

中国计算机软件专业技术资格和水平考试用书（新大纲）

程序员考试辅导教程与同步 试题训练

王玉岗 黄义平 钟飞 编著

北 京

冶金工业出版社

2002

内容简介

本书是中国计算机软件专业技术资格和水平考试（程序员级）的辅导用书，根据 2002 年度程序员级考试大纲的要求编写。全书共分 10 章，内容包括：计算机硬件基础知识、程序语言与语言处理程序基础知识、操作系统基础知识、软件工程基础知识、数据库基础知识、网络基础知识、多媒体基础知识、数据结构基础、常用算法设计方法、C 语言基础等。

本书参照考试大纲的要求编写，内容丰富，注重基本概念，针对性强。每章后面备有例题分析和模拟试题的练习。本书可供参加中国计算机软件专业技术资格和水平考试（程序员级）的考生学习参考，同时也是大专院校相关专业师生和广大计算机工作者的实用参考书。

图书在版编目（CIP）数据

程序员考试辅导教程与同步试题训练 / 王玉岗等编著.
北京：冶金工业出版社，2002.6
ISBN 7-5024-3036-9

I. 程... II. 王... III. 程序设计 - 水平考试 - 试题
IV. TP311.1-44

中国版本图书馆 CIP 数据核字（2002）第 031632 号

出版人 曹胜利（北京沙滩嵩祝院北巷 39 号，邮编 100009）

责任编辑 程志宏

广东出版技校彩印厂印刷；冶金工业出版社发行；各地新华书店经销

2002 年 7 月第 1 版，2002 年 7 月第 1 次印刷

787mm×1092mm 1/16；30 印张；700 千字；472 页；1-1900 册

35.00 元

冶金工业出版社发行部 电话：（010）64044283 传真：（010）64027893

冶金书店 地址：北京东四西大街 46 号（100711） 电话：（010）65289081

（本社图书如有印装质量问题，本社发行部负责退换）

前 言

1. 关于本书

计算机科学是一门高速发展的学科。近十年来，计算机技术的发展更是一日千里，对于信息技术人才的需求也大大增加。

中国计算机软件专业技术资格和水平考试实施至今已有十多年。不仅成为信息技术人才培养的重要渠道，而且在社会上产生了很大影响，为我国软件产业的发展做出了重要贡献。为了适应新形势的要求，信息产业部中国计算机软件专业技术资格和水平考试中心于2001年组织和制订了新的考试大纲。为了配合中国计算机软件专业技术资格和水平考试中的程序员考试和推广计算机知识，我们编写了此书。

2. 本书特点

本书按照最新的考试大纲编写，介绍的知识全面，详略得当。在基础知识部分，注重基本概念，结合大纲的要求，突出重点地阐述相关的基本原理和基本方法；在程序设计部分以数据结构和基本算法为主线，并且详细地讲解了C语言的基本概念和程序设计方法。除此之外，在每章里还给出了大量的例题和模拟试题，让读者能巩固和检查所学的知识。

