢。

编 者
2005 年 11 月

目 录

第 1 章 C++概述.....	1	2.3.3 指针.....	50
1.1 程序设计语言	1	2.3.4 引用.....	54
1.2 面向对象程序设计概念.....	2	2.4 函数	55
1.2.1 面向过程的程序设计.....	2	2.4.1 函数的定义方法	
1.2.2 面向对象的程序设计.....	4	和调用方法.....	55
1.2.3 数据的抽象与封装.....	6	2.4.2 函数的参数值的	
1.3 C++语言	7	传递和返回值.....	56
1.3.1 C++的起源	7	2.4.3 递归函数.....	60
1.3.2 评论 C++与 C.....	8	3.4.4 内联函数.....	63
1.3.3 C++的特点	8	2.4.5 指针与函数.....	63
1.4 C++程序的结构	9	2.5 小结	66
1.4.1 简单的 C++程序	9	2.6 实训	67
1.4.2 C++源程序编写过程.....	9	2.7 习题	69
1.5 C++的集成开发环境	10	第 3 章 类与对象.....	74
1.5.1 Visual C++6.0 集成		3.1 类与对象的基本概念.....	74
开发环境	10	3.1.1 类的定义.....	74
1.5.2 开发第一个程序.....	11	3.1.2 方法(成员函数).....	76
1.5.3 调试程序	14	3.1.3 对象的实现与引用.....	77
1.5.4 系统生成的文件.....	18	3.1.4 作用域.....	78
1.6 小结	19	3.2 构造函数与析构函数.....	79
1.7 实训	19	3.2.1 构造函数.....	79
1.8 习题	20	3.2.2 析构函数.....	82
第 2 章 C++语言的基本构成.....	22	3.2.3 构造函数的重载.....	83
2.1 C++程序的基本语法构成	22	3.2.4 构造函数的复制.....	83
2.1.1 数据类型	22	3.3 类的派生	86
2.1.2 表达式和基本运算.....	25	3.3.1 派生.....	86
2.2 C++基本语句	30	3.3.2 基类与派生类.....	86
2.2.1 顺序控制语句.....	30	3.3.3 派生类的构造与析构函数.....	89
2.2.2 选择控制语句.....	30	3.3.4 保护成员.....	90
2.2.3 循环控制语句.....	34	3.4 友元	90
2.3 数组、指针与引用	42	3.4.1 友元函数.....	91
2.3.1 数组的定义与应用.....	42	3.4.2 友元成员.....	91
2.3.2 字符串与字符数组.....	47	3.4.3 友元类.....	92

3.5 静态成员	92	5.4 小结	146
3.5.1 静态数据成员	92	5.5 实训	146
3.5.2 静态成员函数	94	5.6 习题	148
3.6 对象数组和对象指针	94	第6章 I/O 流	149
3.6.1 对象数组	94	6.1 输入输出流	149
3.6.2 对象指针	95	6.1.1 流的概念	149
3.6.3 this 指针	96	6.1.2 输出流	150
3.7 小结	97	6.1.3 输入流	151
3.8 实训	98	6.1.4 格式控制	152
3.9 习题	100	6.2 文件流	159
第4章 多态性	105	6.2.1 文件的概念	160
4.1 多态性	105	6.2.2 一个简单的文件流实例	161
4.2 重载	106	6.2.3 文本文件的操作	163
4.2.1 函数重载	106	6.2.4 二进制文件的访问操作	168
4.2.2 成员函数的重载	108	6.2.5 文件流常用函数	169
4.2.3 运算符的重载	111	6.3 字符串流	173
4.3 虚函数	122	6.4 小结	174
4.3.1 虚函数的引入	122	6.5 实训	175
4.3.2 虚函数的定义与使用方法	123	6.6 习题	179
4.3.3 虚函数和成员函数 重载的区别	125	第7章 MFC 编程基础	181
4.3.4 虚析构函数	126	7.1 Windows 编程基础	181
4.3.5 空的虚函数	127	7.1.1 Windows 操作系统 的技术特点	181
4.3.6 纯虚函数和抽象类	128	7.1.2 事件驱动模式	182
4.4 小结	130	7.1.3 消息处理机制	183
4.5 实训	130	7.1.4 窗口	184
4.6 习题	132	7.2 MFC 基础	185
第5章 模板	137	7.2.1 MFC 概述	185
5.1 模板的概念	137	7.2.2 MFC 应用程序框架	188
5.1.1 模板的引入	137	7.2.3 程序运行过程	188
5.1.2 模板的概念	138	7.2.4 创建一个简单的 MFC 应用程序	189
5.2 函数模板	139	7.3 MFC 应用程序分析	195
5.2.1 函数模板的定义	139	7.4 消息处理实例	196
5.2.2 函数模板的使用	139	7.4.1 鼠标消息处理	196
5.3 类模板	141	7.4.2 键盘消息处理	199
5.3.1 类模板的定义和使用	141	7.5 小结	201
5.3.2 类模板的应用	142		

