

国外计算机科学经典教材

Mc
Graw
Hill Education

PROGRAMMING IN C++

Lessons and Applications

C++ 课堂教学 与编程演练

——科学与工程问题应用

(美) Timothy B. D'Orazio 著
侯普秀 冯 飞 译



清华大学出版社

国外计算机科学经典教材

C++课堂教学与编程 演练——科学与工程 问题应用

(美) Timothy B. D'Orazio 著
侯普秀 冯飞 译

清华大学出版社

北 京

Timothy B. D'Orazio

Programming in C++: lessons and applications

EISBN: 0-07-242412-5

Copyright © 2004 by The McGraw-Hill Companies, Inc.

Original language published by The McGraw-Hill Companies, Inc. All Rights reserved. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

Simplified Chinese translation edition is published and distributed exclusively by Tsinghua University Press under the authorization by McGraw-Hill Education(Asia) Co., within the territory of the People's Republic of China only (excluding Hong Kong, Macao SAR and Taiwan). Unauthorized export of this edition is a violation of the Copyright Act. Violation of this Law is subject to Civil and Criminal Penalties.

本书中文简体字翻译版由美国麦格劳-希尔教育出版(亚洲)公司授权清华大学出版社在中华人民共和国境内(不包括中国香港、澳门特别行政区和中国台湾地区)独家出版发行。未经许可之出口视为违反著作权法,将受法律之制裁。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

北京市版权局著作权合同登记号 图字: 01-2004-3175

版权所有, 翻印必究。举报电话: 010-62782989 13901104297 13801310933

本书封面贴有 McGraw-Hill 公司防伪标签, 无标签者不得销售。

图书在版编目(CIP)数据

C++课堂教学与编程演练——科学与工程问题应用/(美)奥瑞兹(D'Orazio,T.B.)著;侯普秀,冯飞译.

—北京:清华大学出版社,2004.9

书名原文: Programming in C++: lessons and applications

(国外计算机科学经典教材)

ISBN 7-302-08679-6

I. C… II. ①奥…②侯…③冯… III. C 语言—程序设计—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 047672 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客户服务: 010-62776969

组稿编辑: 曹 康

文稿编辑: 于 平

封面设计: 康 博

版式设计: 康 博

印 刷 者: 北京市世界知识印刷厂

装 订 者: 三河市金元装订厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印张: 56.5 字数: 1445 千字

版 次: 2004 年 9 月第 1 版 2004 年 9 月第 1 次印刷

书 号: ISBN 7-302-08679-6/TP·6220

印 数: 1~4000

定 价: 98.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话: (010)62770175-3103 或 (010)62795704

出版说明

近年来,我国高等学校的计算机学科教育进行了较大的改革,急需一批门类齐全、具有国际水平的计算机经典教材,以适应当前的教学需要。引进国外经典教材,可以了解并吸收国际先进的教学思想和教学方法,使我国的计算机学科教育能够与国际接轨,从而培育更多具有国际水准的计算机专业人才,增强我国信息产业的核心竞争力。Pearson、Thomson、McGraw-Hill、Springer、John Wiley 等出版集团都是全球最有影响的图书出版机构,它们在高等教育领域也都有着不凡的表现,为全世界的高等学校计算机教学提供了大量的优秀教材。为了满足我国高等学校计算机学科的教学需要,我社计划从这些知名的国外出版集团引进计算机学科经典教材。

为了保证引进版教材的质量,我们在全国范围内组织并成立了“清华大学计算机外版教材编审委员会”(以下简称“编委会”),旨在对引进教材进行审定、对教材翻译质量进行评审。

“编委会”成员皆为全国各类重点院校教学与科研第一线的知名教授,其中许多教授为各校相关院、系的院长或系主任。“编委会”一致认为,引进版教材要能够满足国内各高校计算机教学与国际接轨的需要,要有特色风格,有创新性、先进性、示范性和一定的前瞻性,是真正的经典教材。为了保证外版教材的翻译质量,我们聘请了高校计算机相关专业教学与科研第一线的教师及相关领域的专家担纲译者,其中许多译者为海外留学回国人员。为了尽可能地保留与发扬教材原著的精华,在经过翻译和编辑加工之后,由“编委会”成员对文稿进行审定,以最大程度地弥补和修正在前面一系列加工过程中对教材造成的误差和瑕疵。