3. 结构安排

全书共分10章和2个附录，具体安排如下：

第1章介绍计算机硬件的基础知识。

第2章介绍程序语言与语言处理程序基础知识。

第3章介绍操作系统基础知识。

第4章介绍软件工程基础知识。

第5章介绍数据库基础知识。

第6章介绍网络基础知识。

第7章介绍多媒体基础知识。

第8章介绍数据结构基础。

第9章介绍常用算法设计方法。

第10章介绍C语言基础。

本书最后的附录A是2002年度程序员级考试大纲，附录B是按照最新考试大纲精选的一套模拟试题，可以帮助读者熟悉考试题型，巩固所学的知识。

4. 适用对象

本书可作为参加中国计算机软件专业技术资格和水平考试考生的辅导教程，也可作为大专院校相关专业师生和计算机工作者的参考书。

本书第1、2、9章由王玉岗编写；第3、4、5、10章由黄义平编写；第6、7、8章由钟飞编写。

本书在编写过程中，参考了许多相关的书籍和资料，编者在此对这些参考文献的作者表示感谢。

因水平有限，书中难免存在不足之处，望广大读者和专家给予指正，以利于改进和提高。

编 者
2002年5月

目 录

第 1 章 计算机硬件基础知识1	
1.1 计算机发展史.....1	
1.2 计算机系统的结构简介.....2	
1.3 计算机中数据的表示.....3	
1.3.1 数制及其转换.....3	
1.3.2 计算机中数值的存放方式.....7	
1.3.3 原码、补码、反码、移码.....9	
1.3.4 定点数与浮点数的机内表示.....11	
1.3.5 符号数据的表示方法.....14	
1.3.6 汉字的处理.....15	
1.4 数据的常用校验方法.....17	
1.4.1 奇偶校验码.....17	
1.4.2 海明码.....18	
1.4.3 循环冗余码.....20	
1.5 计算机运算基础.....22	
1.5.1 计算机二进制数的算术运算.....22	
1.5.2 逻辑代数及逻辑运算基础.....25	
1.6 计算机的体系结构和主要部件.....27	
1.6.1 中央处理器.....28	
1.6.2 存储器.....38	
1.6.3 输入/输出系统 (I/O 系统).....44	
1.6.4 总线结构.....50	
1.7 指令系统.....52	
1.7.1 指令格式.....52	
1.7.2 指令的分类和功能.....54	
1.7.3 常用的寻址方式.....57	
1.7.4 指令的执行过程.....58	
1.8 例题分析.....58	
1.9 模拟试题.....64	
第 2 章 程序语言与语言处理程序基础知识68	
2.1 程序语言基础.....68	
2.1.1 基本概念.....68	
2.1.2 程序语言数据类型.....71	
2.1.3 程序语言的控制结构.....77	
2.2 汇编程序基础知识.....80	
2.2.1 汇编语言概述.....80	
2.2.2 汇编程序工作过程.....81	
2.3 解释程序基础.....83	
2.3.1 高级语言实现方法.....83	
2.3.2 解释系统的结构.....84	
2.4 编译程序基础.....85	
2.5 例题分析.....87	
2.6 模拟试题.....91	
第 3 章 操作系统基础知识94	
3.1 操作系统简介.....94	
3.1.1 操作系统的定义.....94	
3.1.2 操作系统的功能.....95	
3.1.3 分析和设计操作系统的几种观点.....96	
3.1.4 操作系统的形成和发展.....97	
3.1.5 操作系统的结构.....100	
3.1.6 操作系统的硬件基础.....100	
3.2 进程管理.....101	
3.2.1 进程的概念.....101	
3.2.2 进程控制.....104	
3.2.3 进程间通信.....105	
3.2.4 进程的互斥与同步.....106	
3.2.5 进程死锁.....108	
3.2.6 进程调度.....109	
3.3 存储管理.....110	
3.3.1 存储管理简介.....110	
3.3.2 单用户系统存储管理.....111	
3.3.3 多用户系统的分区管理.....111	
3.3.4 分页存储管理.....113	
3.3.5 分段存储管理.....114	
3.3.6 段页式存储管理.....115	
3.3.7 存储保护.....115	
3.4 设备管理.....116	
3.4.1 设备管理简介.....116	
3.4.2 设备的分配.....118	
3.4.3 磁盘调度算法.....118	
3.5 文件管理.....119	
3.5.1 文件管理简介.....119	

3.5.2	文件的结构和组织	120	4.6.2	测试的步骤	161
3.5.3	文件的访问方法	121	4.6.3	白盒测试	162
3.5.4	文件存储设备管理	121	4.6.4	黑盒测试	162
3.5.5	文件控制块与文件目录	122	4.7	软件维护	162
3.5.6	文件的使用	122	4.7.1	软件维护的分类	162
3.6	作业管理	123	4.7.2	可维护性	163
3.6.1	作业管理简介	123	4.7.3	与软件维护有关的问题	163
3.6.2	用户作业管理	124	4.8	软件管理	163
3.6.3	作业调度	125	4.9	软件质量	165
3.6.4	用户接口	126	4.9.1	软件质量的特性	165
3.7	多处理器与线程	126	4.9.2	软件质量的保证	165
3.8	操作系统实例	127	4.10	面向对象的基本概念	166
3.8.1	UNIX 系统	127	4.10.1	对象	167
3.8.2	Windows NT 系统	130	4.10.2	类	167
3.9	例题分析	133	4.10.3	封装	167
3.10	模拟试题	139	4.10.4	继承	168
			4.10.5	消息	168
			4.10.6	多态性和动态绑定	168
第 4 章	软件工程基础知识	143	4.11	例题分析	169
4.1	软件工程简介	143	4.12	模拟试题	176
4.1.1	软件生存周期	143	第 5 章	数据库基础知识	182
4.1.2	软件开发模型	145	5.1	数据库的基本概念	182
4.2	系统分析和软件计划	146	5.1.1	有关数据库的几个概念	182
4.2.1	系统分析	146	5.1.2	数据库的产生和发展	182
4.2.2	可行性分析	146	5.2	数据模型	184
4.2.3	软件开发计划	146	5.2.1	数据模型的基本概念	185
4.3	需求分析	147	5.2.2	数据模式	185
4.3.1	需求分析的任务	147	5.2.3	基本的数据模型	186
4.3.2	结构化分析简介	148	5.3	数据库系统	188
4.4	软件设计	149	5.3.1	数据库系统的构成	188
4.4.1	概要设计	149	5.3.2	数据库系统的工作过程	190
4.4.2	结构化设计简介	153	5.4	关系模型的数学理论——关系代数	191
4.4.3	详细设计	154	5.4.1	关系模型的基本概念	191
4.5	编码	157	5.4.2	关系运算	191
4.5.1	程序编码注意事项	157	5.4.3	关系数据库的 DML 特点	195
4.5.2	程序设计方法	157	5.5	SQL 语言简介	196
4.5.3	源程序的质量指标	158	5.5.1	SQL 语言概述	196
4.5.4	编程风格	159	5.5.2	SQL 的数据定义功能 DDL	197
4.6	软件测试	160			
4.6.1	测试的基本概念	161			

