



全国普通本科规划教材

世纪高等教育精品大系

(计算机类)

软件技术基础

本书编委会

浙江科学技术出版社
科学出版

浙江省高等教育重点教材

软件技术基础

本书编委会



全国普通本科规划教材
世纪高等教育精品大系
(计算机类)

浙江科学技术出版社
科学出版社

图书在版编目 (CIP) 数据

软件技术基础/《软件技术基础》编委会编.-杭州:
浙江科学技术出版社, 2004. 8

(世纪高等教育精品大系)

ISBN 7-5341-2402-6

I. 软... II. 软... III. 软件-高等学校-教材

IV. TP31

中国版本图书馆 CIP 数据核字 (2004) 第 072323 号

丛 书 名	世纪高等教育精品大系
书 名	软件技术基础
编 著	本书编委会
出版发行	浙江科学技术出版社 科 学 出 版 社
联系电话	(0571) 85152486
印 刷	杭州大众美术印刷厂
开 本	787×1092 1/16
印 张	24.25
字 数	619000
版 次	2004 年 8 月第 1 版
印 次	2004 年 8 月第 1 次印刷
书 号	ISBN 7-5341-2402-6
定 价	30.00 元
责任编辑	刘丽丽 张祝娟
封面设计	孙 菁

前言

随着计算机科学与技术的发展及普及,计算机已经在科学研究、工农业生产、社会服务、国防建设乃至人类社会的各个领域都显示出无比巨大的作用。现今,充分发挥计算机的潜能,使用这一现代化的工具发展科学技术、推动人类社会的进步,不仅要依靠计算机专业人员,更要依靠广大非计算机专业的人员,由他们在各个领域从事更广阔、更深层次的开发与应用。

在高等院校,目前,为非计算机专业普遍开设的计算机课程有两门,一门是“计算机文化基础”,一门是“高级语言程序设计”。计算机文化基础主要是解决操作使用计算机的问题,这是使用计算机必需具备的操作技能;高级语言程序设计主要是学会一种计算机语言,这是为利用计算机解决实际问题必须掌握的语言能力。但是,计算机应用是一门博大、精深的学科,光会操作计算机,仅掌握一两门高级语言是远远不够的,它涉及计算机的硬件与软件技术,特别是计算机软件技术,包括操作系统、语言处理系统、数据库系统、数据结构、算法设计、软件工程、网络软件系统、多媒体技术、图形图像技术、信息安全等。只有较好掌握计算机软件技术,才具有专业领域计算机应用系统的开发与维护的能力。由于非计算机专业用于开设计算机课程的课时数有限,不可能为上述各个软件技术分别设置为独立课程。因此,按照教育部提出的非计算机专业计算机基础课程三层次教学体系的要求,为非计算机专业的学生开设“软件技术基础”课程,就成为必然的选择与趋势。

“软件技术基础”教材的编写及其课程的教学,历来是一个没有很好解决的难题。究其原因主要有四个方面:(1)内容庞杂,课时有限;(2)学生已掌握的计算机语言不同,水平不一;(3)学科发展迅速,新技术层出不穷;(4)偏重理论,实践与应用少。

本教材主要从开发、使用与维护计算机应用系统的角度出发,选取数据结构、操作系统、数据库系统、软件工程和计算机网络这5个重要技术的最基本的内容予以介绍,按照68学时编写。为了使没有学过C语言的读者(但学过Visual Basic或Visual FoxPro)也能看懂数据结构等章节的程序,本教材设计了一种类C语言,该语言既简便易懂,又保留了C语言的风格,用它编写的程序只要稍加变换就可以直接在C语言环境运行。操作系统介绍的是Windows 2000/XP,这是一种从最小的移动设备到最大的电子商务服务器都适用的基于网络的最新操作系统;数据库系统重点介绍关系数据库的结构化查询语言SQL及其在数据库编程中的使用方法;软件工程突出了面向对象技术,重点是UML的思想和方法;计算机网络,重点介绍TCP/IP协议及网络编程技术。在文字描述与实例选取方面,尽量做到由浅入深,结合实际。