由于时间紧迫和能力所限,本套外版教材在出版过程中还可能存在一些不足和遗憾,欢迎广大师生批评指正。同时,也欢迎读者朋友积极向我们推荐各类优秀的国外计算机教材,共同为我国高等学校的计算机教育事业贡献力量。

清华大学出版社

国外计算机科学经典教材

编审委员会

主任委员：

孙家广 清华大学教授

副主任委员：

周立柱 清华大学教授

委员（按姓氏笔画排序）：

王成山	天津大学教授
王 珊	中国人民大学教授
冯少荣	厦门大学教授
冯全源	西南交通大学教授
刘乐善	华中科技大学教授
刘腾红	中南财经政法大学教授
吉根林	南京师范大学教授
孙吉贵	吉林大学教授
阮秋琦	北京交通大学教授
何 晨	上海交通大学教授
吴百锋	复旦大学教授
李 彤	云南大学教授
沈钧毅	西安交通大学教授
邵志清	华东理工大学教授
陈 纯	浙江大学教授
陈 钟	北京大学教授
陈道蓄	南京大学教授
周伯生	北京航空航天大学教授
孟祥旭	山东大学教授
姚淑珍	北京航空航天大学教授
徐佩霞	中国科学技术大学教授
徐晓飞	哈尔滨工业大学教授
秦小麟	南京航空航天大学教授
钱培德	苏州大学教授
曹元大	北京理工大学教授
龚声蓉	苏州大学教授
谢希仁	中国人民解放军理工大学教授

前 言

本书为不具备实际编程经验的学生讲授 C++ 语言和面向对象的程序设计。本书适合于大学一年级、二年级的学生。学生最好具备高等数学、物理和化学方面的知识，但并非必须具备。

本书特色

- 提供了 30 多个案例研究。这些应用示例说明了如何解决许多领域的实际问题。
- 使用了多种方法便于学生阅读并理解源代码。每一个课堂教学部分都使用了多种技巧(代码注释、问题、主要内容，以及练习)，以使学生理解代码。
- 提前介绍程序的调试。在第 2 章的最后，向学生介绍了一些发现程序错误以及开发他们自己的程序的方法。
- 给出了跟踪及调试循环的详细描述。在许多章中都包括了开发及检查简单或复杂循环的可靠性的方法。
- 提供了许多图形。经常用图形来说明 C++ 中比较难的概念。
- 给出了从简单到复杂的应用示例。应用示例有很大的难度跨度，无论学生是否具有编程经验，都将面临挑战。
- 逐步地介绍程序开发的方法。每一个应用示例都说明了开发程序的结构化方法。
- 提前介绍类和对象。类和对象在第 8 章中就介绍了。但是，老师可以在第 14 章再讲解这些内容。
- 提供了数值方法示例。应用示例包含了一些基本的数值方法的说明，以及为其编码的方式。
- 提供了修改练习。这些练习适合于具有 2~3 个小时实验的课程。学生应该阅读特定的应用示例，为实验做准备。如果他们这样做了，许多这种练习都可以在 2~3 个小时内完成。
- 介绍了 UML。向学生描述了 UML 的基础知识，并且讲述了如何将 UML 图转换为 C++ 代码。
- 简单直观地介绍了 C++ 标准模板库。其目的是让学生能够尽快使用标准模板库。书中没有使用强制理解的术语，因此学生可以很快地使用这个库的基础部分。

本书结构

本书的每一章都分为两个部分——课堂教学及应用示例。课堂教学部分讲述 C++ 语言的基本内容和简单的编程技术，应用示例部分讲述程序设计。

课堂教学部分

课堂教学部分讲授了 C++ 语言的各个方面，并让学生知道在编写程序的时候，什么是可以利用的。这些内容包括基本符号、如何获得输入输出、运算符的意义、可用的库函数，以及如