5.5.3 SQL 的数据操纵功能 DML.....	199	6.6.3 C/S 的优缺点.....	247
5.5.4 SQL 的访问控制.....	203	6.7 Browser/Server 结构.....	248
5.5.5 嵌入式 SQL.....	204	6.7.1 B/S 结构的基本概念.....	248
5.6 数据库设计.....	204	6.7.2 B/S 结构与 C/S 结构的区别.....	248
5.6.1 数据库系统生存期.....	205	6.7.3 B/S 结构的优势.....	249
5.6.2 数据库的具体过程.....	205	6.8 网络安全性初步.....	249
5.7 例题分析.....	208	6.8.1 网络安全性的基本概念.....	249
5.8 模拟试题.....	218	6.8.2 网络信息安全初步.....	250
第 6 章 网络基础知识.....	221	6.8.3 防火墙的基本概念.....	251
6.1 计算机网络的基本概念与发展简介.....	221	6.9 Internet 与 Intranet 初步.....	252
6.1.1 基本概念.....	221	6.9.1 Internet 服务.....	252
6.1.2 发展简介.....	221	6.9.2 Internet 地址.....	255
6.2 计算机网络的应用与分类.....	222	6.9.3 Internet 上的主机域名.....	256
6.2.1 计算机网络的功能.....	222	6.9.4 Intranet 初步.....	257
6.2.2 计算机网络的分类.....	223	6.10 例题分析.....	258
6.2.3 计算机网络的组成.....	226	6.11 模拟试题.....	263
6.2.4 计算机网络的应用.....	227	第 7 章 多媒体基础知识.....	266
6.3 基本的网络协议与标准.....	229	7.1 多媒体基本概念与特征.....	266
6.3.1 网络的标准化.....	229	7.1.1 什么是多媒体.....	266
6.3.2 网络协议模型.....	230	7.1.2 多媒体计算机.....	266
6.3.3 OSI/RM 协议模型.....	230	7.1.3 多媒体的关键技术.....	267
6.3.4 TCP/IP 协议模型.....	231	7.1.4 多媒体系统的构成.....	268
6.3.5 OSI/RM 协议模型与 TCP/IP 模型的比较.....	232	7.2 图形与图像.....	270
6.4 常用网络操作系统.....	232	7.2.1 基本概念.....	270
6.4.1 网络操作系统概述.....	232	7.2.2 常见图形与图像的格式.....	272
6.4.2 UNIX.....	233	7.3 数字化音频.....	275
6.4.3 Microsoft Windows NT Server/Windows 2000 Server.....	234	7.3.1 基本概念.....	275
6.4.4 Novell Netware.....	234	7.3.2 音频信号的数字化.....	276
6.5 局域网与广域网.....	235	7.3.3 语音、语言合成和语音识别.....	277
6.5.1 常用的网络设备.....	235	7.3.4 音频文件格式.....	279
6.5.2 网络采用的主要传输媒体.....	237	7.4 视频与动画.....	280
6.5.3 局域网的协议标准和控制方式.....	239	7.4.1 基本概念.....	280
6.5.4 广域网技术.....	241	7.4.2 视频文件格式.....	281
6.6 Client/Server 结构.....	246	7.5 多媒体创作.....	284
6.6.1 C/S 的基本概念.....	246	7.5.1 多媒体创作工具.....	285
6.6.2 C/S 的体系结构与处理方式.....	246	7.5.2 多媒体的创作过程.....	286
		7.6 例题分析.....	287
		7.7 模拟试题.....	289

