



研究生教材

软件开发技术

冯博琴 陆丽娜 魏恒义
齐 勇 姜锦虎

西安交通大学出版社

研究生教材

软件开发技术

冯博琴 陆丽娜 魏恒义 齐勇 姜锦虎

西安交通大学出版社

内容简介

本书介绍了软件开发技术的基本理论知识、技术支持和开发高质量软件的技术手段。全书分为三大部分,共8章。第一部分(第1~3章)介绍了数据结构、编译原理和操作系统;第二部分(第4~7章)是软件开发的工具、环境和方法,这是进行软件开发的最基础的知识,非常实用;第三部分是软件工程(第8章),它是软件开发进入工程化,设计高质量、商品化软件所必须掌握的内容。

全书从深、广度方面加强了研究生的软件基础知识,提高软件素质,同时为他们开始动手进行软件开发到设计高质量软件提供了丰富有效的技术支持。

本书可作为研究生和大学高年级学生软件开发教材,也可供从事计算机软件应用和开发的技术人员参考。

(陕)新登字 007 号

软件开发技术

冯博琴 陆丽娜 魏恒义

齐 勇 姜锦虎

责任编辑 曹晓梅 李 勤

*

西安交通大学出版社出版发行

西安市咸宁西路28号 邮政编码 710049

陕西省轻工业厅印刷厂印装

陕西省新华书店经销

开本:787×1092 1/32 印张:16.875 字数:425千字

1995年9月第1版 1995年9月第1次印刷

印数:1—5000

ISBN7-5605-0770-0/TP·105 定价:15.00元

研究生教材总序

研究生教育是为国家培养高层次人才的,它是我国高等教育的最高层次。研究生必须在本门学科中掌握坚实的基础理论和系统的专门知识,具有从事科学研究或担负专门技术工作的能力。这些要求具体体现在研究生的学位课程和学位论文中。

认真建设好研究生学位课程是搞好研究生教学的重要环节。为此,我们组织出版这套以公共课和一批新型学位课程为主的研究生教材,以满足当前研究生教学的需要。这套教材的作者都是多年从事教学、科研,具有丰富经验的教师。

这套教材首先着眼于研究生未来工作和高技术发展的需要,充分反映国内外最新学术动态,使研究学习之后能迅速接近当前科技发展的前沿,以适应“四化”建设的要求;其次,也注意到应有的基本理论和基本内容,以保持学位课程内容的相对稳定性和系统性,并具有足够的深广度。

这套研究生教材虽然从提出选题、拟定大纲、组织编写到编辑出版,都经过了认真的调查论证和细致的工作,但毕竟是第一次出版这样高层次的系列教材,水平和经验都感不足,缺点和错误在所难免。希望通过反复的教学实践,广泛听取校内外专家学者和使用者的意见,使其不断改进和完善。

西安交通大学研究生院
西安交通大学出版社

75157/406

前 言

人类将在下一世纪进入信息化社会,人类的各项活动都将和信息密切相关。计算机是信息化社会的三大基础之一(其余两个基础是通信和控制),因此,计算机技术的重要性易被人们普遍接受。

事实上,计算机技术已成为一门社会技术,许多专业工作,如工程设计、交通通信、财政金融、商业外贸、行政管理、农业技术、动力能源、文化教育等都离不开计算机,工程技术人员使用计算机水平的高低直接影响他们的工作质量。随着信息化社会进程加快,现在人们的工作方式也在迅速改变,如电子邮件、校园网和综合业务数字网等信息技术日渐普遍地应用,这要求高等技术人才必须具备很强的计算机应用技术和意识,因此,用人部门也将计算机技术水平高低作为衡量人才的主要标准之一。

研究生教育作为高层次教育必然要注视和适应信息化社会对于人才素质的培养和知识结构的更新所提出的全部要求。本书是根据目前大学毕业生的计算机教育水平和研究生教育的实际需要编写的,旨在提高研究生的计算机素质,并为他们进行软件开发课题提供必要的技术支持。