本书编写得到了浙江省高校计算机教学研究会的大力支持,全书由杭州电子科技大学胡维华教授拟定编写大纲并最后修改定稿。第1章由胡维华教授编写,第2章由浙江大学陈天洲博士、副教授编写,第3章由浙江大学季江民博士、副教授编写,第4章由浙江工商大学魏贵义博士、副教授编写,第5章由浙江大学魏宝刚博士、副教授编写,第6章由杭州电子科技大学董云耀副教授编写。

由于编者水平有限,本教材的内容取舍和论述定有许多不当之处,书中的错误和疏漏在所难免,敬请广大读者批评指正。

本书编委会
2004年7月

目 录

第1章 概 论	1
1.1 计算机的硬件和软件	1
1.2 软件的发展过程	2
1.3 软件的基本内容	3
1.3.1 软件语言	3
1.3.2 数据结构	4
1.3.3 离散数学	4
1.3.4 软件方法学	5
1.3.5 软件工程	5
1.3.6 软件系统	6
小结	6
习题	7
第2章 数据结构与基本算法	8
2.1 序 论	8
2.1.1 基本概念和术语	8
2.1.2 数据类型和抽象数据类型	11
2.1.3 算法描述与算法分析	13
2.2 线性表	20
2.2.1 线性表的基本概念	20
2.2.2 线性表的顺序存储	22
2.2.3 线性表的链式存储	26
2.3 栈	34
2.3.1 栈的定义和运算	35
2.3.2 栈的逻辑结构特征	35
2.3.3 栈的实现	36
2.4 队 列	40
2.4.1 队列的定义和基本运算	40
2.4.2 队列的存储结构	41
2.5 数 组	47
2.5.1 数组的定义、逻辑结构和基本运算	47
2.5.2 数组的顺序表示和实现	48
2.5.3 矩阵的压缩存储	51
2.6 树与二叉树	56

2.6.1 树的基本概念	56
2.6.2 树的逻辑结构	57
2.6.3 二叉树的定义及性质	57
2.6.4 二叉树的存储结构	59
2.6.5 二叉树的遍历	61
2.6.6 二叉排序树	62
2.6.7 树的应用	63
2.7 查找	64
2.7.1 查找的基本概念	64
2.7.2 顺序查找	65
2.7.3 二分查找	65
2.7.4 索引顺序查找	67
2.7.5 Hash 查找	68
2.8 排序	70
2.8.1 排序的基本概念	70
2.8.2 直接插入排序	72
2.8.3 交换排序	73
2.8.4 选择排序	75
2.8.5 归并排序	77
小结	78
习题	79
第3章 操作系统	82
3.1 操作系统概述	82
3.1.1 操作系统的作用	82
3.1.2 操作系统类型	82
3.1.3 操作系统特征和功能	84
3.2 处理机管理	86
3.2.1 进程	86
3.2.2 进程控制	88
3.2.3 线程	89
3.2.4 进程同步	90
3.2.5 进程通信	94
3.2.6 进程调度	95
3.2.7 死锁	99
3.3 存储管理	105
3.3.1 存储管理概述	105
3.3.2 存储器的连续分配方式	107
3.3.3 分页存储管理方式	109
3.3.4 分段存储管理方式	111
3.3.5 段页式存储管理方式	114

3.3.6 虚拟存储器管理技术	114
3.4 设备管理	121
3.4.1 设备管理概述	121
3.4.2 I/O 控制方式	121
3.4.3 缓冲技术	122
3.4.4 设备的分配	123
3.4.5 设备驱动程序	125
3.4.6 磁盘 I/O	126
3.5 文件系统	127
3.5.1 文件和文件系统	127
3.5.2 文件的目录和管理	128
3.5.3 文件系统的实现	129
3.6 操作系统接口	133
3.6.1 用户与操作系统的接口	133
3.6.2 Windows 的接口	134
3.6.3 Linux 常用命令	135
3.7 Windows 2000/XP 操作系统概述	139
3.7.1 Windows 发展概述	139
3.7.2 Windows 2000/XP 的体系结构	140
3.7.3 Windows 2000/XP 的处理器管理	142
3.7.4 Windows 2000/XP 的内存管理	146
3.7.5 Windows 2000/XP 的文件系统	148
3.7.6 Windows 2000/XP 的 I/O 系统	148
小结	149
习题	150
第 4 章 数据库系统	156
4.1 数据库系统简介	156
4.1.1 数据库概念	156
4.1.2 数据模型	158
4.1.3 数据库系统的研究领域	161
4.1.4 关系数据库管理系统	162
4.2 关系模型	163
4.2.1 关系数据结构	163
4.2.2 关系的完整性约束	166
4.2.3 关系数据操作	168
4.3 SQL 查询语言	175
4.3.1 SQL 简介	175
4.3.2 SQL 的数据定义	177
4.3.3 SQL 的数据查询	181
4.3.4 视图	193