第 8 章 数据结构基础	291	8.9.3 链式存储线性表的查找	362
8.1 基本概念	291	8.9.4 散列表的查找	364
8.2 线性表	292	8.10 例题分析	366
8.2.1 线性表的基本运算	292	8.11 模拟试题	370
8.2.2 线性表的存储方式	293	第 9 章 常用算法设计方法	373
8.3 栈	298	9.1 算法概述	373
8.3.1 栈的基本概念	298	9.2 迭代法	373
8.3.2 栈的顺序存储结构	299	9.3 穷举搜索法	374
8.3.3 栈的链式存储结构	300	9.4 递推法	376
8.3.4 栈的应用实例	301	9.5 递归	377
8.4 队列	309	9.6 回溯法	381
8.4.1 队列的基本概念	309	9.7 贪婪法	385
8.4.2 队列的基本操作	310	9.8 分治法	388
8.4.3 队列的顺序存储结构	310	9.9 动态规划法	391
8.4.4 队列的链式存储结构	311	9.10 例题分析	393
8.5 数组	316	9.11 模拟试题	396
8.5.1 数组的基本概念	316	第 10 章 C 语言基础	401
8.5.2 数组的应用	316	10.1 数据类型与表达式	401
8.5.3 稀疏矩阵	320	10.1.1 数据类型	401
8.6 字符串	325	10.1.2 常量	402
8.7 树和二叉树	329	10.1.3 变量	404
8.7.1 树	329	10.1.4 变量的存储方式	413
8.7.2 二叉树	332	10.1.5 运算符	414
8.7.3 二叉排序树	336	10.1.6 表达式	420
8.7.4 二叉线索树	340	10.2 分支、循环、函数和递归	422
8.7.5 树应用实例	343	10.2.1 分支控制语句	422
8.8 排序	345	10.2.2 循环控制语句	424
8.8.1 排序的基本概念	345	10.3 输入输出和文件的处理	427
8.8.2 选择排序	346	10.3.1 数据的输入与输出	427
8.8.3 直接插入排序	347	10.3.2 文件的处理	431
8.8.4 冒泡排序	348	10.4 字符处理和计算求解	442
8.8.5 堆排序	351	10.4.1 字符处理	442
8.8.6 快速排序	353	10.4.2 常用算法举例	449
8.8.7 合并排序	354	10.5 例题分析	455
8.8.8 外部排序	356	10.6 模拟试题	456
8.9 查找	359	附录 A 2002 年度程序员级考试大纲	458
8.9.1 顺序存储线性表的查找	360	附录 B 模拟试题	461
8.9.2 分块查找	362		

第 1 章 计算机硬件基础知识

1.1 计算机发展史

计算机的发展历程，是从 1946 年 2 月现代电子计算机的鼻祖 ENIAC (Electronic Numerical Integrator and Computer) 的研制成功开始的。ENIAC 是计算机发展史上的一座纪念碑。自计算机发明以来，在 50 年中经历了巨大的发展变革。根据计算机系统所采用的元器件不同，计算机的发展历程可以划分为以下几个阶段。

1. 电子管计算机时代

从 1946 年到 50 年代后期 (1946—1957)，当时计算机的元器件主要由电子管 (vacuum tube) 组成。其特点是体积庞大、功耗高、运算速度较低。如 ENIAC 其总体积约 90 立方米，占地 170 平方米，重达 30 吨，功耗为 140 千瓦，有 18000 多个电子管，每秒钟能进行 5000 次加法计算。ENIAC 最初是为了进行弹道计算而设计的专用计算机。在这一阶段，计算机主要用于军事、国防等尖端技术领域。1945 年左右，冯·诺依曼等人在研制 EDVAC (Electronic Discrete Variable Computer) 时，提出了著名的存储程序 (Stored-Program) 概念。这一概念成为以后计算机发展的基石，被称为冯·诺依曼计算机。现代的计算机大多数是基于他的概念。1954 年 12 月 IBM 公司推出的 IBM 650 是第一代计算机的代表。

2. 晶体管计算机时代

从 20 世纪 50 年代后期到 60 年代中期 (1958—1964) 为晶体管计算机时代。电子管元件有许多明显的缺点。例如，在运行时产生的热量太多，可靠性较差，运算速度不快，价格昂贵，体积庞大，这些都使计算机发展受到限制。晶体管 (Transistor) 于 1947 年在贝尔实验室诞生后，体积小、功耗低、价格便宜的晶体管就马上取代了电子管。不仅提高了计算机的性能，也使计算机在科研、商业等领域内得到广泛地应用。第二代计算机除了采用了晶体管器件之外，在存储器方面也改用了速度更快的磁芯存储器；与此同时高级编程语言和系统软件的出现，也大大提高了计算机的性能和拓宽了其应用领域。这一时期计算机的代表主要有 DEC 公司 1957 年推出的 PDPI、IBM 公司于 1962 年推出的 7094 以及 CDC 公司 1964 年研制成功的 CDC6600。

3. 集成电路计算机时代

从 20 世纪 60 年代中期到 70 年代初期 (1965—1971) 为集成电路计算机时代。前两代的计算机均采用分离器件 (Discrete Component) 组成。第三代计算机采用了当时出现的集成电路 (Integrated Circuit)。与晶体管相比，集成电路的体积更小，功耗更低，而可靠性更高，造价更低廉，因此得到迅速发展。由于采用了集成电路，使得计算机的制造成本迅速下降；同时因为逻辑和存储器件集成化的封装，大大提高了运行速度，功耗也随之下降；集成电路的使用，使得计算机内各部分的互联更加简单和可靠，计算机的体积也进一步缩小。IBM 360 系统是最早使用集成电路元件的通用计算机系列，它开创了民用计算机使用集成电路的先例，计算机从此进入了集成电路时代。IBM 360 成为第三代计算机的里