7.6 实训	202	8.3.3 查找与替换对话框	234
7.7 习题	204	8.3.4 字体对话框	235
第 8 章 可视化编程	205	8.3.5 打印对话框	237
8.1 文档与视图的概念	205	8.4 菜单	238
8.1.1 文档视图结构	205	8.4.1 菜单资源	238
8.1.2 单文档(SDI)应用程序	206	8.4.2 编辑菜单资源	239
8.1.3 多文档(MDI)应用程序	209	8.4.3 映射菜单消息	241
8.2 对话框和控件	211	8.4.4 快捷菜单	242
8.2.1 对话框	211	8.5 小结	244
8.2.2 常用控件	216	8.6 实训	244
8.2.3 控件使用举例	225	8.7 习题	247
8.3 通用对话框	232	参考文献	248
8.3.1 颜色对话框	232		
8.3.2 文件对话框	233		

第 1 章 C++概述

教学提示：C++是在 C 语言的基础上拓展出来的，并引入了面向对象的机制。目前随着 C++语言的广泛应用，面向对象技术得到了大大的发展。C++是一种功能强大的程序设计语言，通过编程实践可以更具具体、更实际地理解面向对象的诸如抽象、封装、继承、多态等基本特性，为学好和掌握面向对象程序设计的基本思想和技术打下良好的基础。

教学目标：理解和掌握 C++程序的基本知识，简略了解面向对象技术的形成和发展过程，掌握 Visual C++集成开发系统的安装和使用方法，了解 C++的主要特点。

1.1 程序设计语言

人们经常使用语言或文字来表达思想、交流和互通信息，如汉语、英语、法语等。人类相互交流信息所用的语言被称为自然语言，但是计算机不能识别人类的自然语言，我们必须使用计算机能识别的，便于人与计算机之间进行信息交流的语言，这种语言被称为计算机程序设计语言，它是编写计算机程序的重要工具。

程序设计语言是人类与计算机交流的重要工具，它针对某些要解决的问题和任务用计算机能理解的描述形式来代替人类的工作。程序设计语言的目的是促使计算机为人类工作。人们用编程语言对问题和任务进行描述，同时随着使用群体的扩大，程序设计语言也成为人们相互之间交流过程描述与信息表达的工具。

随着计算机技术的不断发展，程序设计语言也在不断地发展，早期主要有机器语言和汇编语言，后来发展到高级语言。由 0 和 1 两种数码组成的二进制指令代码、为计算机直接识别执行的语言称为机器语言，机器语言的特点是不直观、难读难懂、编写不方便，但能被机器识别执行。为了克服机器语言难读难懂的缺点，人们用助词符来表示机器语言指令，这种语言被称为汇编语言。如以“ADD”表示加法，“MOV”表示传送，可以看出汇编语言比机器语言直观、易读易记。汇编语言指令代码与机器语言指令代码是一一对应的关系。但汇编语言是计算机不能直接识别执行的，执行之前必须先翻译成机器语言指令，才能被计算机识别执行，这一翻译的过程称为汇编。

高级语言是一种接近自然语言和数学语言的具有通用性的计算机语言，更容易被人们理解和阅读。常用于科学计算和数据处理软件的编写。高级语言当然也要经过编译才能被计算机识别执行。目前大部分程序员使用的都是高级语言。C++则是一种常用的高级语言。

高级语言又大致经历了基础语言、结构化程序设计和面向对象程序设计三个阶段。

基础语言是在 20 世纪 50 年代，Fortran、Cobol 和 Basic 等语言为代表，奠定了程序设计的最初基础；结构化程序设计是随着软件规模的扩大而提出的，如 Pascal 和 C 语言的结构化程序设计语言以及软件工程的思想，大大提高了程序设计的效率与质量。面向对象程