4.4 SQL 语言的数据库编程	197
4.4.1 嵌入式 SQL 简介	197
4.4.2 状态控制	201
4.4.3 一般嵌入式 SQL 的表达	202
4.4.4 数据库事务编程	210
4.5 数据库设计	213
4.5.1 数据库设计的特点	213
4.5.2 数据库设计方法介绍	214
4.5.3 数据库设计的基本步骤	214
4.6 并行与分布式数据库简介	215
4.6.1 并行数据库系统的可用结构	216
4.6.2 并行查询处理	217
4.6.3 数据操作的并行化	218
4.6.4 分布式数据库简介	220
小结	222
习题	222
第 5 章 面向对象的软件工程	225
5.1 软件工程概述	225
5.1.1 软件工程的定义	225
5.1.2 瀑布模型	226
5.1.3 快速原型模型	227
5.1.4 演化模型	228
5.2 需求分析和建模	229
5.2.1 软件需求与需求规格说明	229
5.2.2 结构化分析与建模	232
5.2.3 结构化程序设计	237
5.3 面向对象的概念和方法	239
5.3.1 抽象	240
5.3.2 类和对象	240
5.3.3 数据的封装	242
5.3.4 继承	243
5.3.5 聚合	244
5.3.6 关联	244
5.3.7 消息	245
5.3.8 多态性	245
5.3.9 面向对象模型	247
5.4 面向对象的分析/设计方法与模型	248
5.4.1 面向对象的分析和设计方法	248
5.4.2 UML 的概念模型	250
5.4.3 UML 的结构	251

5.4.4 UML 的图	252
5.5 UML 的静态建模机制	253
5.5.1 用例图	253
5.5.2 角 色	253
5.5.3 用 例	254
5.5.4 类 图	254
5.5.5 包 图	256
5.6 UML 的动态建模机制	257
5.6.1 UML 的消息	257
5.6.2 状态图	257
5.6.3 顺序图	258
5.6.4 协作图	259
5.6.5 活动图	261
5.7 面向对象的程序设计	264
5.7.1 封装性	264
5.7.2 继承性	266
5.7.3 多态性	268
5.7.4 模 板	273
5.8 基于构件的软件重用	275
5.8.1 构件特征和分类	276
5.8.2 类构件实现软件重用	277
5.8.3 基于构件的软件开发模型	277
5.8.4 CBSE 过程模型和标准	278
5.9 软件测试	281
5.9.1 测试计划	282
5.9.2 白盒与黑盒测试	282
5.9.3 测试步骤	284
5.9.4 面向对象的测试	287
小结	288
习题	288
第 6 章 计算机网络	291
6.1 计算机网络概论	291
6.1.1 计算机网络的基本组成	291
6.1.2 计算机网络的功能	293
6.1.3 计算机网络提供的服务	293
6.1.4 计算机网络的分类及拓扑结构	295
6.1.5 计算机网络的应用	297
6.2 数据通信基本概念	299
6.2.1 数据通信系统	299
6.2.2 多路复用	301

