

第 2 卷

全国高等院校财经专业试用教材

信息处理概论

江 昭 吴民伟 晋良颖

中国人民大学出版社

全国高等院校财经专业试用教材

信息处理概论

第二卷

江 昭 吴民伟 晋良颖

中国人民大学出版社

全国高等院校财经专业试用教材

信息处理概论

第二卷

江 昭 吴民伟 晋良颖

中国人民大学出版社出版

(北京西郊海淀路39号)

民族印刷厂印刷

新华书店北京发行所发行

开本: 850×1168毫米32开 印张: 8.25

1987年10月第1版 1987年10月第1次印刷

字数: 203,000 册数: 1—10,000

ISBN 7-300-00126-2/0·10

书号: 13011·37 定价: 2.00元

目 录

第三篇 信息存贮的组织与技术

第八章 信息存贮.....	(1)
第一节 引论.....	(1)
第二节 信息结构.....	(3)
第三节 数据结构的存贮与处置.....	(9)
第九章 文件组织.....	(57)
第一节 文件组织简介.....	(57)
第二节 顺序文件.....	(59)
第三节 随机处理文件.....	(61)
第四节 索引链接文件和倒排文件.....	(66)
第五节 B 树文件的概念.....	(67)
第六节 文件系统.....	(69)
第十章 数据库系统.....	(77)
第一节 数据库系统的特点.....	(77)
第二节 数据库的模型.....	(79)
第三节 数据库管理系统.....	(81)
第四节 数据库的应用系统.....	(85)
习 题.....	(94)

第四篇 信息系统的研制与管理

第十一章 信息系统的功能.....	(97)
-------------------	--------

第一节	国外在管理中使用计算机的历史及 经验.....	(97)
第二节	在管理中应用的信息系统.....	(106)
第三节	信息系统对组织的影响.....	(121)
第十二章	信息系统的研制方法.....	(128)
第一节	系统研制概述.....	(128)
第二节	系统分析.....	(136)
第三节	系统设计.....	(162)
第四节	系统实施.....	(187)
第五节	系统的维护与评价.....	(198)
第六节	项目管理.....	(200)
第七节	原型设计方法.....	(216)
第十三章	信息系统的管理.....	(221)
第一节	实现信息化过程中的管理策略.....	(221)
第二节	规划提供信息的技术方式.....	(225)
第三节	硬件和软件发展计划.....	(230)
第四节	信息中心的管理.....	(242)
第五节	资源使用的管理方法.....	(247)
习 题		(252)
参考文献		(257)

第三篇 信息存贮的组织与技术

信息存贮是信息系统中重要的一环，它是信息收集与提供间的纽带。

对于信息存贮的主要要求是资料安全、查找迅速、更新方便、空间节省。为此，必须对信息进行科学的、合理的组织，即把信息组织成一定的信息结构（逻辑结构），然后根据不同的查找和更新需要，采用一种或多种存贮方法来组织对它们的物理存贮，并解决在各种组织中进行处置的算法。

本篇共分三章。第八章，首先介绍信息结构的三种主要模型，它们是组织信息存贮的基础。在该章的其余部分，介绍几种基本的数据（信息）结构：线性表列、栈、队、链表、树等，主要介绍这些数据结构的概念、具体的存贮方式和在结构中进行的常用处置的算法。第九章，介绍文件组织，具体介绍利用外存设备进行存贮的各类文件的组织和存贮方式及相应的更新和检索方法，并对文件系统这一负责存取和管理文件信息的软件机构进行简单介绍。第十章，对数据库的组织进行简单介绍，包括数据库的主要特点、主要数据模型及它的管理系统和应用系统概况。

