



Problem Solving and Program Design in C, 4E

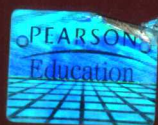
问题求解与程序设计 C语言版

(第4版)



Jeri R. Hanly 著
Elliot B. Koffman

朱剑平 译



清华大学出版社

问题求解与程序设计 C 语言版

(第 4 版)

Jeri R. Hanly 著

Elliot B. Koffman

朱剑平 译

清华大学出版社

北 京

Simplified Chinese edition copyright © 2006 by PEARSON EDUCATION ASIA LIMITED and TSINGHUA UNIVERSITY PRESS.

Original English language title from Proprietor's edition of the Work.

Original English language title: Problem Solving and Program Design in C, 4E by Jeri R. Hanly & Elliot B. Koffman, Copyright © 2004

EISBN: 0-321-19803-4

All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison-Wesley.

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macao).

本书中文简体翻译版由 Addison-Wesley 授权给清华大学出版社在中国境内(不包括中国香港、澳门特别行政区)出版发行。

北京市版权局著作权合同登记号 图字: 01-2005-5288 号

版权所有, 侵权必究。侵权举报电话: 010-62782989 13501256678 13801310933

本书封面贴有 Pearson Education(培生教育出版集团)激光防伪标签, 无标签者不得销售。

图书在版编目(CIP)数据

问题求解与程序设计 C 语言版(第 4 版) / (美) 亨利(Hanly, J. R.), (美) 考夫曼(Koffman, E. B.) 著; 朱剑平译. —北京: 清华大学出版社, 2007.1

书名原文: Problem Solving and Program Design in C, 4E

ISBN 978-7-302-14295-9

I. 问… II. ①亨… ②考… ③朱… III. C 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 152714 号

责任编辑: 常晓波

责任校对: 张 剑

责任印制: 孟凡玉

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

c-service@tup.tsinghua.edu.cn

社 总 机: 010-62770175

邮购热线: 010-62786544

投稿咨询: 010-62772015

客户服务: 010-62776969

印 刷 者: 北京鑫海金澳胶印有限公司

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: 40.5 字数: 1085 千字

版 次: 2007 年 1 月第 1 版 印次: 2007 年 1 月第 1 次印刷

印 数: 1~3000

定 价: 76.00 元

本书如存在文字不清、漏印、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:

010-62770177 转 3103 产品编号: 020252-01

谨以本书献给我们的家人

Jeri Hanly 的家人

Brian、Kevin 和 Trinity

Eric、Jennifer 和 Mical

Elliot Koffman 的家人

Caryn 和 Debbie

Robin、Jeff 和 Jonathan

Richard、Jacquie 和 Dustin

这本教材介绍的是一种很规则的方法，用于解决问题以及将广泛接受的软件工程方法作为内聚的、可读的、可重用的模块来应用到程序解决方案的设计中。本书将用 ANSI C 子集作为实现工具，ANSI C 由于其强大的功能和可移植性而成为业界的标准编程语言。本书也可以作为编程方法入门教材，因为本书假定读者事先并不具有计算机或者编程方面的知识。本书包含了很多案例研究和练习题，可作为计算机工程专业新生或者其他学科学生的 C 语言编程的教材。

这一版本中的新内容

在这一版的准备过程中，我们增加了一个计算机术语的词汇表，其中包含了对类型强转（type casting）和字符串扩展等内容的详细说明。第 5 章中的新案例研究探索了太阳热能领域中一些最新的研究成果。通过本书的学习，同学们将利用数目众多的全新项目来实践所学到的编程技能。另外，这一新版本中的附录还包含了最流行的开发环境之一即 Borland C++ Builder。本书的附录 D 还进一步探讨了指针的一些附加的算法。最后，本书的创作团队为本书定制了全新的样式，为范例代码都加上了行号，以便于同学们参考引用。

用 C 语言讲授程序开发

我们的两个目的——讲授程序设计和讲授 C 语言——看上去似乎有某种一致性。C 是大家公认的最难的编程语言，只有用其他更友好的语言学习了编程的基础知识之后，才能开始学习 C 语言。C 语言特别难学的认识可以追溯到其刚刚诞生的时候。由于 C 语言是为 UNIX 操作系统设计的编程工具，因此其最初的用户群是那些能够理解操作系统的复杂性和底层机制，并且能够很自然地将这些知识应用到所编写的程序中的程序员。所以，很多以讲授 C 语言为主要目标的教科书都要求学生事先已经掌握了学习这门介绍性编程语言时原本并不需要了解的关于机器语言方面的知识。