何创建类并使用对象。还讲授了基本的编程操作，如查找一组数据的最大值、数组求和，以及使用随机数。

每一课堂教学部分都集中在一个示例程序上，并按顺序包含以下部分和一些教学观念。

各个部分	教学观念
主要内容——将要讲述的主题的列表	提醒学生将要学习的内容
引言——简要介绍主题的背景	让学生知道将要讲述内容的重要性，并让他们准备好阅读源代码
问题——向学生提出一些与程序有关的问题	在向学生解释代码之前，鼓励他们尝试着自己来解释代码
带注释的、源代码及输出——示例程序及输出，用来说明本课堂教学的主要内容	源代码说明要讲授的技巧，注释使学生关注新的程序设计内容中的重要细节。注释用方框括了起来，因为它们比“代码中的”说明更为突出，并且允许给出更多的细节。如果学生除了源代码之外没有阅读任何内容，至少可以学习该课堂教学的基本内容
说明——描述示例程序讲述的内容，以及如何将这些技巧应用于其他程序。提供了图解、表格，以及 C++语言特性的列表	说明部分为了解详细的技术内容提供了“软”环境。使用图解向学生提供概念及操作的可视图像。为了方便后面使用，列出了表格及 C++语言特性列表
每节练习(各章结尾)——判断对错、回答简短问题，以及基本程序任务	这些练习突出了课程的重点。问题相对简单，因此学生愿意在每一节的结尾做自我测试。这强化了所学的内容。程序任务非常简单。当学生仅关注语言部分而没有问题需要解决时，教师可以给他们分配这些任务
每节练习答案(各章结尾)——判断对错及简短问题的答案	答案很容易找到，这样便于给学生提供及时的反馈

总之，每个课堂教学的各个部分都鼓励学生阅读并理解 C++代码。目标是提高学生理解他人编写的程序的技能。当学完了这些课程后，学生就可以熟练地使用 C++语言，因为他们已经能够阅读并理解本书中的 C++代码。

应用示例部分

因为学生仅知道语言的运行方式和简单的编程操作是不够的，所以本书使用应用示例来说明程序设计以及如何解决实际问题。

应用示例是用结构化的多步方法(structured multistep method)开发的。学生可以按照同样的方法来创建他们自己的程序。为了设计实际的程序，在整个过程中作了许多强调。并对多页的程序都做了解释。

应用示例包含如下部分及教学观念。

各个部分	教学概念
问题描述——描述输入输出要求的问题说明	这个问题是为了强调特定的技能，以及说明它们为什么在编程解决问题时是需要的。例如，在具体的问题中使用复合运算符。因为本书是针对大一/大二学生的，所以特意没有涉及高等数学
解决方案：相关方程——描述问题的技术层面，包括需要使用的方程	使学生深入了解如何处理方程以获得适用于计算机解决方案的形式。对于有可能稍微超过学生背景知识的问题，在概念上做了进一步的描述
解决方案：具体示例——使用具体的输入数据进行示例计算，获得数字结果	显示如何使用相关方程及需要的步骤来获得数字结果
解决方案：类和对象——讨论程序中的类、对象，以及所需要的面向对象设计特性。仅在强调类和对象的程序中使用	说明开发类的整个过程和方法
解决方案：算法及代码段——列出了解决问题所需的步骤，并显示各个步骤的代码(对于复杂的示例程序)	说明如何将步骤转化为源代码。为了说明面向对象设计，在这个部分将类和对象组合起来
解决方案：源代码——解决问题的源代码及注解	说明如何将良好构想的设计(及算法)转换为完整的可运行的代码
说明——说明源代码的含义，提示可以用相似的方法来开发其他程序。并指出所显示代码的缺点	这部分使得学生超越所编写的程序。其中指明了如何使程序的功能更强大，以及在这种情况下可能会出现的问题
修改练习——可以通过修改给定的源代码而完成的任务	本部分的目的是使学生能使用他人编写的代码工作。有些修改相当简单，另一些修改需要做的工作多一些。所有的情况都要求学生理解源代码以进行修改。这练习适合于2个或3个小时的实验课。学生可以在实验前阅读特定的应用示例。如果仔细阅读了应用示例，许多练习可以在2~3个小时内完成
应用练习(各章的最后)——这些练习都是技术上的面向对象程序设计任务	这些练习大概是1~3个星期的任务。学生必须从头开始编写一个程序。可以使用在应用示例中学到的技术来开发这个程序