第八章 信息存贮

第一节 引 论

存贮在现代信息系统中，是十分重要和复杂的一环。

随着信息量的增加，需要存贮的信息越来越多，对信息存贮的要求也越来越高。首先，存贮的资料是否安全可靠，必须认真考虑。对各种自然的、技术的及社会的因素可能造成的资料毁坏或丢失，都必须有相应的处理和防范措施。其次，对于大量资料的存贮，如何节约存贮空间也是需要研究的，例如，采用科学的编码体系，以缩短相同信息所需的代码。第三，信息存贮必须满足存取方便、迅速的需要，否则就会给信息处理带来困难甚至失败。因此，信息存贮不仅是为了保存信息以便再用，而且是信息收集与提供间的纽带。

对于存贮的信息，需要进行各种各样的查找，如按某个数据项查或按许多组合条件查。对于查找的响应时间也有不同的要求，如实时处理中，往往需要立即回答。这就要求对存贮的信息进行各种科学的组织。此外，对资料进行更新也是经常性的工作，例如，增添新资料、删除无用资料、修改变化了的资料等等。信息存贮必须考虑更新是否方便，这也取决于对资料存贮的组织。

所以，存贮的关键问题是如何科学地、合理地组织数据。在一个系统中，各种信息间存在着相互联系和制约的逻辑关系，各种信息本身和它们间的逻辑关系就构成一定的逻辑结构。它是信息处理的依据，也是组织存贮的基础。这就是说，存贮的不只是资料本身，还要存贮它们之间的关系。由此可见，在存贮信息之前，必须对系统中的各种信息进行全面分析，确定系统总体及各个局部的逻辑结构，得到一个逻辑结构的“模型”。有了具体模型之后，还要考虑将来的处理要求，如主要操作是检索还是更新，然后再具体组织物理存贮。

此外，在取定了一种结构和存贮方式之后，对信息进行有关处置的方法也可能不同。本章将介绍信息结构模型、存贮组织及在某个特定的结构中如何进行有关的处置。

第二节 信息结构

对信息结构的研究是现代信息系统中组织存贮的基础。信息结构是对信息之间有机联系的科学描述。它包括信息结构要素与逻辑关系两个方面。

一、信息结构要素

(一) 项 (基本项)

项是对客观事物某一特征的描述。项有名和值之分。项名是项目的标识或说明, 项值则是信息的实体。项名可看作一个变量, 项值则是变量所取的一个值, 取值范围则称为项目的值域。

表8·1是一个简化了的职工工资表。其中, 职工号、基本工资、奖金、托儿费、水费、电费、实发工资都是基本项的名。对某一具体职工, 如李民, 在表中填入的各项值就是相应的基本项之值。

表 8·1 职工工资表

职 工 号	姓 名	应 发 金 额		扣 除			实 发 工 资
		基本工资	奖 金	托 儿 费	水 电 费		
					水费	电费	
0031	李民	106.0	2.5	10.5	0.8	3.5	93.7

项是由一个或一个以上的字符组成的。但在多个字符组成的项中, 只有这些字符的集合才在整体上代表了一定的逻辑含义, 其中的任意一个字符都不能单独表现其含义, 因而, 字符只能作

为信息的最小物理单位，而逻辑上的最小单位是项。所以说基本项是具有一定逻辑含义的、有名字的、最小数据单位，是最基本的结构要素。

（二）组项

组项是由基本项或组项组成的。例如，表 8·1 中的应发金额就是由基本工资、奖金这两个基本项构成的组项，而扣除是由基本项托儿费和组项水电费组成的组项。显然，组项不是最小的数据单位，它可以分解成若干基本项。

基本项和组项都是用来描述实体的某种属性的。

（三）记录

记录是由若干相关的项或组项组成的数据单位，它是关于一个实体在某个处理问题中数据的总和。如表 8·1 中，所有基本项和组项就构成一个职工工资记录。记录有型和值之分。记录型是组成该记录的各项的项名的有序集合。记录值由与记录型相对应的项值组成，例如，表 8·1 中，工资表的一个具体行就是一个记录值。记录值是信息的实体，通常一个记录值可以代表一个事物。例如，一个职工工资记录值就代表了某个职工有关工资方面的全部信息。记录的型只是对记录组成的一种说明。