6.2.3 数据交换技术.....	303
6.2.4 差错控制.....	305
6.3 计算机网络基本技术.....	306
6.3.1 LAN 技术.....	306
6.3.2 WAN 技术.....	313
6.3.3 网络互连技术.....	318
6.3.4 路由选择技术.....	320
6.3.5 流量控制技术.....	321
6.3.6 拥塞控制技术.....	322
6.3.7 网络管理技术.....	324
6.3.8 网络安全技术.....	326
6.4 计算机网络体系结构与网络协议.....	327
6.4.1 网络协议与网络体系结构.....	327
6.4.2 网络体系结构参考模型.....	328
6.4.3 各层典型协议.....	331
6.5 网络软件设计基础.....	342
6.5.1 典型网络操作系统简介.....	342
6.5.2 网络计算模式.....	344
6.5.3 Windows 2000 Server 安装和配置.....	347
6.5.4 网络应用程序开发.....	350
6.5.5 网站的软件设计.....	353
小结.....	367
习题.....	367
部分习题参考答案.....	369
主要参考文献.....	377

第 1 章

概 论

计算机是 20 世纪人类的伟大创造，它对人类社会的发展与进步产生了巨大的影响。计算机由硬件与软件组成。硬件是计算机的物质基础，软件是计算机的灵魂，没有软件计算机就根本无法工作。计算机与人类历史上其他所有机器的根本区别，就在于它有软件这个灵魂。软件是智力产品，软件是一门精深的学科，软件有庞大的技术群。本章简要介绍计算机软件的作用，计算机软件的发展过程，以及计算机软件的基本内容，包括软件语言、数据结构、离散数学、软件方法学、软件工程和软件系统等。本章是对软件的一个概述，也是全书的导引。

1.1 计算机的硬件和软件

计算机是一个复杂的系统，它由硬件和软件组成。可以认为，硬件是计算机的“躯体”，软件是计算机的“灵魂”，两者相互依存，缺一不可。

广义的硬件包含硬件本身及其工程实现，即构成计算机系统的物质元器件、部件、设备，以及它们的工程实现（包括设计、制造和检测等技术）。构成计算机系统的部件和设备包括运算器、控制器、主存储器、辅助存储器、输入和输出设备、电源等。元器件包括集成电路、印制电路板及其他磁性元件、电子元件、光电子元件和插件等。硬件工程技术是指计算机的直接实现技术，主要有：印制电路板制造、高密度组装、抗环境（光、磁、电）干扰、抗恶劣环境（高温、高湿、震动、灰尘）破坏等技术，以及为提高计算机的可靠性、可维护性、可用性，在进行设计和制造时所采取的措施。

计算机硬件是计算机的物质体现。计算机问世初期，“计算机”一词实际上只是指计算机硬件。进入 20 世纪 60 年代，由于程序设计技术的进步，才形成“计算机硬件和软件”的概念。

软件包含各种程序与有关文档，以及它的工程实现（程序设计、软件工程等）。程序是计算任务的处理对象和处理规则的描述，文档是为了便于设计和了解程序所需的阐明性资料。程序必须装入机器内部才能工作，文档一般是给人看的，不一定装入机器。细言之，软件一词有 3 层含义：

一为个体含义，即指计算机系统上的程序和文档。

二为整体含义，即指在特定计算机系统中所有上述个体含义下的软件的总称。

三为学科含义，即指在研究、开发、维护以及使用前述含义下的软件所涉及的理论、方法、

技术所构成的学科。

在第 3 种含义下, 软件宜称为“软件学”, 但一般仍称作软件。“软件”一词源于程序, 到了 20 世纪 60 年代初期, 人们逐渐认识到和程序有关的文档的重要性, 从而出现了软件一词。

软件是用户与硬件之间的接口界面。要使用计算机, 就必须编制程序, 必须有软件。用户主要通过软件与计算机进行交往。软件是计算机系统设计的重要依据。为了方便用户, 为了使计算机系统具有较高的总体效用, 在设计计算机系统时, 必须全局考虑软件与硬件的结合, 以及用户的要求和软件的要求。

发展计算机科学与技术, 硬件和软件都是不可缺少的重要方面。软件的发展以硬件为基础, 其发展也促进了硬件、计算机科学与技术, 以及其他学科与技术的发展, 它在社会信息化和人类文化的发展中具有重要作用。