结构化的多步方法的目的是使学生遵循与程序设计相一致的方法。即使指导老师有不同的方法，学生通过阅读本书也将发现程序不是以随意的方式开发的。

应用示例详细显示了利用复杂循环、数组、类和指针工作的思考过程。

目 录

第 1 章 计算机和计算的基本原理	1
1.1 电子计算机的发展历史	1
1.2 体系结构	3
1.2.1 主存储器	3
1.2.2 中央处理器	5
1.2.3 外围设备	6
1.2.4 控制器及其与外围设备的通信	7
1.3 网络	7
1.4 使用位来表示字符和符号、整数、实数、地址以及指令	8
1.4.1 字符和符号	8
1.4.2 整数	9
1.4.3 实数	11
1.4.4 十六进制和八进制表示法	12
1.4.5 地址	14
1.4.6 指令	15
1.4.7 注释	15
1.5 编程语言	15
1.5.1 汇编语言	15
1.5.2 高级语言	16
1.6 软件	17
1.6.1 系统软件	18
1.6.2 应用软件	20
1.7 软件工程、C++语言和 C++编译器	20
1.7.1 C++中的程序组织和设计	21
1.7.2 C++语言的发展	23
1.7.3 C++语言的开发环境	23
1.8 关于本书以及如何最大程度地利用本书	25
1.8.1 课堂教学	26
1.8.2 应用示例	26

第2章 程序设计入门——程序结构、屏幕输出和注释	28
2.1 课堂教学：程序的基本结构	28
2.2 课堂教学：编写注释	35
2.3 课堂教学：输出中生成新行	38
2.4 应用示例：输出标志和表头	41
2.5 应用示例：调试	44
第3章 变量和算术运算	60
3.1 课堂教学：变量(1)——命名、声明、赋值以及显示变量值	60
3.2 课堂教学：变量(2)——创建常量以及更多显示变量值的信息	67
3.3 课堂教学：变量(3)——赋值以及显示单个字符数据	74
3.4 课堂教学：算术运算(1)——算术运算符和表达式	78
3.5 课堂教学：算术运算(2)——混合类型算术、复合赋值、运算符优先级， 以及类型强制转换	85
3.6 课堂教学：算术运算(3)——数学库函数和数据类型	96
3.7 应用示例：模式识别	101
3.8 应用示例：温度单位转换	105
3.9 应用示例：变位词解决方案	109
第4章 基本输入输出	124
4.1 课堂教学：从键盘读取数据	124
4.2 课堂教学：将输出写入文件	126
4.3 课堂教学：从文件读取数据	129
4.4 课堂教学：从键盘读取字符	132
4.5 应用示例：恺撒密码	136
4.6 应用示例：山崩分析	139
4.7 应用示例：长方体在光滑表面上滑动	143
第5章 作出决策	158
5.1 课堂教学：if 控制结构(1)——基础知识	158
5.2 课堂教学：if 控制结构(2)——简单的 if-else 控制结构	163
5.3 课堂教学：if 控制结构(3)——嵌套的 if-else 语句	166
5.4 课堂教学：逻辑运算符(1)——逻辑运算符	168
5.5 课堂教学：逻辑运算符(2)——关系表达式的值以及关系运算符 和逻辑运算符的优先次序	171
5.6 课堂教学：if-else-if 控制结构和 switch 控制结构	174
5.7 课堂教学：bool 数据类型	180
5.8 应用示例：求解一元二次方程	185