作者根据长期从事研究生计算机教育的教学经验积累,把研究生进入课题之前必须具备的软件开发技术进行了精选,编成本书。全书由三个部分组成。第一部分是软件的基础,包括数据结构、编译原理和操作系统;第二部分是软件开发的工具、环境和方法的最基本内容,包括语言工具和环境、编辑工具、硬件配置方法、语法分析和错误检测,以及程序的动态调试;第三部分是软件工程的基本知识。这三大部分构成一个整体;第一步是对研究生已有的软件基础进行提高和系统化,第二步为他们进入课题提供最直接的技

术支持,第三步为进行高质量软件开发提供技术手段。

本书由冯博琴主编,参加编写的有冯博琴(第1、2章),陆丽娜(第3章),魏恒义(第4、6章),齐勇(第5、7章),姜锦虎(第8章)。此书编写中一直得到我校研究生院林贤玖老师的关心和帮助,在此表示衷心的感谢。

计算机技术日新月异,加之作者水平有限,因此本书取材可能有不当之处,谬误也在所难免,恳望同行指正。

冯博琴

1995年3月



作者简介

冯博琴, 1942 年生于浙江温州, 1965 年毕业于西安交通大学数理系应用数学专业, 同年留校任教。现任计算机系教授, 从事计算机软件的研究生和本科生教学以及科研工作, 在计算机编译系统、智能 CAD、管理信息系统和 CAI 方面做过不少工作, 获得省部级和校级科研成果十余项, 发表论文廿余篇, 同时出版了《实用专家系统》、《编译原理辅助教程》、《机械零件的计算机辅助设计》, 主编《跟我学高级语言编程系列丛书》、《非计算机专业计算机教育系列丛书》等 17 本著作。

3031406

目 录

第1章 数据结构

§ 1.1 线性数据结构	(2)
1.1.1 线性表及其存储结构	(2)
1.1.2 栈	(5)
1.1.3 队列	(9)
§ 1.2 链表和串	(14)
1.2.1 链表	(15)
1.2.2 串	(27)
§ 1.3 非线性数据结构	(30)
1.3.1 树的基本概念	(30)
1.3.2 二叉树	(33)
1.3.3 遍历二叉树	(36)
§ 1.4 查找	(37)
1.4.1 顺序查找	(37)
1.4.2 折半查找	(38)
1.4.3 HASH 法	(39)
§ 1.5 排序	(43)
1.5.1 插入排序	(44)
1.5.2 冒泡排序	(45)
1.5.3 二路合并排序	(47)
§ 1.6 递归	(50)
1.6.1 递归定义	(51)
1.6.2 递归算法	(51)

1.6.3 递归的代价.....	(54)
§ 1.7 实例.....	(56)
1.7.1 迷宫问题.....	(56)
1.7.2 Tower of Hanoi 问题	(59)
1.7.3 Huffman 树.....	(60)

第2章 编译初步

§ 2.1 编译过程.....	(63)
§ 2.2 词法分析.....	(69)
2.2.1 词法分析器的手工实现.....	(69)
2.2.2 正则表达式和有限自动机.....	(75)
2.2.3 词法分析器的自动产生.....	(81)
§ 2.3 语法分析.....	(85)
2.3.1 文法和语言.....	(85)
2.3.2 语法树和二义性.....	(88)
2.3.3 语法分析方法概述.....	(90)
2.3.4 算符优先分析法.....	(93)
2.3.5 递归下降分析法.....	(96)
§ 2.4 中间代码生成	(100)
2.4.1 中间代码的形式	(101)
2.4.2 简单算术表达式和赋值语句的翻译	(106)
2.4.3 布尔表达式的翻译	(108)
2.4.4 控制类语句的翻译	(114)
2.4.5 标号与 GO TO 语句	(117)
§ 2.5 符号表	(118)
2.5.1 符号表的内容及运算	(119)
2.5.2 Fortran 语言符号表.....	(119)
§ 2.6 运行时存储分配	(121)
2.6.1 静态与动态存储方案	(122)
2.6.2 Fortran 存储组织与分配.....	(122)