程碑。

4. 大规模集成电路计算机时代

从 20 世纪 70 年代初期到 70 年代后期（1972—1978）为大规模集成电路（LSI）计算机时代。进入 20 世纪 60 年代后，微电子技术发展迅猛。20 世纪 70 年代初半导体存储器的出现，迅速取代了磁芯存储器，在 1967 年和 1977 年，分别出现了大规模集成电路和超大规模集成电路，并立即在电子计算机上得到了应用。接着就进入了超大规模集成电路（VLSI）计算机时代。美国 ILLIAC-IV 计算机，是第一台全面使用大规模集成电路作为逻辑元件和存储器的计算机，它标志着计算机的发展已到了第四代。

5. 智能电子计算机

第五代电子计算机是智能电子计算机，它是一种有知识、会学习、能推理的计算机，具有能理解自然语言，声音，文字和图像的能力，并且具有说话的能力，使人机能够用自然语言直接对话，它可以利用已有的和不断学习到的知识，进行思维、联想、推理、并得出结论，能解决复杂问题，具有汇集、记忆、检索有关知识的能力。智能计算机突破了传统的冯·诺依曼式机器的概念，舍弃了二进制结构，把许多处理机并联起来，并行处理信息，速度大大提高。它的智能化人机接口使人们不必编写程序，只需发出命令或提出要求，电脑就会完成推理和判断，并且给出解释。

6. 神经网络计算机

第六代电子计算机是模仿人的大脑判断能力和适应能力，并具有可并行处理多种数据功能的神经网络计算机。与以逻辑处理为主的第五代计算机不同，它本身可以判断对象的性质与状态，并能采取相应的行动，而且它可同时并行处理实时变化的大量数据，并引出结论。以往的信息处理系统只能处理条理清晰，经络分明的数据。而入的大脑却具有能处理支离破碎，含糊不清信息的灵活性，第六代电子计算机将类似人脑的智慧和灵活性。电子计算机的发展已经进入了第六代，这种发展可能仅仅是刚起步，前途没有止境。

随着技术的日新月异，软件和通信的重要性也逐步上升，成为和硬件一样举足轻重的因素。同时系统结构的特点对计算机的性能也有巨大的影响（中断系统、Cache 存储器、流水线技术等等）。

实际上在第三代计算机以后，就很难找到一个统一的标准进行划分。目前更流行的是认为计算机的发展经过了三次浪潮（wave）。第一次为单个主机（Mainframe）的时期，IBM 360、370 是其典型代表。第二次为客户机/服务器（Client/Server）的时期，这时期出现了小型机、微型机和局域网。现在处于第三次浪潮，网络计算的时期，即以网络为中心或以网络为基础的计算时期。

1.2 计算机系统的结构简介

计算机的工作过程一般是由用户（程序员）使用各种编程语言把所需要完成的任务以程序的形式提交给计算机，然后翻译成计算机能直接执行的机器语言程序，在计算机上运行以实现用户需要的功能。

计算机系统可以看作是一个多层的系统，不同的用户（或者应用）与不同的层次打交道。通常把计算机系统分作 4 层：物理机器（机器语言机器）、操作系统语言机器、汇编语

言机器和高级语言机器，如图 1-1 所示。

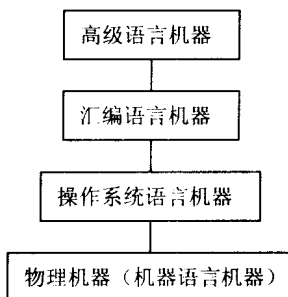


图 1-1 计算机系统的层次结构

第 1 层实际上就是由硬件组成的计算机本身，它所能理解和执行的是机器语言。第 2~4 层其实都是“虚拟”机器，通过软件组成的层次结构，隐藏了机器的硬件实现细节。用户在其上见到的“机器”实际上是在物理机器上运行的软件表示的虚拟机器。

1945 年，冯·诺依曼在设计 EDVAC 时提出了存储程序的概念。1946 年他和同事总结当时计算机的研究成果后提出了一种概念性的结构。这种结构也就称为冯·诺依曼结构。如图 1-2 所示。

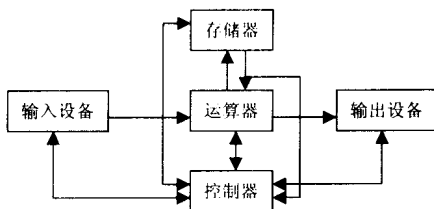


图 1-2 冯·诺依曼计算机结构

其特点是：以运算器 ALU (Arithmetic Logical Unit) 为中心，各部分联系由控制器 PCU (Primary Control Unit) 集中控制；采用存储程序原理，数据和程序事先存放主存 (main memory) 中，运行时顺序取出执行，指令和数据同等对待；主存按地址访问，是一个一维空间；数据以二进制编码等一系列特点。其中最根本是存储程序原理。这些结构和特点为现代计算机的发展奠定了基础，并起了很大作用。