5.9 应用示例: 构件的负荷-变形	189
5.10 应用示例: 安排会议时间	193
第 6 章 迭代	212
6.1 课堂教学: while 循环(1)	212
6.2 课堂教学: while 循环(2)	215
6.3 课堂教学: do-while 循环	218
6.4 课堂教学: for 循环	220
6.5 课堂教学: 嵌套的 for 循环	225
6.6 应用示例: 面积计算(1)	232
6.7 应用示例: 面积计算(2)	234
6.8 应用示例: 温度单位转换	236
6.9 应用示例: 利息计算	239
6.10 应用示例: 数学级数	242
6.11 应用示例: 消息的加密及解密	249
第 7 章 函数	271
7.1 课堂教学: 函数基础	273
7.2 课堂教学: 返回单个值的函数	282
7.3 课堂教学: 通过引用传递	286
7.4 课堂教学: 作用域	289
7.5 课堂教学: 存储类	292
7.6 课堂教学: 带有默认参数的函数	298
7.7 课堂教学: 函数重载	302
7.8 课堂教学: 产生随机数	308
7.9 应用示例: 利用梯形规则计算积分	311
7.10 应用示例: 使用网格	316
7.11 应用示例: 对棋盘游戏的模拟	325
第 8 章 类和对象	350
8.1 课堂教学: 结构的基本概念——数据集合	350
8.2 课堂教学: 类的基本概念——数据和函数的集合	358
8.3 课堂教学: 构造函数(1)——无参数	367
8.4 课堂教学: 构造函数(2)——带参数	371
8.5 课堂教学: 构造函数(3)——重载和默认复制构造函数	375
8.6 应用示例: 线的交点	379
8.7 应用示例: 平行四边形的面积及平行六面体的体积	386

第 9 章 一维数值型数组	405
9.1 课堂教学: 简介	406
9.2 课堂教学: 初始化	411
9.3 课堂教学: 输入/输出和计算	415
9.4 课堂教学: 一维数组和函数	423
9.5 课堂教学: 使用数组作为数据成员类	426
9.6 课堂教学: 对象数组	430
9.7 应用示例: 计算电压测量数据	434
9.8 应用示例: 浪高分析	438
9.9 应用示例: 为 4 个玩家每人发 5 张牌	445
9.10 应用示例: 冒泡排序和交换排序	453
第 10 章 多维数值型数组	469
10.1 课堂教学: 声明、初始化和输出	470
10.2 课堂教学: 从一个文件中读取二维数组、存储分配和函数	475
10.3 应用示例: 矩阵-向量的操作	482
10.4 应用示例: 搜索和文件压缩	486
第 11 章 指针变量	498
11.1 课堂教学: 指针变量和地址的基础知识	498
11.2 课堂教学: 指针变量和函数	506
11.3 课堂教学: 指针变量和数组、指针变量和函数	508
11.4 课堂教学: 指针变量的其他用法	511
11.5 课堂教学: 从函数返回地址	513
11.6 课堂教学: 从多维数组返回地址	515
11.7 课堂教学: 指向对象的指针	522
11.8 课堂教学: 指针作为数据成员	524
11.9 课堂教学: 动态内存分配	527
11.10 应用示例: 地下污染羽烟的边界问题	531
第 12 章 字符数组	552
12.1 课堂教学: 声明、初始化和输出	552
12.2 课堂教学: 部分字符函数和字符串库函数	557
12.3 课堂教学: 二维字符数组	563
12.4 课堂教学: 键盘输入和文件输入	565
12.5 课堂教学: 声明和指针变量	569
12.6 课堂教学: 字符串和函数	574
12.7 应用示例: 创建一个电子数据表类型的程序	579