2.6.3	C 语言的存储组织与分配	(132)
第 3 章 操作系统		
§ 3.1	操作系统的基本概念	(137)
3.1.1	什么是操作系统	(137)
3.1.2	操作系统的分类	(141)
§ 3.2	操作系统的基本功能	(150)
3.2.1	作业管理	(151)
3.2.2	进程管理	(153)
3.2.3	存储管理	(162)
3.2.4	设备管理	(167)
3.2.5	文件管理	(170)
§ 3.3	微机常用操作系统举例	(174)
3.3.1	IBM PC-DOS	(174)
3.3.2	XENIX 操作系统的使用	(196)
3.3.3	NOVELL Netware 局域网络操作系统	(220)
§ 3.4	MS-DOS 应用实例	(244)
3.4.1	MS-DOS 的实时应用	(244)
3.4.2	计时器应用举例	(255)
3.4.3	双机通信的实现(RS232 异步通信的实现) ...	(263)
第 4 章 软件开发过程中对环境工具的要求		
§ 4.1	软件开发的一般过程	(273)
§ 4.2	对环境及工具的要求	(274)
4.2.1	程序设计语言的比较及选用	(275)
4.2.2	机器硬件配置对软件开发的影响	(277)
4.2.3	PC 机系统配置	(279)
4.2.4	扩展内存,扩充内存的使用	(282)
4.2.5	系统软件的选用	(283)
§ 4.3	软件工具概况	(286)
4.3.1	系统管理工具	(287)

4.3.2 机器性能测试工具	(292)
第5章 面向程序设计语言的编辑工具	
§ 5.1 编辑器的引入	(300)
§ 5.2 编辑器的功能及实现思想	(300)
§ 5.3 编辑程序的分类	(302)
§ 5.4 常用通用编辑器介绍	(303)
5.4.1 DOS 环境下的行编辑器 Edlin 及 UNIX/XENIX 环境下的行编辑器 ed.	(303)
5.4.2 两个典型的全屏幕编辑器 vi 和 PE	(319)
§ 5.5 嵌入到集成环境中的编辑器的使用	(334)
5.5.1 PC Tool 中的编辑功能(PCShell 类似)	(335)
5.5.2 Turbo 系列集成环境下编辑器的使用	(341)
§ 5.6 没有编辑工具时如何建立文本文件	(345)
§ 5.7 如何选择编辑器	(346)
第6章 语法分析及错误检测	
§ 6.1 概述	(348)
§ 6.2 如何使源程序成为可执行的机器代码	(348)
§ 6.3 一些常见编译错误的排除	(349)
6.3.1 错误的表现形式	(349)
6.3.2 几种常用的排错方法	(350)
§ 6.4 利用语言处理程序调试程序	(351)
6.4.1 编译程序的组织及使用方法	(351)
6.4.2 用 Turbo Pascal 编译系统排错	(352)
第7章 程序的动态调试	
§ 7.1 运行错误及纠错方法	(358)
7.1.1 运行错误及原因	(358)
7.1.2 如何纠正运行错误	(362)
§ 7.2 逻辑错误及纠正方法	(364)

第8章 软件工程

§ 8.1 绪论	(370)
8.1.1 软件危机	(370)
8.1.2 软件的特点和分类	(371)
8.1.3 软件生存周期	(375)
8.1.4 软件质量	(383)
8.1.5 软件开发讲究的原则	(387)
8.1.6 软件开发中存在的几种偏见	(391)
§ 8.2 需求分析	(393)
8.2.1 需求分析的任务	(393)
8.2.2 需求分析的内容	(394)
8.2.3 需求分析的方法	(395)
8.2.4 需求分析的文档	(403)
8.2.5 评审	(405)
§ 8.3 概要设计	(405)
8.3.1 概要设计的任务	(405)
8.3.2 软件结构与结构图	(406)
8.3.3 软件设计的概念和原理	(409)
8.3.4 设计准则	(415)
8.3.5 面向数据流的设计方法	(420)
8.3.6 概要设计应交付的文档	(430)
§ 8.4 详细设计	(433)
8.4.1 详细设计的主要任务	(433)
8.4.2 详细设计思想	(434)
8.4.3 详细设计方法	(437)
8.4.4 模块开发卷宗	(451)
8.4.5 详细设计评审	(451)
§ 8.5 编码	(452)
8.5.1 编码阶段的任务	(452)