1.3 计算机中数据的表示

1.3.1 数制及其转换

计算机的存储器 (memory) 和寄存器 (register) 都是两态器件，适合于存放二进制数据。所以各种信息包括数据、指令、地址、内容等都是以二进制的方式在计算机里存储和运算，这也是冯·诺依曼计算机的一个重要特点。尽管为了方便起见，也使用其他进制，如十进制、十六进制，但是在计算机内部数据还是以二进制的形式存放和处理。二进制只使用了两个不同的数字符号，易于用物理器件来实现：在物理世界中具有两个稳定状态的物理器件很多，如晶体管的“截止”与“导通”、电容的“充电”与“放电”、电压信号的

“高”与“低”、脉冲的“有”与“无”，电磁单元的“正向磁化”与“反向磁化”等等，只要规定其中一种稳定状态表示“1”，另一种稳定状态表示“0”，就可以用来表示二进制数位了。

其次，二进制的运算规则非常简单，易于用电子器件来实现。但是二进制书写起来太长，为了方便起见，有时也需采用不同的数制（number system）来表示数据，例如，十进制、十六进制、八进制等。

1. 数制介绍

所谓数制（number system）是指数的表示系统。如人们常用的十进制和钟表使用的六十进制，计算机的二进制等。

基数（base）是指一个数制包含的数字符号的个数。如十进制含有 10 个数码：0, 1, 2, 3, 4, 5, 6, 7, 8, 9。对于八进制来说就有 8 个数码，分别为 0~7；而十六进制的 16 个数码分别为 0~9 和 A~F。二进制的基数为 2，数码为 0 和 1。

位值（place value）也叫权（power）。一个数字由一串数字符号组成，每一位数字符号表示的数值除本身外，与所处的位置有关。与位置有关的值就叫位值或权。

数值按权展开。任意一个数用 r 进制可以表示为：

$$N = (R_n R_{n-1} \dots R_1 R_0 R_{-1} \dots R_{-m})_r$$

$$N_r = \sum_{i=-m}^n R_i \times r^i$$

上式称为数 N 在 R 进制中的按权（Power）展开式。任何一个数都是各位数字本身的值和与其权的积的总和。

【例 1-1】十进制数 12345.67 按权展开。

解：(12345.67)₁₀ = $1 \times 10^4 + 2 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0 + 6 \times 10^{-1} + 7 \times 10^{-2}$ （下标表示数制）

【例 1-2】八进制数 12345 按权展开。

解：(12345)₈ = $1 \times 8^4 + 2 \times 8^3 + 3 \times 8^2 + 4 \times 8^1 + 5 \times 8^0$ （下标表示数制）

【例 1-3】十六进制数 ABCDEF 按权展开。

解：(ABCDEF)₁₆ = $A \times 16^5 + B \times 16^4 + C \times 16^3 + D \times 16^2 + E \times 16^1 + F \times 16^0$ （下标表示数制）

表 1-1 列出了常用的十、十六、八和二进制数的对照表。

表 1-1 常用的十、十六、八和二进制数的对照表

十进制	十六进制	八进制	二进制	十进制	十六进制	八进制	二进制
1	1	1	1	9	9	11	1001
2	2	2	10	10	A	12	1010
3	3	3	11	11	B	13	1011
4	4	4	100	12	C	14	1100
5	5	5	101	13	D	15	1101
6	6	6	110	14	E	16	1110
7	7	7	111	15	F	17	1111
8	8	10	1000				

2. 数制间的转换

• 从十进制向 r 进制转换

要把一个十进制数转换为等价的 r 进制数，把整数部分和小数部分分开处理，分别转换为对应的 r 进制数。然后再把这两部分合起来就完成了。

1) 整数部分转换算法。

对于整数，可以用基数 r 不断去除被转换的十进制数，直至商为 0。每次相除所得的余数就是对应的各位数字。这种算法被称为基数辗转相除法或除 r 取余法。

【例 1-4】将 $(451)_{10}$ 转换为二进制数。

转换过程为：

K	商 (Q)	余数 (R_k)
	451	
1	225	1
2	112	1
3	56	0
4	28	0
5	14	0
6	7	0
7	3	1
8	1	1
9	0	1

$$(451)_{10} = (111100011)_2$$

【例 1-5】将 $(451)_{10}$ 转换为八进制数。

转换过程为：

K	商 (Q)	余数 (R_k)
	451	
1	56	3
2	7	0
3	0	7

$$(451)_{10} = (703)_8$$

【例 1-6】将 $(451)_{10}$ 转换为十六进制数。

转换过程为：

K	商 (Q)	余数 (R_k)
	451	
1	28	3
2	1	C
3	0	1

$$(451)_{10} = (1C3)_{16}$$

2) 小数部分转换算法。

对于小数，可以用基数 r 不断去乘被转换的十进制数小数。每次相乘所得的整数就是对应的各位数字。这种算法被称为基数辗转相乘法或乘 r 取整法。由此过程可知，并不是每个十进制数小数都可以经过有限次相乘而表示为 r 进制小数。一般可能只得到近似的结果。