12.8	应用示例：单位转换程序	587
第 13 章	C++ string 类	603
13.1	课堂教学：C++ string 类(1)的介绍	603
13.2	课堂教学：C++ string 类(2)——成员函数	606
13.3	课堂教学：C++ string 类(3)——键盘输入和文件输入	612
13.4	课堂教学：C++ string 类(4)——字符串和函数	615
13.5	课堂教学：字符串、类和对象	617
13.6	应用示例：地震事件的报告分析	619
第 14 章	关于类、对象以及面向对象设计的更多内容	634
14.1	课堂教学：复制构造函数及析构函数	634
14.2	课堂教学：特殊限定符(1)——static	639
14.3	课堂教学：特殊限定符(2)——const	644
14.4	课堂教学：特殊限定符(3)——friend(函数)	652
14.5	课堂教学：特殊限定符(4)——friend(类)	656
14.6	课堂教学：运算符重载	660
14.7	课堂教学：UML 和面向对象设计	673
14.8	应用示例：交通事故分析	680
14.9	应用示例：汽车模拟	688
第 15 章	继承及多态性	703
15.1	课堂教学：继承(1)——基础知识	704
15.2	课堂教学：继承(2)——构造函数及析构函数	710
15.3	课堂教学：继承(3)——继承的层次	717
15.4	课堂教学：继承(4)——多重继承	720
15.5	课堂教学：虚函数及多态性	723
15.6	应用示例：电路电阻	727
第 16 章	数据结构和递归	736
16.1	课堂教学：一个编程人员定义的链表类	737
16.2	课堂教学：编程人员定义的堆栈类	749
16.3	课堂教学：编程人员定义的队列类	755
16.4	课堂教学：用户定义的二叉树类	760
16.5	课堂教学：采用递归调用的函数	769
16.6	应用示例：PI 的计算	777
16.7	应用示例：进行算术运算	783
16.8	应用示例：快速排序算法	794

第 17 章 模板与C++标准模板库	808
17.1 课堂教学：函数模板	808
17.2 课堂教学：类模板	812
17.3 课堂教学：标准模板库(1) —— 序列容器	817
17.4 课堂教学：标准模板库(2) —— 迭代器	823
17.5 课堂教学：标准模板库(3) —— 算法	826
第 18 章 其他主题	835
18.1 课堂教学：开发多文档程序	835
18.2 课堂教学：按位操作	840
18.3 课堂教学：二进制文件	850
附录A 数字的二进制和十进制表示法	857
附录B 字符数组函数	865
附录C 操作符	876
附录D ASCII表	879

第1章 计算机和计算的基本原理

本章主题

通过本章学习，您将了解到如何：

- 有效编程所需要掌握的主要计算机基础知识
- 存储指令的概念
- 基本的计算机体系结构——电子设备以及它们的连接
- 多种不同类型的信息如何按位存储
- 不同编程语言之间的一些区别
- 开发软件的相关步骤和软件工程的概念
- C++语言的背景知识
- 使用 IDE 创建一个可执行的程序
- 如何有效地使用本书

1.1 电子计算机的发展历史

电子计算机的发展历史相对较短。20 世纪 30 年代于 Iowa 州立大学研制的 Atanasoff-Berry 计算机(ABC)仅仅用于求解大量的联立方程。此后不久研制的 ENIAC(电子数字积分计算机)则用于通用计算的军用计算机。但是，每当执行一个新任务时，该计算机的接线必须以类似于老式交换机的工作方式进行重新配置。对以上两种计算机而言，一旦设置好电子接线，则只有数据(即需要添加的数或进行工作所需要的其他信息)才被输入到计算机的内存中。

数学家 John von Neumann(冯·诺伊曼)提出了一种重新配置接线过程的替代方案，从而在计算理论上取得了重大进展。他提出了将计算机的指令存储在自身内存中的概念。其中，指令会指示电子脉冲流经的方向和位置，这与导线指引电流的方式大致相同。对于基于冯·诺伊曼概念构造的计算机，数据和计算机指令都被输入到计算机的内存中。