组成记录的项目如何选择是信息结构和信息存贮组织中的一个重要问题。因为在数据处理和信息存贮组织中通常都以记录为单位。将经常需要联系在一起使用的项目组成一个记录，对信息存贮的更新，查找是有利的。显然，对单一的使用目的，记录的组成是很容易考虑的。但当同一类信息要服务于多个使用目的时，则应考虑尽可能地减少冗余，实现共享。例如，在一个系统中，同一批职工有人事记录、工资记录、业务记录、健康记录等，分别服务于各种检索和处理需要。但这些记录中都包含了一些相同数据，如职工姓名、性别、出生年月、工作单位、职工号等。因此可以把这些项单独构成一个基本情况记录，其它的记

录,如健康记录就可只包括职工号、身高、体重、血压、病史等项目。当要了解一个完整的健康记录时,可以通过职工号查找相应的基本情况记录即可。这样,就可以使各种具体应用目的的记录共享基本情况记录。如果各种记录的逻辑顺序是一一对应的,则职工号也可以只放在基本记录中,在其它各具体记录中就不必重复了。

(四) 文件 (记录群)

同一记录型的若干记录值的集合称为文件。例如表 8·1 所示工资表中各行所组成的整体或者说整个工资表就是某单位职工工资文件。

文件是数据组织和存贮的基本形式,根据不同的使用目的和操作方式,文件组织有各种不同形式。第九章将具体论述。

以上对文件的论述是从数据的逻辑关系上来进行的。在计算机科学中,文件组织有时也由数据存贮的物理介质来进行分类,如内存文件、磁带文件、磁盘文件、卡片文件、打印文件等等。

(五) 数据库

数据库是由若干文件组成的信息集合。数据库中的记录型通常有多种,而且彼此之间有一定联系,以便满足一个信息系统所需的信息。数据库是一个信息系统中信息结构的最高层。

二、逻辑结构

信息结构要素之间的逻辑关系,通常是记录间的逻辑关系。这种关系又分两类:

- 1.同类记录值间的关系,又称内部逻辑关系;
- 2.不同型记录值间的关系,又称外部逻辑关系。

无论是同类记录值间还是异型记录值之间,它们的逻辑关系都可以概括为以下三种结构。

(一) 线性结构

在这种结构中,信息要素按序逐个排列,如图8·1所示。

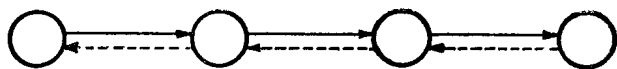


图 8.1 线性关系

图中，圆圈表示结构要素（称为结点），实线箭头表示正向关系，虚线箭头表示逆向关系。例如，在一个学校中，有教师记录、职工记录、学生记录，可以规定一个顺序，先教师，后职工，最后学生，这是一种顺序关系，是不同型记录间的外部逻辑关系。同样，对于一个类型的记录，如学生记录可以按学号升序排列，这就是记录间的内部逻辑关系，也是一种顺序关系。