【例 1-7】将 $(0.6789)_{10}$ 转换为二进制数。

转换过程为：

K 整数部分 (R_k) 小数部分 (F')

		0.6789
1	1	0.3578
2	0	0.7156
3	1	0.4312
4	0	0.8624
5	1	0.7248
6	1	0.4496

... .. (根据小数精度要求终止过程)

$$(0.6789)_{10} = (0.101011\dots)_2$$

【例 1-8】将 $(0.1875)_{10}$ 转换为八进制数。

转换过程为：

K 整数部分 (R_k) 小数部分 (F')

		0.1875
1	1	0.5
2	4	0

$$(0.1875)_{10} = (0.14)_8$$

【例 1-9】将 $(0.1875)_{10}$ 转换为十六进制数。

转换过程为：

K 整数部分 (R_k) 小数部分 (F')

		0.1875
1	3	0

$$(0.1875)_{10} = (0.3)_{16}$$

• 二进制数/八进制数/十六进制数之间的相互转换

因为 $2^3 = 8$, $2^4 = 16$, 所以 3 位二进制数对应着一个八进制数码, 4 位二进制数对应着一个十六进制数码, 如表 1-1 所示。因而二进制数/八进制数/十六进制数之间的相互转换存在一种很简单的方法, 就是: 从小数点开始, 分别向左、右每 3 位二进制数编成一组, 若不够 3 位, 则小数点左侧的最高位和右侧的最低位用 0 补充; 然后每一组用对应的八进制数码表示即可。八进制向二进制转换的方法: 从小数点开始, 把每一位八进制数码转换成对应的 3 位二进制数即可。其小数点左侧的最高位或右侧的最低位如果有 0 的话可以省去。二进制和十六进制数之间的转换与此类似, 只不过每组的二进制位数是 4 而不是 3。

【例 1-10】将 $(3456.7)_8$ 转换为二进制数。

转换过程为：

$$\begin{array}{cccccc} 3 & 4 & 5 & 6 & . & 7 \\ 011 & 100 & 101 & 110 & . & 111 \\ (3456.7)_8 = (11100101110.111)_2 \end{array}$$

【例 1-11】将 $(ABCD.EF)_{16}$ 转换为二进制数。

转换过程为：

$$\begin{array}{cccccc} A & B & C & D & . & E & F \\ 1010 & 1011 & 1100 & 1101 & . & 1110 & 1111 \\ (ABCD.EF)_{16} = (1010101111001101.1110\ 1111)_2 \end{array}$$

在计算机中八进制数、十六进制数通常被认为是二进制数的另一种表示方法。

- 其他一些特殊方法

1) 由于八进制与二进制之间的转换十分简单，因此有时十进制与二进制之间的转换可以以八进制为中介。

$$(451)_{10} = (703)_8 = (111000011)_2$$

2) 如果一个分数的分母为 2 的整数次幂，则可以用以下方法转换。

$$(7/16) = 7 \times 2^{-4} = (111)_2 \times (0.0001)_2 = (0.0111)_2$$

3) 从十进制向八进制/十六进制转换可以先把十进制转换为二进制，然后再从二进制转换为八进制/十六进制。

$$(564.1875) = (1000110100.0011)_2 = (1064.14)_8 = (234.3)_{16}$$

1.3.2 计算机中数值的存放方式

1. 数据种类

计算机指令处理的基本数据有 4 种：地址数据、数值数据、字符数据和逻辑数据。

- 地址数据

指令中的地址信息实际上也是一种数据，常常对其执行某种运算以求得新的操作数地址。从这种意义上看，地址数据其实就是无符号的整数。

- 数值数据

常用的数值数据有定点数（整数）、浮点数（实数）以及十进制数等。因为计算机中用于表示数值数据的位数是有限的，而且在用浮点法表示数据时会有精度损失，所以在计算机中进行运算时，不可避免地会遇上结果的溢出（上溢出或下溢出）或者舍入，需要正确地处理这些情况。

尽管计算机内部适用于二进制方式表示数据，但对于人而言还是习惯于用十进制进行运算。所以在用计算机处理问题时就需要进行两次转换：十进制输入，然后转变为二进制进行处理，最后再转变为十进制输出。在某些应用中，大量的处理在于数据的输入/输出，而相对来说计算只占一小部分且较简单，在这种情况下，更适合于用十进制数进行处理，以提高效率。

- 字符（串）数据

我们需要交给计算机处理的数据除了用以计算的数值数据外，还有用于文字信息处理

的字符数据。这些字符数据不能直接用于基于二进制的计算机中，因此必须经过某种形式的编码（如 ASCII 码、EBCDIC 码）用 H 进制位串来表示字符。

- 逻辑（布尔）数据

一般情况下，数据是以其存储单位 addressable unit（可寻址单位）作为一个整体来处理的，例如字节（byte）、半字（half word）、字（word）和双字（double word）等。但是在某些情况下，需要对数据的二进制位（bit）逐位进行处理——这些数据被看作为逻辑数据。