本书讲授了编程开发的关系方法以及 ANSI C 的内容，但我们的重点放在编程开发的关系方法上。有人可能会担心这种选择会导致本书在内容上对 ANSI C 的忽视。其实相反，本书将编程概念与用 C 语言来实现这些概念有机地融合在一起，重点展示了 ANSI C 作为一种高级编程语言的强大功能，而这一点正是许多以泛泛介绍 ANSI C 的教科书所忽略掉的。即使是遵循着这种先介绍 C 语言特性再介绍程序设计过程的教学方法，本书还是很全面地涵盖了 C 语言最本质的结构。

本书内容的组织结构与“指针”

本书介绍C语言各个主题的顺序是根据初级编程学生的学习需要来进行的,而不是根据C编程语言的结构来进行的。同学们可能会很惊讶于为什么本书没有名为“指针”的一章。之所以遗漏了这一章主题,是因为本书将C作为高级语言来对待,而不是没有意识到指针在C中所扮演的关键性角色。

与其他高级语言对输出参数和数组分别采取单独的语言结构不同的是,C语言将这些概念全包括在指针概念中,因此急剧地增加了学习C语言的复杂度。本书通过不同角度来讨论指针,从而简化了指针的学习过程,使同学们能够每次掌握一点关于指针的内容来循序渐进地学习。这样,本书就可以使用传统的高级语言的命名方式——输出参数、数组、数组下标、字符串——来展示基本概念,并且使得同学们不需要事先具备汇编语言的背景知识,就能掌握指针使用的各个方面。

因此,本书并不是没有专门介绍指针,而是用了整整4章的篇幅来分别介绍指针。第6章介绍了指针在简单输出和输入/输出参数方面的使用方法,第8章介绍了指针在数组中的使用方法,第9章介绍了指针在字符串和数组中的使用方法,第14章在回顾了前面介绍过的指针使用方法后讨论了动态内存分配。另外,第2章和第12章也讨论了文件指针。

软件工程概念

本书展示了软件工程的许多方面。某些内容是直接介绍的,还有一些内容则是通过范例来讲授的。良好的问题解决技能和高效软件开发之间的联系早在第1章中就确立了,第1章中专门用一节的篇幅来讨论问题解决的方法论。第1章中所讲述的软件开发现场阶段用于解决第一个案例研究,也将应用于本书后续的所有案例研究。主要的程序风格将用特殊格式突出显示出来,而范例中所使用的编码风格是基于C软件产业的标准。在好几章中都有一些小节讨论算法的跟踪、程序的调试和测试。

第3章介绍了通过选定的C库函数来进行过程抽象,以及无参数的空函数和接收输入参数并返回一个值的函数。第4章和第5章包含了一些附加的函数范例,第6章结束了具有简单参数的函数的学习。这一章将通过讨论指针的使用来展示输出参数和输入/输出参数。第7章介绍了将函数作为参数的使用方法。

第6章、第8章和第11章中的案例研究和范例程序介绍了数据抽象的概念,以及数据类型与运算符的封装的概念。第13章介绍了C语言在过程形式化方面的简便性,以及在由独立的头文件和实现文件定义的个人库中的数据抽象。第15章介绍了面向对象设计的概念,就像C++语言所实现的那样。

可视化函数接口的使用贯穿全书。直到第13章前,本书都没有使用全局变量,在全局变量的危险性和价值之间,我们需要谨言慎行。

本书在教学方面的特性

为了使本书能够成为一种更有用的教学工具,我们在本书中包含了以下教学方面的特性:

每个小节末尾的练习题 大多数小节的末尾都包含了一些自测题。其中包括需要分析程序代码段的习题,以及比较短的编程练习。

范例和案例研究 本书包含了大量的编程范例。我们尽可能地包含完整的程序或函数,而不是不完整的程序段。每一章都包含一个或多个案例研究,可用该章中所学的软件开发方法来

解决。许多案例研究使同学们可以对一些重要的应用有一个大概的认识，比如数据库搜索、支付与销售分析等商务应用、字处理，以及辐射等级监控和水资源保护等环境方面的应用。

语法显示框 本书用特殊的语法显示框来描述新的 C 特性的语法规义，并提供范例。

程序风格显示 程序风格显示框用于向同学们展示好的编程风格。

错误讨论和章节回顾 每一章都包含有一个小节来讨论常见的编程错误。章节回顾用表格形式展示了新遇到的 C 的语言结构。

每章末尾的练习题 每一章的章节回顾之后，都包含了带有答案的快速自测题。另外还包含有一些复习题。

每章末尾的编程练习 每一章末尾都会有一些编程练习。

附录

附录 F 介绍了如何使用最流行的开发环境之一即 Borland C++ Builder，此外这个附录还介绍了如何使用可从 Borland 网站 (www.Borland.com) 免费下载的命令行解释器。关于 ANSI C 结构的一个表包含在本书的最后，而附录 A 则包含了字符集的表格。由于本书内容只涉及 ANSI C 的一个子集，因此剩余的附录将扮演尤其重要的角色，这些章节能大大提高本书的参考价值。附录 B 是一个按字母顺序排列的 ANSI C 标准库的表格。附录 C 的表格展示的是所有 ANSI C 运算符的优先级和结合规则，以及在本附录中没有解释过的运算符。在本书中，数组的引用是通过下标来完成的，附录 D 只介绍了指针的算法。附录 E 列出了 ANSI C 的所有保留字。

教辅材料

本书的源代码和勘误表可以从 www.aw.com/cssupport 下载。

需要以下教辅材料的教师，请与 longqm@tup.tsinghua.edu.cn 联系。

- 包含答案的教师手册：包含了每一章的提要以及建议，还包含了自测题、复习题和部分编程练习的答案。
- 所有 PowerPoint 文件。
- 源代码。

致谢

许多人都参与了本书编写工作，在此我们要感谢来自加州理工学院喷气推进实验室的 Joan C. Horvath，感谢他提供了一些编程练习。我们要特别感谢霍华德大学学生 Paul Onakoya 的贡献，他为本书编写了新的词汇表，核查了习题答案，并为许多新的编程练习提供了教师手册答案。我们还要感谢坦普尔大学和怀俄明州大学前几届学生的协助，他们帮助我们检查了编程范例，并为大多数习题提供了答案，这些学生包括 Mark Thoney、Lynne Doherty、Andrew Wrobel、Steve Babiak、Donna Chrupcala、Masoud Kermani 和 Thayne Routh。

我们很高兴与 Addison-Wesley 团队合作。责任编辑 Susan Hartman Sullivan 和 Michael Hirsch 以及助理编辑 Galia Shokry 在我们编写本书的过程中给予了很多的指导。Juliet Silveri 监督了本书的编写工作，Nathan Schultz 和 Lesly Hershman 为本书的市场宣传做了很多工作。

J.R.H.

E.B.K.

第 1 章 计算机与编程概述	1
1.1 电子计算机的昨日今朝	2
1.2 计算机硬件	4
1.3 计算机软件	10
1.4 软件开发方法	15
1.5 软件开发方法的应用	18
案例研究：将英里转换为公里	18
本章小结	21
第 2 章 C 语言概述	23
2.1 C 语言的元素	24
2.2 变量声明和数据类型	29
2.3 可执行语句	32
2.4 C 程序的一般格式	39
2.5 算术表达式	41
案例研究：硬币集合求值	48
2.6 程序输出中数字的格式	52
2.7 交互模式，批处理模式和数据文件	54
2.8 常见编程错误	58
本章小结	63
第 3 章 自顶向下的函数设计	69
3.1 从现有信息中构建程序	70
案例研究：求圆的面积与圆周	71
案例研究：计算一批平垫圈的重量	73
3.2 库函数	77
3.3 自顶向下的设计与结构图	82
案例研究：绘制简单图	82
3.4 无参量函数	83
3.5 带有输入参量的函数	90
3.6 常见编程错误	98
本章小结	98
第 4 章 选择结构：If 语句和 Switch 语句	105
4.1 控制结构	106