8.5.2	程序设计语言	(452)
8.5.3	编码风格	(456)
§ 8.6	软件测试	(465)
8.6.1	测试的目的	(466)
8.6.2	测试方法论	(466)
8.6.3	测试步骤	(468)
8.6.4	单元测试	(468)
8.6.5	组装测试	(472)
8.6.6	有效性测试	(475)
8.6.7	测试用例设计	(476)
8.6.8	测试文档	(478)
§ 8.7	一个实例	(478)
参考文献		(522)

第 1 章 数据结构

处理信息至少要设法把这些信息“逻辑地”描述清楚,然后“物理地”在计算机内存储起来,并且根据需要编写一些算法对这些信息进行加工。数据结构正是研究在确定的结构中各数据元素之间的逻辑关系,在计算机内的存储方式以及对它们允许进行的运算。数据结构的研究工作对于计算机科学发展有着举足轻重的作用。学术界有一种观点,认为计算机科学中具有核心作用的概念是“信息结构的转换”,计算机科学就是“一门关于信息转换的科学”,因而构造有关信息结构转换的模型,对它进行研究,是计算机科学中的根本性问题。另一种观点认为“计算机科学是算法的学问”。我们且不去断言谁是谁非,但“算法”与“信息结构”之间的紧密联系是勿容置疑的:没有信息结构,算法研究是无的放矢,没有算法的信息结构则无存在价值。

近十年来,微机的应用非常普及,计算机技术成了大众化工具,人们的计算机意识日益加强,因此计算机处理的对象日趋复杂化,信息的载体由数字、文字发展到声音、图形和图象等,使信息量直线上升,存储量、速度等矛盾突出……,这些问题都迫使人们去研究更好的算法和更加理想的数据结构技术。

数据结构是计算机科学的一门非常重要的基础课程,本章介绍它的基本概念、算法,以及一些重要应用,对于熟悉基本概念的读者,可以跳过它们。

§ 1.1 线性数据结构

本节讨论数据结构的一种最简单、最常见形式——线性表。

1.1.1 线性表及其存储结构

1. 线性表

线性表或者为空,或者可写成

$$(a_1, a_2, \dots, a_n)$$

其中, a_i 是取自某一集合的元素。

因此线性表是数据元素 $a_1, a_2, \dots, a_n$ 的有限集合, 并且 $n \geq 0$ 。它有以下特点:

(1) 数据元素的有限序列。它们之间的相对位置是线性的, 即除 a_1 和 a_n 外, 所有的 a_i 有且仅有一个直接前趋和一个直接后继元素。

(2) 线性表的长度是数据元素的个数 n 。

(3) 元素 a_i 在表中位置由序号 i ($0 \leq i \leq n$) 决定。

2. 线性表的存储结构

原始数据(信息)必须进入计算机的存储器, 存储器中存入或取出的单位容量叫单元。每一个单元都有唯一的地址供识别和存取用。存储单元中的内容可以是数值、符号串或一个图象元素等数据。当然, 一个单元的地址和该单元的内容是不同的。

对于线性表的数据元素(以下简称元素)可以用一组连续的存储单元依次存放, 此方法称作顺序分配法。当然还可以用其它的存储方式来表示。这里仅介绍线性表的顺序分配法。当元素由若干数据项构成, 如每个数据元素均为 100 个字符时, 该数据元素往往需要多个存储单元进行存储和处理。下面讨论线性表的顺序分配表示法。