线性结构是一种最简单的结构，它的正关系和逆关系都是一对一的。

（二）树型结构

树型结构反映的关系是客观世界中十分普遍的一种层次关系，如图8.2所示。

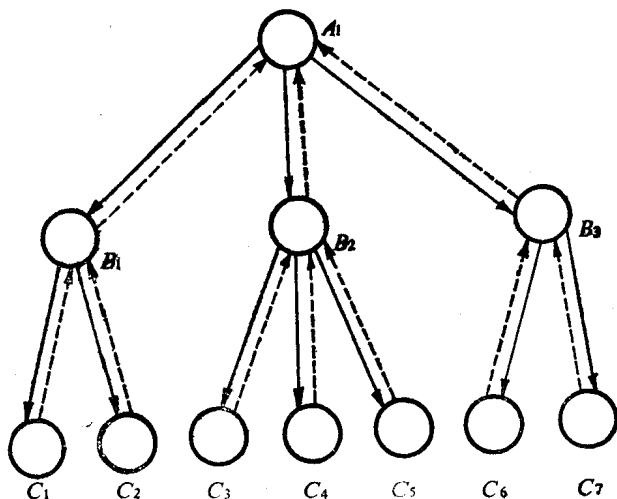


图 8.2 树型关系

树型结构的正关系是一对多的，即一个上层元素之下可以有多个下级的元素。而逆关系则是一对一的，即一个下级元素只有一个直接上级元素。图8·2中，如果 A 、 B 、 C 可以代表三种记录型，设 A_1 代表信息系， B_1 、 B_2 、 B_3 分别代表该系所属的教研室， C_1 、 C_2 是属于 B_1 教研室的教师， C_3 、 C_4 、 C_5 是属于 B_2 教研室的教师， C_6 、 C_7 是属于 B_3 教研室的教师，则图8·2代表了记录间的外部逻辑关系。如果把图8·2中的各结点看成是同型的记录，例如是十一种机械零件记录，而 C_1 、 C_2 可以组成 B_1 ， C_3 、 C_4 、 C_5 可以组成 B_2 ， C_6 、 C_7 组合成 B_3 ， B_1 、 B_2 、 B_3 又组合成 A_1 ，则图8·2的树型关系就反映了记录的内部逻辑关系。