4.2	条件	106
4.3	if 语句	114
4.4	使用复合语句的 if 语句	118
4.5	算法中的决策步骤	121
	案例研究: 水费问题	121
4.6	更多问题解决方案	129
	案例研究: 有节约要求的水费问题	130
4.7	嵌套的 if 语句和多选决策	132
4.8	switch 语句	139
4.9	常见编程错误	143
	本章小结	144
第 5 章	循环与循环语句	153
5.1	程序中的循环	154
5.2	计数循环和 while 语句	155
5.3	在循环中求和或求积	158
5.4	for 语句	162
5.5	条件循环	169
5.6	循环设计	173
5.7	嵌套循环	179
5.8	do-while 语句和标志控制的循环	182
5.9	问题求解示例	185
	案例研究: 太阳能式房屋的采光面积	186
5.10	如何调试和测试程序	191
5.11	常见编程错误	193
	本章小结	195
第 6 章	模块化编程	205
6.1	包含简单输出参数的函数	206
6.2	用输入/输出参数多次调用一个函数	212
6.3	名称的作用域	216
6.4	作为实参的输出形参	218
6.5	包含多个函数的程序	221
	案例研究: 简分数运算	221
6.6	程序的调试与测试	229
6.7	常见编程错误	231
	本章小结	231
第 7 章	简单数据类型	241
7.1	数字类型的表示与转换	242
7.2	字符类型的表示与转换	247
7.3	枚举类型	249

7.4 迭代近似法	253
案例研究：求根的二分法	255
7.5 常见编程错误	261
本章小结	261
第 8 章 数组	270
8.1 数组的声明与引用	271
8.2 数组下标	274
8.3 使用 for 循环顺序访问	275
8.4 使用数组元素作为函数参量	279
8.5 数组参量	281
8.6 数组的搜索与排序	291
8.7 多维数组	295
8.8 数组处理	299
案例研究：销售数据分析	299
8.9 常见编程错误	306
本章小结	307
第 9 章 字符串	317
9.1 字符串基础	318
9.2 字符串库函数：赋值与子串	322
9.3 更长的字符串：串联和整行输入	329
9.4 字符串比较	333
9.5 指针数组	336
9.6 字符操作	341
9.7 字符串与数字的相互转换	344
9.8 字符串处理示例	350
案例研究：文本编辑器	350
9.9 常见编程错误	357
本章小结	358
第 10 章 递归	367
10.1 递归的特性	368
10.2 跟踪递归函数	371
10.3 递归数学函数	378
10.4 使用数组和字符串参数的递归函数	382
案例研究：找出字符串中的大写字母	382
案例研究：递归选择排序	385
10.5 用递归解决问题	388
案例研究：集合操作	388
10.6 递归的经典案例：汉诺塔	395
10.7 常见编程错误	399

本章小结	400
第 11 章 结构与联合类型	405
11.1 用户定义的结构类型	406
11.2 结构类型数据作为输入和输出参数	410
11.3 结果值为结构的函数	414
11.4 结构类型问题求解	417
案例研究: 用于处理复数的用户定义类型	417
11.5 平行数组和结构的数组	423
案例研究: 通用度量转换	425
11.6 联合类型(选修)	432
11.7 常见编程错误	436
本章小结	437
第 12 章 文本与二进制文件处理	446
12.1 输入/输出文件: 过去与未来	447
12.2 二进制文件	454
12.3 搜索数据库	459
案例研究: 数据库查询	459
12.4 常见编程错误	467
本章小结	468
第 13 章 编写大型程序	474
13.1 使用抽象管理复杂性	475
13.2 个人库: 头文件	477
13.3 个人库: 实现文件	480
13.4 存储类别	483
13.5 为包含进库中修改函数	487
13.6 条件编译	489
13.7 main 函数的参量	492
13.8 定义包含参数的宏	494
13.9 常见编程错误	498
本章小结	498
第 14 章 动态数据结构	505
14.1 指针	506
14.2 动态内存分配	509
14.3 链表	513
14.4 链表运算符	518
14.5 用链表表示堆栈	522
14.6 用链表表示队列	525
14.7 有序列表	531