1.2 软件的发展过程

软件的发展过程大致可分为 3 个阶段。

从第一台计算机上的第一个程序的出现到实用的高级程序设计语言出现以前为第一阶段 (1946~1956 年)。计算机的工作是由储存在其内部的程序所指挥的, 这是冯·诺依曼式计算机的重要特色。当时计算机的应用领域较窄, 主要是科学计算。就一项计算任务而言, 输入、输出量并不大, 但计算量较大, 主要处理一些数值数据。机器结构以中央处理器为中心, 存储容量较小。编制程序所用的工具是低级语言, 即以机器基本指令集为主的机器语言和在机器语言基础上稍加符号化的汇编语言。突出的问题是, 程序的设计和编制工作复杂、繁琐、费时和易出差错。衡量程序质量的标准主要是功效, 即运行时间省、占用内存小, 很少考虑到结构清晰、易读性和易维护性。设计和编制程序采用个体工作方式, 强调编程技巧, 主要研究科学计算程序、服务性程序和程序库, 研究对象是顺序程序。当时人们对和程序有关的文档的重要性认识不足, 重点考虑程序本身, 尚未出现软件一词, 但毕竟由于程序是软件的主体, 从发展的连续性来看, 将其归为第一阶段。

从实用的高级程序设计语言出现以后到软件工程出现以前为第二阶段 (1956~1968 年)。随着计算机应用领域的逐步扩大, 除了科学计算继续发展以外, 出现了大量的数据处理问题, 其性质和科学计算有明显区别, 涉及到非数值数据。就一项计算任务而言, 计算量不大, 但输入、输出量较大。这时, 机器结构转向以存储控制为中心, 出现了大容量的存储器, 外围设备发展迅速。为了提高程序人员的工作效率, 出现了实用的高级程序设计语言。为了充分利用系统资源, 出现了操作系统。为了适应大量数据处理问题的需要, 开始出现数据库及其管理系统。在 20 世纪 50 年代后期, 人们逐渐认识到和程序有关的文档的重要性, 从而到了 60 年代初期, 出现了软件一词, 融入程序及其有关的文档为一体。这时, 软件的复杂程度迅速提高, 研制周期变长, 正确性难以保证, 可靠性问题相当突出。到了 60 年代中期, 出现了人们难以控制的局面, 即所谓软件危机。为了克服这一危机, 人们进行了以下 3 个方面的工作: ①提出结构程序设计方法; ②提出用工程方法开发软件; ③从理论上探讨程序正确性和软件可靠性问题。这一阶段的研究对象增加了并发程序, 着重研究高级程序设计语言、编译程序、操作系统, 以及各种应

用软件。计算机系统的处理能力得到加强。设计与编制程序的工作方式逐步转向合作方式。

软件工程出现以后迄今为第三阶段（1968 年以来）。由于大型软件的开发是一项工程性任务，采用个体或合作方式不仅效率低、产品可靠性差，而且很难完成，只有采用工程方法才能适应。从而在 1968 年的大西洋公约学术会议上提出软件工程。30 多年来，软件领域工作的主要特点是：

第一，随着微型计算机的出现，分布式应用和分布式技术得到发展；随着应用领域的不断扩大，出现了嵌入式应用和嵌入式软件，其特点是受制于它所嵌入的宿主系统，而不只是受制于其功能要求；为了适应计算机网络的需要，出现了网络软件；为了使计算机综合处理文本、图形、图像、声音、动画、视频图像等多种不同类型媒体的信息，出现了多媒体软件；为了让人能使用自然技能对虚拟世界中的物体进行考察和操作，同时提供视觉、听觉、触觉等多种直观而又自然的感知，出现了虚拟现实软件。

第二，开发方式逐步由个体合作方式转向工程方式，软件工程发展迅速，特别是出现了计算机辅助软件工程。除了开发各类工具与环境，用以支撑软件的开发与维护外，还出现了一些实验性的软件自动化系统。