对于线性表

$$(a_1, a_2, \dots, a_n)$$

假设每个元素 $a_i (1 \leq i \leq n)$ 占 K 个存储单元, 用 $\text{Loc}(a_i)$ 表示第一个元素存储单元的地址, 则表的第 i 个元素的存储地址 $\text{Loc}(a_i)$ 为

$$\text{Loc}(a_i) = \text{Loc}(a_1) + (i - 1) * K$$

公式(2-2)可由下面的关系式推出

$$\begin{aligned} \text{Loc}(a_i) &= \text{Loc}(a_{i-1}) + K = \text{Loc}(a_{i-2}) + 2 * K \\ &= \dots = \text{Loc}(a_1) + (i - 1) * K \end{aligned}$$

可见每个元素存储地址都与第一个元素的地址相差一个与序号成正比的常数。用图 1-1 示意线性表的顺序分配。

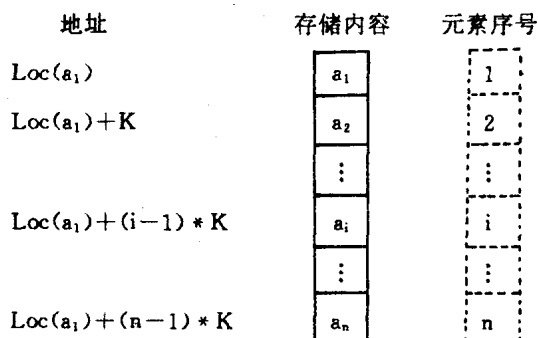


图 1-1 线性表顺序分配示意图

从图中不难看出, 线性表元素之间的相互位置可以用序号来描述。这种物理结构的特点可以用下述方法表示:

(1) 用任一字母数字串来表示线性表, 如: A、V、AX1、BWATC、XYZ 等可用来表示某一线性表;

(2) 用一个整数标号(亦称下标)标志线性表中元素, 如 $A[i]$ ($1 \leq i \leq n$) 表示线性表 A 的第 i 个元素 a_i 在计算机存储器中的映象。

可见存储线性表中的元素只要确定元素的下标值即可。因此, 顺序分配的线性表中任意一个元素的存取所需的时间都相等, 这是一个重要特点。

3. 线性表的运算

线性表的运算种类很多,其中包括以下几种:

- (1) 求表的长度;
- (2) 从左到右(或从右到左)读表;
- (3) 检索或修改第 i 个元素 ($1 \leq i \leq n$);
- (4) 将一个新元素放在位置 i 上, $1 \leq i \leq n+1$, 使原编号 $i, i+1, \dots, n$ 的元素变成 $i+1, i+2, \dots, n+1$;
- (5) 删去位置 i 上的元素 ($1 \leq i \leq n$), 使编号为 $i+1, i+2, \dots, n$ 的元素变成 $i, i+1, \dots, n-1$ 。

使用顺序分配法对线性表中任意元素的值进行检索或修改,方法较简单,只要控制下标值即可随机存、取和修改。操作 4、5 要花点功夫,必须移动某些余下的元素使顺序映象保持适当的形式。下面给出相应的算法。

算法 1-1 线性表的插入算法

```
procedure INSERT-LIST(v,n,i,x);  
{在长度为 n 的线性表中,插入一个元素 x 到第 i 个元素之前。  
v 为存储线性表的一维数组}  
begin  
  if (i<1) or (i>n+1) then write ('ERROR')  
  else [for j:=n down to i do  
        v[j+1] := v[j];  
        v[i] := x;  
        n := n+1 {改变表长度}]  
end; {INSERT-LIST}
```

算法 1-2 线性表的删除算法

```
procedure DELETE-LIST(v,n,i,y);  
{在长度为 n 的线性表中删除第 i 个元素,并送到 y 中}  
begin  
  if (i<1) or (i>n)
```