案例研究：维护整数有序列表	531
14.8 二叉树	540
14.9 常见编程错误	548
本章小结	548
第 15 章 C++入门	555
15.1 C++控制结构、输入/输出和函数	556
15.2 C++面向对象编程	560
本章小结	570
附录 A	575
附录 B ANSI C 标准库*	577
附录 C C 的运算符	590
附录 D 指针进阶	595
附录 E ANSI C 保留字	599
附录 F 使用 C++ BUILDER 和 BORLAND C++编译器	600
本书练习答案	606
词汇表	629

计算机与编程概述

1.1 电子计算机的昨日今朝

1.2 计算机硬件

1.3 计算机软件

1.4 软件开发方法

1.5 软件开发方法的应用

案例研究：将英里转换为公里

本章小结

在科技发达地区, 21 世纪的生活算得上是包围在名副其实的计算机的海洋中。从清晨自动开启的咖啡壶, 到用来做早餐的微波炉, 到驾驶着去上班的汽车, 再到自动提款机, 我们生活的方方面面都要依靠计算机^[1]。这些机器接收、存储、处理和输出信息, 可应付各种类型的数据: 数字、文本、图像、声音, 等等。

在这种技术中, 计算机程序才是最根本的, 如果没有一系列的指令, 计算机就成了废铁一堆。编程语言使我们可以编写程序来与计算机进行交流。

下面我们选择现今最通用的编程语言——C 语言——来开始计算机科学的学习。本章将介绍计算机及其组件, 以及编程语言的主要类别。本章会讨论计算机如何处理 C 语言程序。另外本章还会介绍一种用来解决编程问题的系统方法, 称为软件开发方法, 并要向同学们讲授如何应用这种系统方法。

1.1 电子计算机的昨日今朝

在我们的日常生活中会频繁地与计算机打交道, 我们有的人使用计算机进行文字处理, 或者在高校中使用计算机来学习编程。但是事情并不一直是这样, 就在不久之前, 大多数人还都认为计算机是一种很神秘的设备, 只有少数计算机精灵级的专家才明了其中的秘密。

第一台电子计算机诞生于 20 世纪 30 年代, 是由爱荷华州立大学的 John Atanasoff 博士和 Clifford Berry 博士构建的。Atanasoff 设计计算机的目的是帮助毕业生完成核物理课程中的数学计算。

第一台庞大的、用途简单的电子数字计算机的名称是 ENIAC, 是美国军方于 1946 年在宾夕法尼亚大学构建的。这台计算机重约 30 吨, 占地 30×50 英尺见方, 主要用于计算机弹道数据、预报天气以及原子能计算。

这些早期的计算机使用真空管作为基本的电子单元。电子元器件的设计水平与制造工艺的进步导致了新一代计算机的诞生, 它们比早期的计算机更小、更快、更便宜。

由于使用了先进的技术, 计算机处理器的整个电路都可以封装进一片相当于邮戳大小的电子元器件中, 我们可以称之为计算机芯片^[2]或微处理器芯片(如图 1.1 所示)。由于便宜的价格和

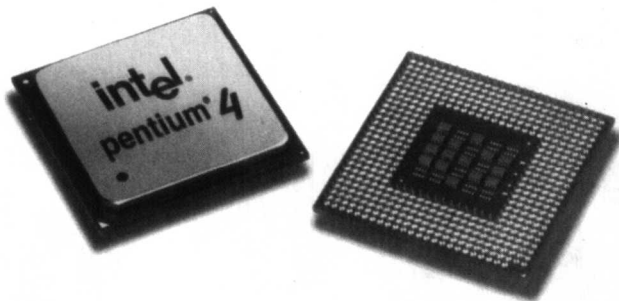


图 1.1 Intel 公司的 Pentium 4 处理器芯片是一种包含了中央处理单元所有电路的集成电路。这款处理器一秒种可以处理 60 亿条指令 (Intel 公司授权转载, © Intel Corporation 2003)