程序设计是到了 20 世纪 70 年代以后, 由于高技术的研究和对图形、图像和声音等非结构化数据类型的需求, 提出对这些非结构化数据信息处理, 要求将软件分割成若干个软件模块, 每个软件模块具有更相对的独立性, 便于对程序的管理与扩充, 由此, 社会生产的实践使面向对象的程序设计语言逐渐成为主流。程序设计的焦点从过程化程序设计方法转移到了抽象数据类型 (ADT) 的程序设计上, 通常称之为基于对象(object based)的程序设计。在基于对象的程序设计方法中, 我们通过一组数据抽象来建立问题的模型, 在 C++ 中把这些抽象称为类(class)。

我们设计编写的程序由两个主要方面组成: 算法的集合和数据的集合。算法的集合是将指令组织成程序来解决某个特定的问题, 而数据的集合是算法在这些数据上操作以提供问题的解决方案。

1.2 面向对象程序设计概念

程序设计语言随着计算机的不断发展, 出现了很多变化。最早出现了机器语言, 随后又出现了汇编语言, 之后结构化设计语言出现, 使较为复杂的问题显得更为简单了。随着项目规模的扩大, 面向对象的程序设计取而代之。

1.2.1 面向过程的程序设计

从程序设计方法的角度来说, 程序设计是组织成一定形式的操作序列, 同一问题的不同处理方法或操作序列, 反映了程序设计的不同方法。科学计算的数据往往是非常复杂的, 依赖数据类型可以完成对数据的各种操作, 如数据复制或数据赋值等操作。结构化程序设计思想的思路是: 自顶向下, 逐步求精; 其程序结构是按功能分成若干个基本模块, 各模块之间的关系尽可能简单, 在功能上相对独立。其调用关系如图 1.1 所示。

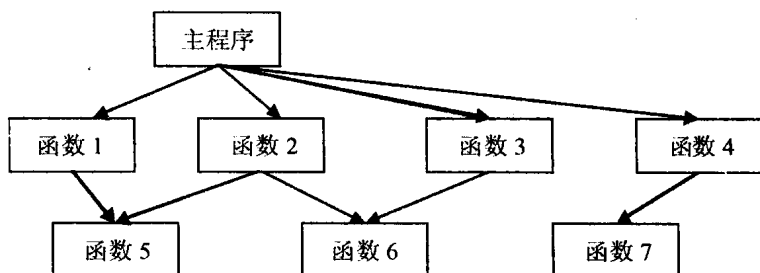


图 1.1 面向过程的程序功能模块

过程化的程序从系统的功能入手。系统是实现模块功能的一个集合, 由于用户需求的多样性和用户硬件更新的多变性, 按照功能划分设计的系统模块必然是易变的、不稳定的, 这样开发的模块可重用性不高。

1. 过程的实现

在过程化程序中, 过程是通过模块或函数来实现的, 每个过程只有一个入口和一个出

口。描述任何操作可以利用“顺序、选择、循环”三种基本控制结构，而这三种结构对于解决一般问题都是足够的。而在每个过程内部可以“顺序、选择、循环”三种基本结构组成，如图 1.2 所示。

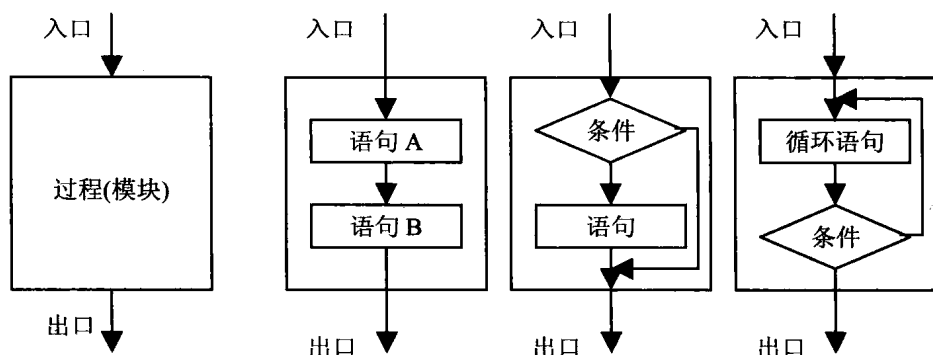


图 1.2 过程模块示意图

而在过程内部有可能是按序执行的“顺序”结构，也有可能是根据条件判断的选择结构，当然也可能是循环结构。而就外部来看，过程本身是独立的，我们只需要关心“入口”参数和“出口”参数，过程如何实现功能与外部的其他模块无关。