第三，注意研究软件理论，致力研究软件开发过程本身，研究各种软件开发风格与模型，例如，功能分解风格与模型、面向对象风格与模型等。

第四，除了软件传统技术继续发展外，人们着重研究以智能化、自动化、并行化、网络化，以及自然化为标志的软件开发新技术。

1.3 软件的基本内容

按照软件的学科含义，它的基本内容主要有软件语言、数据结构、离散数学、软件方法学、软件工程、软件系统等。

1.3.1 软件语言

软件语言是指用于书写软件的语言。可分为需求级语言、功能级语言、设计级语言、实现级语言，以及文档语言。需求级语言用以书写软件需求定义，又称为需求定义语言，目前多数仍是非形式的语言或半形式化语言。功能级语言用以书写软件功能规约，又称功能规约语言，可以是形式的，也可以是非形式的。设计级语言用以书写软件设计规约，又称设计规约语言，一般是形式的。实现级语言，即一般的程序设计语言，用以书写计算机程序，而计算机程序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，处理对象是数据或信息，处理规则反映处理动作和步骤。程序设计语言的基本成份是：①数据成份；②运算成份；③控制成份；④传输成份。程序设计语言是最为实质性的，相对来说，较为成熟，也最为庞大，低级语言、高级语言从 FORTRAN、ALGOL、COBOL、LISP、BASIC、PASCAL、C、XCY、Ada，直到 Smalltalk、C++、MATLAB、Delphi、Java、C# 等均属此类。文档语言用以书写文档，目前一般是非形式的。

1.3.2 数据结构

数据结构是指由若干数据成份按照一定方式构成的复合数据以及作用于其上的函数或运算。数据成份及其间的数据约束关系合称为数据结构的逻辑构成或逻辑结构。数据结构是指在数学上可以用适当的数学结构以及在其上的函数变换统一地定义。以字符为例，其逻辑结构是一组同一类型（字符型）的字符，以及在该字符串中字符间的前后顺序关系。在此结构上的常见运算有：求字符串的长度、将字符串分为两部分，以及求两字符串拼接后的新字符串等。在设计数据结构阶段，应当视应用需要确定适当的运算。

一个数据结构的存储结构是指它在计算机中所需用的存储空间、空间的构成结构，以及对该存储结构的访问方式等的统称。一般说来，存储结构的具体设计是在明确该数据结构的定义后进行的。

在传统的程序设计语言（如 PASCAL、C 等）中，定义数据结构的基本途径是采用数据类型。简单的数据结构是用单一的标准数据类型，如整型、字符型、布尔型、实型等所定义的。在使用时，程序对其采取整体访问方式，即读写访问只对整体而不涉及对其某构成部分的访问（如某二进位的内容）。由若干较简单的数据结构，运用程序语言所提供的复合构成方法，诸如数组、记录、集合等，可以构造更为复杂的数据结构。复杂数据结构的特点是：对它既可进行整体性访问，也可单独对某构成成份进行访问。以上所述，虽已被多数熟知的程序设计语言所采用，但是有其局限性。它偏重于逻辑结构及存储结构的构成方面，而对其运算等语义部分的描述不足。根据抽象数据类型理论，程序应将数据结构的逻辑构成和它的运算一起定义，并封装成一个整体。而且数据结构的具体实现，包括采用的存储结构和运算的具体算法实现，应该与上述封装相分离，分别地加以描述。近几年使用的面向对象语言，如 C++、Smalltalk、MATLAB、Delphi、Java 等均采用了这一思想。

存储结构涉及存储分配和数据访问两个方面。鉴于计算机的主存储器具有随机访问及相邻地址顺序存储等物理特点，常把经常被同时访问的数据安排在相邻的物理存储区域内，这称为顺序存储结构。一维数组、字符串和记录等定长的数据结构多采取这种结构，并运用静态存储分配方式，即在程序编译连接时（程序运行前）确定所占用的存储区域的物理地址范围。对顺序存储结构而言，常用的数据访问方式有顺序访问、索引访问和散列访问等多种。另一种存储结构是链接结构。与顺序存储结构不同，它常常采用动态存储方式：在程序运行中当需要创建新的数据结点时，申请和分配存储空间；当删除数据结点时则及时释放其占用空间。这种分配方式适合于变长度的数据结构，如具有递归逻辑关系的结构（表、树等）。

数据的逻辑结构从比较抽象的角度（与存储结构相对独立），刻画数据结构所具有的数学性质：将数据元素抽象为结点，数据元素间的关系当作连接结点的边，而访问数据结构的运算则描述为离散数学结构上的运算。常见的逻辑结构有线性结构、树结构和图结构等。