[1] 计算机——一种可接收、存储、传输和输出各种数据的机器。

[2] 计算机芯片(微处理器芯片)——一种包含了计算机处理器电路的硅芯片。

小巧的体积,微处理器芯片除了应用到计算机中之外,还被装进了电视、便携计算器、照相机、家用电器、汽车中。

如今,在办公室和家庭中,一般都有个人计算机(personal computer, PC),这些计算机的价格一般都在1000美元以下且可以放置在写字台上,而其性能却相当于20年前一台价值100000美元、占地100平方英尺的大型计算机。更小的计算机甚至可以放进公文包里(如图1.2(a)所示)或者拿在手中(如图1.2(b)所示)。

现代计算机是按照体积和性能来分类的。个人计算机(如图1.2(c)所示)同时只能供一个人使用。大型的实时事务处理系统(如ATM和其他银行网络以及旅馆、航空和汽车出租的预定系统)使用的是更强大、更稳定的大型机系统。容量最大、速度最快的大型机系统称做超级计算机,一般用于实验室研究,或者用于计算像天气预报之类的计算密集型应用程序。

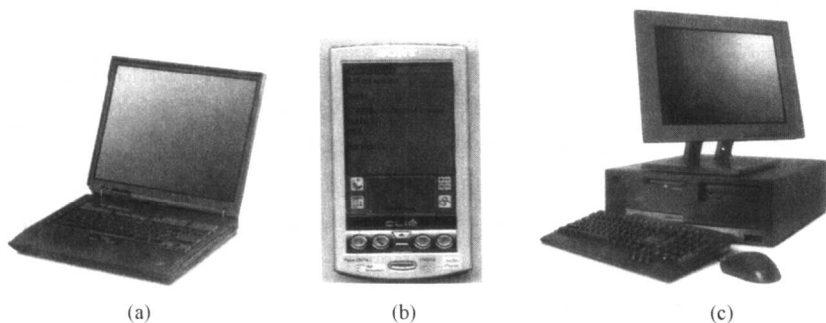


图 1.2 (a) 笔记本计算机(ThinkPad®, IBM 公司授权转载); (b) 掌上电脑(Sony Clié PDA®, Sony 公司授权转载); (c) 台式计算机(IBM NetVista Desktop, IBM 公司授权转载)

计算机系统的组成元素可分为两大类:硬件和软件。硬件^[1]是用于执行所需计算的设备,一般包含中央处理单元(central processing unit, CPU)、显示器、键盘、鼠标、打印机和扬声器。软件^[2]是一些程序^[3],能够通过一系列指令的执行来让我们解决问题。

编写计算机程序的方式近年来也有了巨大的进步。最初,编写计算机程序十分艰难,要求程序员用长长的二进制数^[4](即0和1的序列)来编写程序指令。现在,如C语言等高级编程语言使得程序的编写变得很容易。



第 1.1 小节 练习题

自测题

1. 计算机程序是一个硬件还是软件?
2. 大型机一般用于执行什么应用程序?

[1] 硬件——实际的计算机设备。

[2] 软件——与计算机相关联的程序。

[3] 程序——一系列可使计算机执行特定任务的指令。

[4] 二进制数——由0和1组成的数。

1.2 计算机硬件

尽管在成本、大小和容量上有着巨大的差异,但是现代的计算机存在着一些基本的共同点。本质上说,大多数计算机都包含着下列组件:

- 主存储器(内存)
- 辅助存储器,包括诸如硬盘、软盘、zip 磁盘、CD 和 DVD 等存储设备
- 中央处理单元
- 输入设备,如键盘、鼠标、手写板、扫描仪
- 输出设备,如显示器、打印机和扬声器

图 1.3 显示了这些组件在计算机中如何交互,箭头的指向表示的是信息的流向。程序在执行前必须先由辅助存储器中传输到主存储器中。使用程序的人(即程序的用户)必须提供一些数据来进行处理。这些数据通过输入设备来提供,并存储在计算机的主存储器中,随后中央处理单元(CPU)就可以访问并处理这些数据。处理的结果会返回到主存储器中。最后,主存储器中的信息将通过输出设备显示出来。本小节余下的篇幅将更详细地介绍这些组件。