(三) 网型结构

网型结构即信息结构要素之间形成相互交叉的网状关系，如图8·3所示。

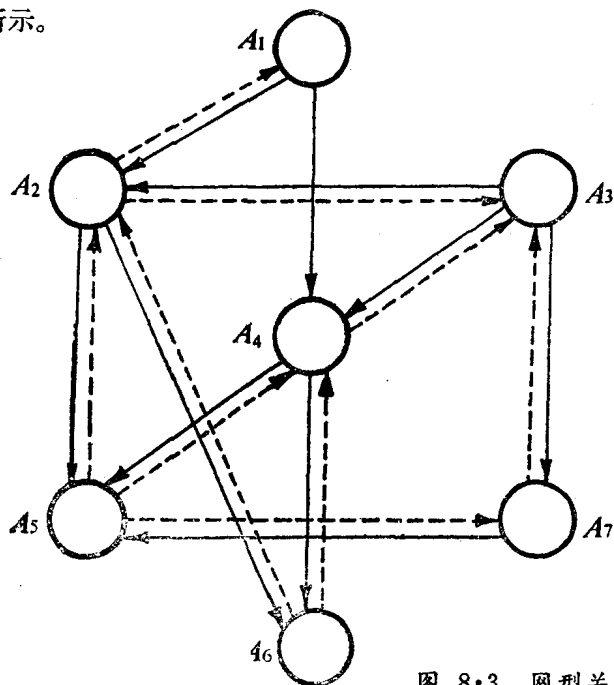


图 8·3 网型关系

网型结构的正关系是一对多的，即一个元素可以与多个下级元素发生直接关系。逆关系，也是一对多的，即下级元素同时直接与多个上级元素有关。

图8·4给出一个网型结构的示例。其中有三种记录：订单、订货明细及产品记录。

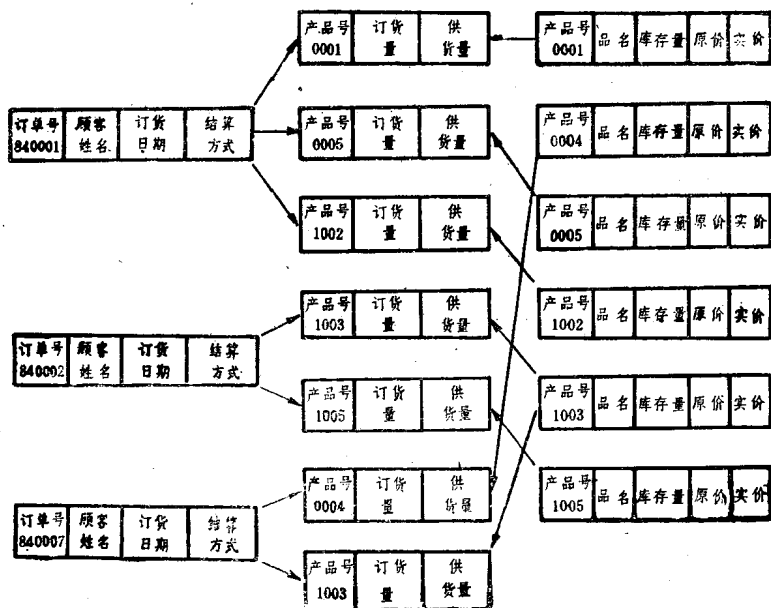


图 8·4 网型结构示例

图中表明一个顾客的一个订单可以订多种产品，而一种产品又可满足多个顾客的要求（如1003号产品同时提供给两个不同的顾客）。

对于同类型的记录，可仍以零部件记录为例，若一种零部件由几种零部件组成，而一个零部件又可为多个其它零部件所使用，则零部件间的内部逻辑关系就是网型结构。如图8·3中，若每一个结点都代表一个零部件记录，则 A_1 由 A_2 、 A_4 组成， A_2 、 A_4 又

同时为 A_3 所使用, A_5 、 A_6 同时为 A_2 、 A_4 所使用……, 形成错综复杂的网状关系。

第三节 数据结构的存贮与处置

数据结构是对数据进行组织的形式。

为了更一般地研究数据结构的存贮和在结构内部进行各种处置的算法, 不妨把一批数据看成是一个数据空间, 而一个具体的数据结构元素是这个空间中的一个点, 称之为结点。而数据间的关系也就是结点间的关系。这样, 就可以把数据结构定义为: 结点的有限集合和结点间关系的有限集合。

上述定义中的结点, 实际上可以是一个数据项, 也可以是一个记录, 甚至是一个文件。为了便于学习数据结构的基本内容, 本节在论述中将不涉及结点的类型及其内部细节。而用一批同类型的数据代表所研究数据结构的结构元素, 每一个数据就是一个结点。同时为便于论述各种处置方法, 假设这批数据都存在内存的一维数组之中, 并可利用数组元素的下标来描述某种数据关系。为了讨论更复杂的关系, 也将数据间的关系集合中的各元素再存贮在数组中。

以上的假设, 只是为了学习上的方便, 丝毫不意味着本章所论述的内容只适用于内存, 也不说明必须用数组存贮结点或结点间关系。

一、线性表列

设有 N 个结点的集合 $a_1, a_2, \dots, a_n$, 结点间存在着唯一的的关系, 即对于任何一个结点 a_i ($2 \leq i \leq n$), 都有 a_{i-1} 为 a_i 的前缀结点, 而 a_i 为 a_{i-1} 的后继结点, 则称这个数据结构为线性表列。在线性表列中, 只有一个结点 a_1 没有前缀, 称为起始结点, 有唯

一的结点 a_n 没有后继，称之为终止结点。

线性表列是简单而常用的一种数据结构，和第一篇介绍的一维数组十分类似，只是这里相当于数组元素的“结点”，有更为广泛的含义罢了。

线性表列以及它的特例——栈和队，都有顺序存贮和链接存贮两种存贮方式，本节只介绍顺序存贮的情况。

设线性表列中，每个结点都占用一个存贮单元，用数组 $A[1:M]$ ($M \geq N$)，表示 M 个相继的存贮单元，在前 N 个单元中按顺序存放着表列的 N 个结点。

在顺序存贮的线性表列中有以下几种常用的处置。

1. 求出线性表列中的第 I 个结点。

```
IF  $I < 1$  OR  $I > N$  THEN
```

```
    OUTPUT ('ERR0')
```

```
ELSE OUTPUT (A [ $I$ ])
```

```
END-IF
```