1.3.3 离散数学

离散数学是数学中一个讨论离散对象的分支。和连续数学不同，离散数学通常涉及整数系。我们知道，数字计算机内部处理的数据都呈整型，因此离散数学在计算机软件，乃至整个计

计算机科学与技术中有着十分重要的作用。通常认为离散数学包括集合论、图论、组合学、数理逻辑、抽象代数、差分方程、离散概率论等。集合论的研究对象是一般集合。按现代数学的观点,数学各分支的研究对象,或者是带有某种结构的集合(如群、环、域、拓扑空间等),或者是可用集合直接定义(如自然数、有理数、实数、函数等),或者是可借助集合定义(如范畴、函子、自然变换等)。图论是研究图的性质的学科。图论中的图并非初等数学中的图,后者只是连续函数的图形,图论中的图却是一组顶点(结点)和一组连接两两顶点的边所构成的集合。组合论讨论计算某类对象个数的方法,多数组合论问题可归结为存在性问题、枚举性问题或选择性问题。数理逻辑研究形式体系,作为其组成部分的命题演算和谓词演算等在计算机软件的研究中作用巨大。诸如程序正确性验证、人工智能等领域无不用到数理逻辑。抽象代数讨论离散对象结构,它在计算机科学与技术中应用广泛。例如,群应用于编码理论,半群应用于形式语言理论和自动机理论。差分方程、离散概率论等在数值计算、计算理论或程序理论中都有着广泛应用。

1.3.4 软件方法学

软件方法学是指软件开发全过程的指导原则与方法体系。其另一种含义是以软件方法为研究对象的学科。从开发风格上看,软件方法有自顶向下的开发方法、自底向上的开发方法。在实际软件开发中,大多是自顶向下与自底向上两种方法的结合,只不过是哪种方法为主而已。从性质上看,有形式方法与非形式方法。形式方法是一种具有坚实数学基础的方法,从而允许对系统和开发过程作严格处理和论证。非形式方法则不把严格性作为其主要着眼点。从适用范围来看,有整体性方法与局部性方法。适用于软件开发全过程的是整体性方法,如自顶向下方法、自底向上方法、各种软件自动化方法均为整体性方法。适用于开发过程个别阶段的为局部性方法,如适用于需求分析阶段的各种需求分析方法,适用于设计阶段的各种设计方法等。此外,由于程序设计方法的发展相对较为成熟,从而早在软件方法学出现以前,就出现了程序设计方法学,它研究各类程序设计方法,如过程式程序设计、逻辑式程序设计、函数式程序设计、对象式程序设计、顺序程序设计、并发程序设计、并程序序设计、分布程序设计和可视程序设计等。

1.3.5 软件工程

软件工程是指应用计算机科学、数学、管理学、经济学等原理,以工程化方法制作软件的工程。它是一门交叉性学科。软件工程的架构可用——三元组刻画,即 $ES = (G, P, Q)$, 其中 ES 表示软件工程, G 为目标, P 为原则, Q 为过程。目标 G 主要包括正确性、可用性,以及价格合宜。正确性反映软件产品实现相应功能规约的程度。E.W.Dijkstra 曾说:“迄今的软件排错措施只能发现软件错误,不能断定软件没有错误。”20 世纪 60 年代中期以来,虽有不少计算机科学家致力于正确性问题的研究,但迄今为止,不论理论上还是实践上正确性问题均未得到很好解决。可用性反映软件的基本结构、实现及其文档为用户可用的程度。由于软件系统的用户广泛,要求因人而异,因而可用性往往是软件工程难以满足的目标。价格合宜反映软件开发与运行的总代价满足用户要求的程度。随着硬件价格的日趋低廉,传统的系统性能显得不太重要,价格合宜扩大成包括软件运行与维护方面的价格考虑。由于复杂软件系统不可能一步开发