实际上在计算机内部，各种数据都是以二进制位的形式存放。具体的数据类型其实是由在数据上执行的操作或运算来决定的。

2. 数据的组织

计算机的存储器一般按字节进行组织，但其指令系统可能支持对半字、字或双字的运算。

为了便于硬件实现，一般要求多字节数据对准边界（alignment）。若数据位数不足时则以空白字节填充。这样可以减少对存储器的访问次数。多字节数据涉及的另一个问题则是其高、低位字节的排列顺序。在不同类型的计算机中其顺序也不同，有的是低位字节存放于低地址，而另一个方案则把高位字节放在低地址。

数值在计算机中存储的形式主要有：位、字节、字、双字、四字、十字节和字符串。

- 位

位（Bit）是由一位二进制数构成。在计算机中，位是最小的表示形式。每一位只能是 0 或 1，通常用于表示一种状态。

- 字节

字节（Byte）是由八位连续的二进制位构成。在计算机内部，字节是存储单元的基本单位。字节最低位称为第 0 位，最高位称为第 7 位。对于无符号整数来说，它可以存储的数值为 0~255 的任意整数。

- 字

字（Word）是由两个连续的字节构成，即由 16 个二进制位构成的。通常将低 8 位称为低有效字节，而高 8 位为高有效字节。

- 双字

双字（Dword）是由两个相邻的连续的字构成，即由 4 个字节构成的，宽度是 32 位。通常用于运算时提高精度。

- 四字

四字（Qword）是由 4 个相邻的连续的字构成，即由 8 个字节构成的，宽度是 64 位。用于运算时进一步提高精度。

- 十字节

十字节（Tbyte）是由 10 个连续的字节构成。用于存储字符串或非常大的数字。

- 字符串

字符串（String）又称为串。如“how are you”。字符串每一个字符存储中都占用一个字节的长度。

1.3.3 原码、补码、反码、移码

在数学中,正数和0都可以表示为无符号数,而负数通常是用“-”号来表示。但在计算机里,任何数值都用二进制表示。也就是说只能用“1”和“0”来表示的。计算机里习惯上用最高位(最左位)来表示数值的符号:用0表示正数,用1表示负数。例如 $(+0111010)_2$ 和 $(-0111010)_2$ 在计算机里分别表示为00111010和10111010。

通常一个数值数据的机内表示形式称为其机器码。一个机器码所代表的数值称为该机器码的真值。对于数值数据而言,无论以定点形式表示或是以浮点形式表示,都需要通过某种编码方法,以计算机所能处理的0和1二进制位来表示其数值大小。

实际应用中常见的编码有4种,分别是:原码、反码、补码和移码(又称增码)。下面假设机器码的宽度为n,最高位是符号位。

1. 原码

定点整数的原码表示法,定义如下所示:

$$[X]_{\text{原}} = \begin{cases} X & 0 \leq X < 2^{n-1} \\ 2^{n-1} - X & -2^{n-1} < X \leq 0 \end{cases}$$

定点小数的原码表示法,定义如下所示:

$$[X]_{\text{原}} = \begin{cases} X & 0 \leq X < 1 \\ 1 - X & -1 < X \leq 0 \end{cases}$$

• 原码的求法

1) 用定义求值。

【例 1-12】机器码为8位,求 $X = (-1010)_2$ 和 $X = (-0.1010)_2$ 的原码。

解:

$$X = (-1010)_2, \text{ 则 } [X]_{\text{原}} = 2^7 - (-1010)_2 = 10000000 + 1010 = 10001010$$

$$X = (-0.1010)_2, \text{ 则 } [X]_{\text{原}} = 1 - (-0.1010)_2 = 1.1010 = 11010000$$

2) 直接写出原码的值。

整数:先写出X的二进制表示,然后用0、1代表正负,放在最高有效位上(本例是第8位)。绝对值在最右端,中间补0。小数:绝对值在最左端,即最高有效位之后。其余同整数。

• 原码的特点

1) 原码表示法的取值范围 $-(2^{n-1}-1) \leq X \leq 2^{n-1}-1$ 。

2) 在原码表示法中,零有正零和负零之分。 $[+0]_{\text{原}} = 0000\dots 0$, $[-0]_{\text{原}} = 1000\dots 0$ 。因而n位二进制数可以表示为 $2^n - 1$ 个原码。

3) 一个数的原码实质为符号位加上数值本身的绝对值。用原码实现乘除运算规则较简单,但做加减运算不方便。原码表示法的一个主要优点在于其真值和编码表示之间对应关系很直观,容易转换。

2. 补码

整数的补码表示法,定义如下所示:

$$[X]_{\text{补}} = \begin{cases} X & 0 \leq X < 2^{n-1} \\ 2^n - |X| = 2^n + X & -2^{n-1} \leq X < 0 \end{cases}$$