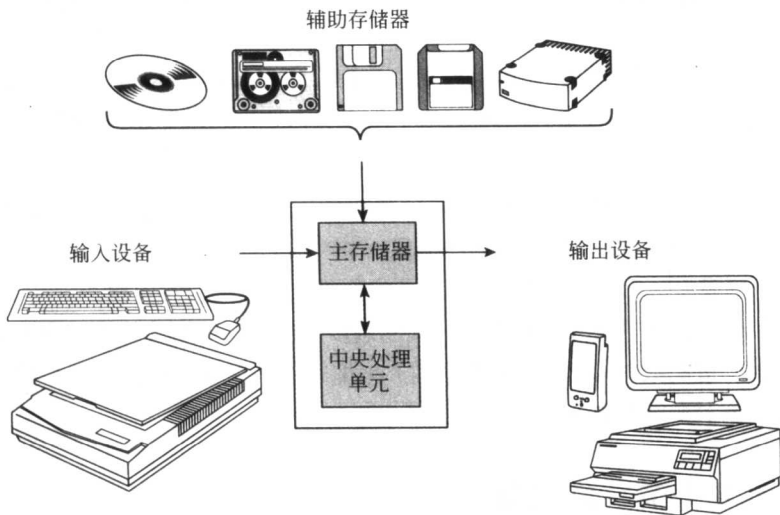


图 1.3 计算机的组件

1.2.1 内存

内存是所有计算机的必需组件。下面让我们来研究一下其组成结构以及计算机使用内存的方式。

内存剖析 我们可以将计算机的内存想象成存储位置(称做**内存单元**^[1])的有序序列(如图 1.4 所示)。为了存储和访问信息,计算机必须可以用某种方式来确认某个内存单元。因为

[1] 内存单元——内存中的一个存储位置。

每个内存单元都有一个唯一的地址^[1]来表示其在内存中的相对位置。图 1.4 展示的是一个包含了 1000 个内存单元的计算机内存，这些内存单元的地址是 0~999。然而，大多数计算机都包含数以百万计的内存单元，每一个单元都有自己的地址。

存储在内存单元中的数据称做内存单元的内容^[2]。每个内存单元通常都包含着内容，尽管我们可能并不需要调用它们。在图 1.4 中，内存单元 3 的内容是数字 -26，而内存单元 4 的内容是字母 H。

尽管没有在图 1.4 中展示出来，但内存单元还可以包含一条程序指令。存储程序和数据的能力被称做存储程序概念^[3]：程序的指令在执行前必须存储在主存储器中。我们可以通过在内存中存储一个不同的程序来改变计算机的操作。

字节与位 每个内存单元实际上都是由更小的称做字节^[4]的单元组成的。一个字节是存储一个字符所需要的存储空间，如在图 1.4 中内存单元 4 中所存储的字母 H。一个内存单元可以包含的字节数也随着计算机的不同而各不相同。每个字节都是由更小的称做位^[5]的单元组成的（如图 1.5 所示）。术语位（bit）源自单词二进制数位（binary digit），是计算机可以处理的最小元素。二进制是指一种基于数字 0 和 1 的计数系统，因此每一位都会是 0 或 1。一般说来，每 8 位组成一个字节。

内存	
地址	内容
0	-27.2
1	354
2	0.005
3	-26
4	H
⋮	⋮
998	X
999	75.62

图 1.4 主存储器中的 1000 个内存单元



图 1.5 字节与位的关系

内存中信息的存储与检索 内存中的任何值都是由某种模式的 0 和 1 来表示的。计算机可以存储一个值，也可以检索一个值。为了存储^[6]一个值，计算机要将选定的内存单元的每一位设置成 0 或 1，此过程将同时销毁该单元中以前存储的内容。要想从内存单元中检索^[7]一个值，计算机要将该单元内存储的 0 和 1 序列复制到另外一个存储区域中以备处理，并且这种复制操作不会销毁正在检索的那个内存单元中所存储的内容。无论信息的类型是字符、数字还是程序

[1] 内存单元的地址——计算机主存储器中某个内存单元的相对地址。

[2] 内存单元的内容——存储在内存单元中的信息，既可以是程序指令也可以是数据。

[3] 存储程序概念——计算机在主存储器中存储程序指令以备执行的能力。

[4] 字节——存储一个字符所需要的存储空间。

[5] 位——二进制数位，0 或者 1。

[6] 数据存储——将内存单元的每一位设置为 0 或 1，同时销毁该内存单元中以前保存的内容。

[7] 数据检索——将特定内存单元的内容复制到另外一个存储区域。