2. 求第 J 个结点的前缀和后继结点。

求前缀：

```
IF  $J \leq 1$  OR  $J > N$  THEN
```

```
    OUTPUT ('ERR0')
```

```
ELSE
```

```
    OUTPUT (D [ $J-1$ ])
```

```
END-IF
```

求后继：

```
IF  $J \geq N$  OR  $J < 1$  THEN
```

```
    OUTPUT ('ERR0')
```

```
ELSE OUTPUT (D [ $J+1$ ])
```

```
END-IF
```

3. 从表列中删去第 J 个结点，删除后仍保持线性表列的顺序

关系。

```
IF J<1 OR J>N THEN
    OUTPUT ('ERR0')
ELSE
    FOR I FROM J TO N-1 BY 1 DO
        A [I]←A [I+1]
    END—FOR
    N←N-1
END—IF
```

4.在第J个结点之后增加一个新结点,并保持线性表列的顺序关系(设新结点值存放在简单变量NEW中)。

```
IF N+1>M THEN
    OUTPUT ('ERR0')
END—IF
IF J<0 OR J>N THEN
    OUTPUT ('ERR0')
ELSE
    FOR I FROM N TO J+1 BY -1 DO
        A [I+1]←A [I]
    END—FOR
    A [J+1]←NEW
    N←N+1
END—IF
```

以上几种处置,由于比较简单,因此只给出程序段,没有写成完整的程序。

顺序存贮的线性表列的优点如下:

(1) 存贮密度高,因为结点的逻辑顺序和物理顺序一致,无须另外占用空间来存贮结点间的关系。

(2) 对结构中的任一结点都可以用相同的速度进行访问。

因此称之为随机存取的数据结构。

顺序存贮的线性表列的缺点是增添和删除比较困难，因为增添、删除之后，要保持顺序，就要对若干结点进行移址，这在结点比较多、存贮量比较大时，是相当费时间的。

(一) 栈

1. 栈的定义及其基本处置。如果对一个线性表列进行增添或删除时，只允许在表列的一端进行，则称这个线性表列是一个栈。在软件和数据处理工作中，特别是进行动态处理时，经常要用到栈结构。

把对栈进行处理的一端称为栈顶。由于增添和删除都在栈顶进行，所以栈顶的位置是动态变化的，需用一个存贮单元作为指针，来指明栈顶的当前位置。向栈里增添一个新结点，常常称为让结点进栈，反之，从栈顶删除一个结点称为让结点出栈。可用图8·5表示一个栈的进、出情况。其中A、B、C、D、E均代表结点，P代表栈顶指针，假设这个栈最多可存四个结点。

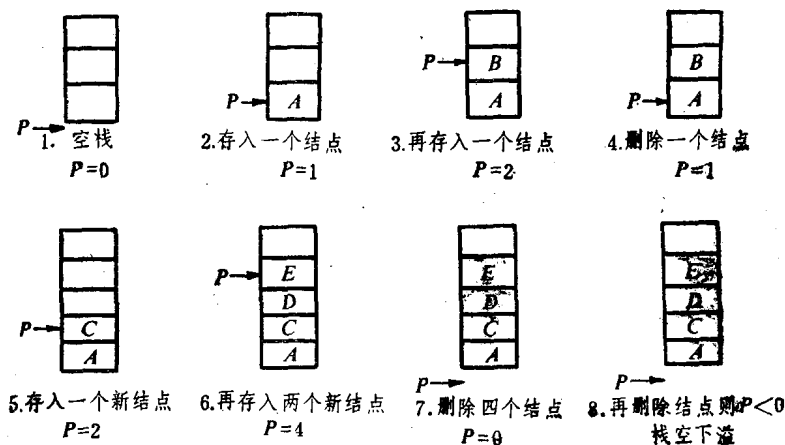


图 8·5