2. 模块化

过程也可以称为模块。模块由数据说明、可执行语句等程序元素组成，可以单独命名，而且可以通过名字来访问，也就是说，可以利用名字表示这一模块。如果把一个程序划分成若干个模块，每个模块完成一个子功能，把这些模块再组装成一个整体，就可以完成指定的复杂功能。这种程序设计方式被称为模块化。

模块化是为了使一个复杂的大型程序能被更好地管理和理解，如果一个大型程序只由一个模块组成，所有功能一一罗列，成千上万行的程序语句都写在一起，那是无法进行编写、维护和理解的。而如果把复杂问题分解成许多容易解决的小问题，分别处理，原来的问题也就很方便地进行分步处理了，这就是模块化的依据。当然，模块的数量也不能无限地增加，因为当模块的数量足够多时，虽然单个模块的规模较小，开发单个模块的工作量减小了，但设计和规划模块之间的接口所需要的工作量却大大增加了。因此，在程序规模不是太大的情况下，程序划分的模块也不能太小，否则模块间的接口成本将会过大。

3. 自顶而下

程序设计采用自顶而下，逐步细分的方法进行。在细分的过程中，充分运用过程控制结构和模块化划分原则。在一定的数据结构之下来设计对应的算法，然后分别实现数据结构设计和算法设计，用这种方法总能符合模块化的要求。

将一个解决复杂问题的大系统组分成若干个相对简单的小系统构成，而这些小系统又由更小的简单系统来构成，这样就把复杂问题简单化。程序设计也在于将一个复杂的程序结构划分为若干个相对独立而简单的程序模块。处于程序结构顶层的模块控制了程序的全部功能并影响全局，在程序结构底层的模块完成对数据的一次处理。

结构化程序设计在不大的程序规模上能起到良好的效果，容易编程与维护，也容易验

证程序的正确性。

而随着计算机技术的进一步发展,人类对计算机的依赖性越来越强,程序的规模不断扩大,模块数量呈指数级递增,各个模块之间的关系和数据传递错综复杂,结构化程序设计的规范已经不能保证程序的正确性、可维护性和重用性,必须将数据从算法中抽取出来,数据结构相对于算法而直接描述,基于对象的程序设计方法应运而生。

1.2.2 面向对象的程序设计

面向对象程序设计(Object-Oriented Program, OOP)被认为是继结构化程序设计方法的又一个重要方法。它是以程序设计为出发点,从描述客观世界中的事物开始,以对象代表一个具体的事物。而对象是把数据和数据的操作方法放在一起而形成的一个相互依存又不可分割的一个整体。

面向对象程序的设计方法关键在于加入了类及其继承性,以类为样板创建对象。这种程序设计方法将计算看作是一个系统开发过程。系统由对象组成,对象之间通过消息传递经历一系列的状态变化以完成各项任务。在面向对象的程序设计方法中,首先要决定所需要的类,每个类都设置有独立的属性和操作方法,并利用继承机制显式地共享共同的属性与操作方法。

面向对象程序设计方法将计算过程当作是一个系统开发过程。系统由对象组成,对象通过消息传递,经历一系列的状态变化来完成计算过程。若只使用对象和消息,则可以称为基于对象的程序设计方法,如果进一步要把所有对象都划分为类,则可称为基于类的程序设计方法。而真正地面向对象的程序设计方法是同时使用对象、类、继承和消息的方法。

1. 对象和类的概念

对象(Object)是客观世界的一个实体,如学校、商店、树木、计算机等都有可能看作成对象,它们之间有着千丝万缕的联系。每一个对象可以概括为两个要素:属性与行为。如学校,它有学校的校名、学校的老师、学校的学生等,另一方面,这个学校包含学校的各种活动,如学校招生、教学过程等。从编程的角度来考虑,一个对象包括对象的属性和对属性的操作。

一个“面向对象”的示例如图 1.3 所示。

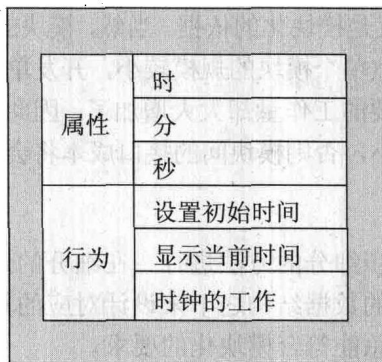


图 1.3 时钟类的示例

```
时钟类{
    时;
    分;
    秒;
```