图 1-1a 图示化地说明了一台采用直接接线方式工作的计算机。存储在单元 A-F 的数据直接连接到输出单元。在图 1-1a 中，单元 A、B 和 C 连接到 X，单元 D、E 和 F 与 Y 相连，因此， $X=A+B+C$ ，而 $Y=D+E+F$ 。如果我们想生成等式 $X=A+C+E$ 和 $Y=B+D+F$ ，那么计算机就必须重新进行接线。

与此相反，在图 1-1b 中，同时显示了数据和指令的输入单元，并且每个输入单元的接线同时接到输出单元 X 和 Y 上。不难想到，通过简单地使用正确的指令，我们可以方便地计算上一段中给出的所有 X 和 Y 等式。对于图 1-1b 中的计算机，为了解决新的问题，我们只需改变指令即可。因此，图 1-1b 中给出的计算机比图 1-1a 中的计算机高级。图 1-1b 表示的是一种现代计算机，因为指令和数据都被提供给计算机。本书将指导您如何通过 C++语言编写计算机指令。

编写计算机指令被称为编程(programming)。

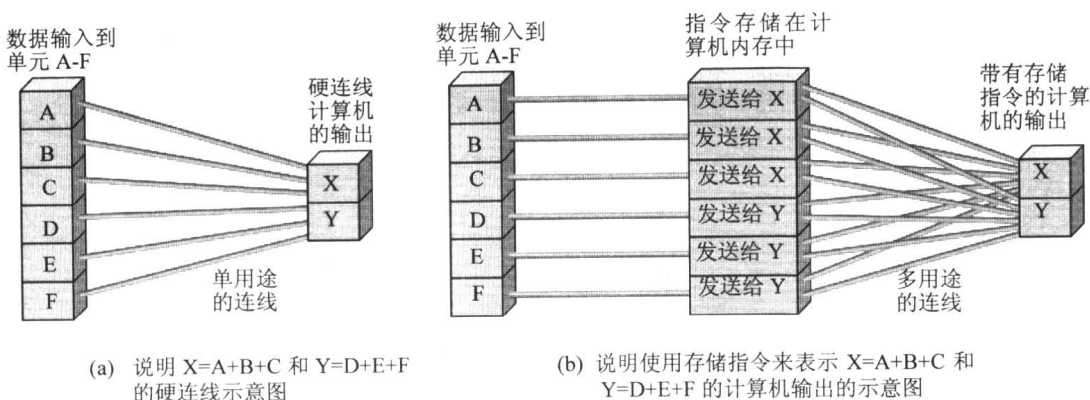


图 1-1 硬连线的计算机和带有存储指令的计算机的概念图

存储程序的概念带动了多用途计算机的发展,因为改变需要解决的问题类型,只需修改计算机的指令即可。实践表明,尽管每条指令输入自身很简单,但是将它与其他简单的指令正确组合起来却能够解决复杂的问题。例如,您可以将一组指令输入到计算机的内存中,使其执行长除法运算,同时可以将另一组指令输入到计算机,使其进行平方根计算。因此,许多这样的指令集可以进行组合来解决更为复杂的问题。由于计算机已经变得日益复杂,所以开发计算机指令已成为一门科学。

从早期的计算至今,电子器件已经发生了巨大的变化。ABC 和 ENIAC 计算机使用的是电子管。在 20 世纪 50 年代后期,大型的晶体管是构成计算机的主要逻辑单元。在 20 世纪 60 年代中期,集成电路出现。集成电路实际上包含了大量非常小的晶体管、电阻器、二极管和电容器。当今,硅芯片上的集成电路构成了大多数计算机的核心。材料和制造工业的技术发展使电子器件的体积更小,而速度则更快。现在,一个硬币大小的微处理器芯片的处理能力可以顶得上一个堆满 20 世纪 50 年代计算机设备的房间所能实现的处理能力。

此外,人们与计算机通信的方式也取得了很大进展。手工移动线路、在插件板上打孔以及敲击键盘是早期使用的一些方法,而鼠标、手写笔和语音系统都是最新的发展成果。

与计算机发展的早期不同,现在存在许多不同种类的计算机。尽管这些类别的界限并不总是很清晰,但当前使用的一些计算机仍然可以分为以下几类:

- 超级计算机——超级计算机是所有计算机中体积最大且运行最快的计算机,它主要用于大规模的科学和军事计算。与老式计算机相似的是,它们占据一个或多个房间。考虑到超级计算机昂贵的成本,很多大学没有自己单独的超级计算机,而是许多大学共同使用一台可能放在另一个州或城市的超级计算机。在您进行高级研究时可能会用到超级计算机。
- 大型计算机——大型计算机的体积和速度都比超级计算机小。大多数大学都拥有一台或多台大型计算机。这些计算机通常只占用房间的一部分,并且从校园的其他位置可以远程访问它们。
- 工作站——工作站比较小,可以在桌面上或者紧挨着工作台进行装配。就价格而言,它们适合于小型和中等规模的企业,但是它们的功能比个人计算机更为强大。
- 个人计算机——它们可以安装在桌面上,并且价格便宜,每个人(甚至学生)都可以拥有一台个人计算机。它们也称为微型计算机。

- 膝上型电脑——它们可以放置在公文包内，并且处理能力接近于个人计算机。
- 掌上型电脑——它们的体积很小，小到可以放到口袋内，但是它们的处理能力低于个人计算机和膝上型电脑。

令人惊奇的是，当前膝上型电脑的处理能力强于非常大型的 ENIAC 计算机。

总之，经过多年的不断发展，计算机已经变得更小、更快速、更为便宜和易于使用。从最初为科学计算和军事应用而开发的计算机，当今的计算机已经发展到用于许多人们生活的每个方面。

1.2 体系结构

无论是大型计算机还是掌上型电脑，典型计算机的体系结构通常包括一个中央处理器 (CPU)、主存储器、控制器和外围设备，图 1-2 给出了该结构的示意图。计算机的能力主要由 CPU 和主存储器决定，实际上当人们谈起计算机时，主要指的就是以上两个部件。

术语“硬件”指的是集成在计算机内部或与计算机相连接的电子和机械设备。按照这个定义，1.2 节中所提到的所有设备都可被看作硬件。

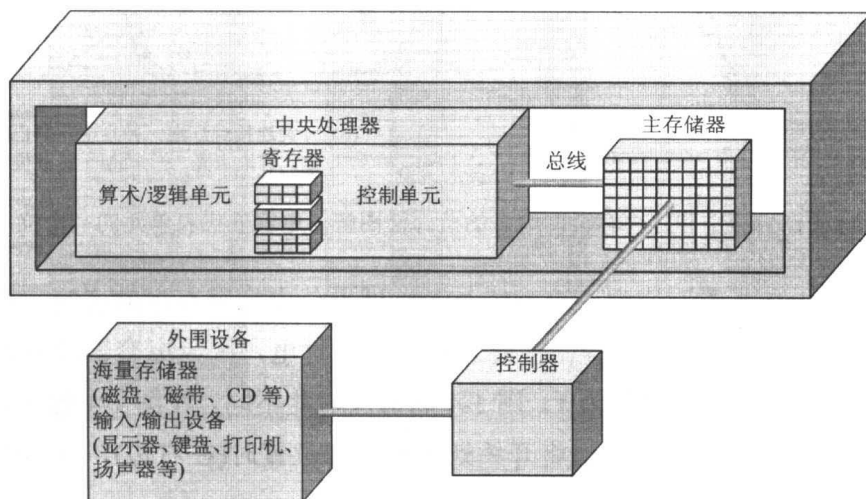


图 1-2 典型计算机的体系结构

1.2.1 主存储器

主存储器存储将被处理的信息或存储如何处理信息的指令。信息按照位的形式(即二进制数字的简化形式)进行存储。每个位可以处于两种状态中的一种。这两种状态可以通过多种形式来描述：

开或关

带电的或放电的

1 或 0

真或假

置位或清零

高或低