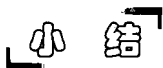
成功,通常采用增殖式开发策略,先生产出一个基本核心产品,随后再改进提高,以实现所需的全部功能。总价格合宜便要求产品能便于修正与改进。原则 P 主要包括易变性原则、综合性原则、支撑性原则,以及管理性原则等。其中易变性原则指必须认识软件需求的易变性,综合性原则指综合使用各种方法与工具以实现软件目标,支撑性指应充分认识软件支撑环境的作用,管理性指必须认识管理过程的作用。过程 Q 涉及软件生存周期中的各类过程,主要包括基本过程、支撑过程和组织过程。

1.3.6 软件系统

软件系统是计算机系统中由软件组成的子系统。软件可分为系统软件与应用软件两大类。

居于计算机系统中最靠近硬件的一层的软件称为系统软件,其他软件一般都通过系统软件发挥作用。它与具体的应用领域无关。典型的系统软件是操作系统、语言处理系统、数据库系统、分布式软件系统、网络软件系统、人机交互软件系统,以及支撑软件开发与维护的各类软件工具。操作系统是用以管理系统资源的软件,旨在提高计算机的总体效用。一般包括处理机管理、存储管理、设备管理、信息管理(文件系统)、用户接口等。语言处理系统包括各种类型的语言处理程序,如解释程序、汇编程序、编译程序、编辑程序、装配程序等。数据库系统包括数据库及其管理系统。数据库是相互关联的在某种特定的数据模式指导下组织而成的各种类型的数据的集合。数据库管理系统则是为数据库的建立、使用和维护而配置的软件,它建立在操作系统的基础上,对数据库进行统一的控制和维护。它一般包括模式翻译、应用程序编译、查询命令解释执行,以及运行管理等部分。分布式软件系统是管理、支撑分布式计算系统的软件系统,它一般包括分布式操作系统、分布式程序设计语言及其编译程序、分布式数据库管理系统、分布式算法及其软件包、分布式开发工具包等。网络软件系统是在计算机网络环境中,用于支持数据通信和各种网络活动的软件系统。它主要包括通信软件、网络协议软件和网络应用系统、网络服务管理系统,以及用于特殊网络站点的软件等。人机交互软件系统是人机交互系统中的软件子系统,它一般包括人机接口软件、命令语言及其处理系统、用户接口管理系统、多媒体软件等。随着计算机科学与技术的发展,软件的开发和维护的代价在整个计算机系统中所占比重很大,远远超过硬件。因此,支撑软件开发与维护的各类软件的研制具有重要意义。目前,主要有环境数据库、各种接口软件和工具组等。

为解决具体应用问题而设计的软件称为应用软件。例如中文信息处理软件、数字图像处理软件、计算机辅助设计软件、计算机辅助管理软件、计算机辅助教学软件、计算机控制软件、计算机仿真软件、计算机图形处理软件、多媒体计算软件、虚拟现实软件、气象预报软件和人口普查软件等,种类繁多,数不胜数。对于具体的应用领域,应用软件的质量往往成为影响实际效果的决定性因素。



本章给出了计算机硬件与软件的定义,概述了计算机软件的作用、发展过程及其 6 大基本内容:软件语言、数据结构、离散数学、软件方法学、软件工程和软件系统,并对操

作系统、语言处理系统、数据库系统、分布式软件系统、网络软件系统、人机交互软件系统和工具软件等常用系统软件作了概括性描述。本书后继章节将对数据结构、操作系统、数据库系统、软件工程和计算机网络这 5 个最基本的技术作具体、深入的介绍,使读者系统了解整个计算机软件技术基础的概念范畴,学会基本数据结构的应用和常用算法的设计,知道多任务图形界面操作系统的工作原理和操作系统接口软件的设计方法,掌握关系数据库模型和应用数据库的设计技能,了解软件工程方法和面向对象软件设计的基本技术,掌握 Internet 的基本原理和网络软件的开发方法,成为本专业领域内计算机应用的高级专门人才打下坚实的基础。

习 题

1. 什么是软件的个体含义、整体含义与学科含义?
2. 简述软件的发展过程。
3. 按照软件的学科含义,它的基本内容主要有哪些?
4. 软件语言可分为哪几级?
5. 试简述数据的逻辑结构与物理结构。
6. 为什么离散数学在计算机软件学科中有着十分重要的作用?
7. 何谓软件方法学与软件工程?
8. 试列出软件系统的基本组成。