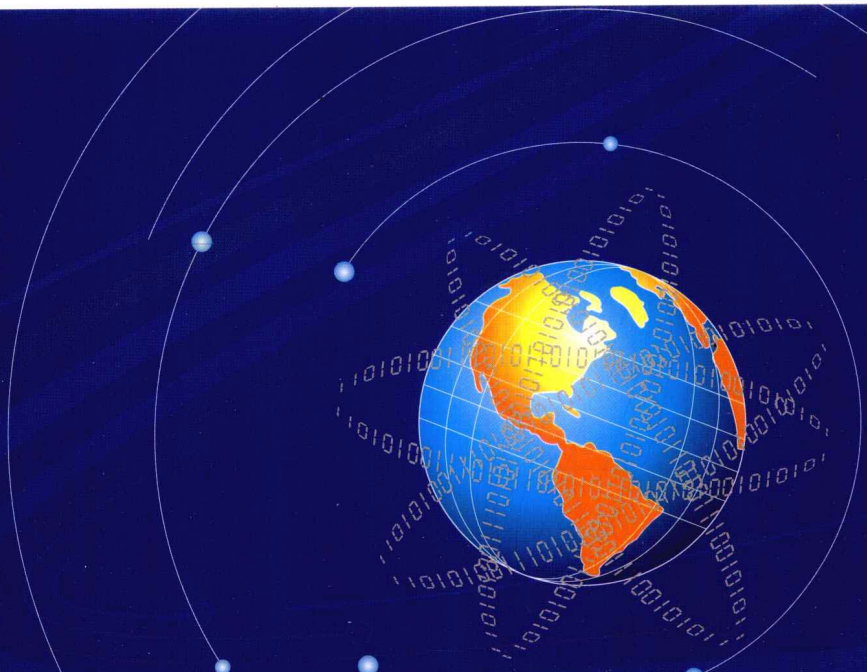
高等学校计算机类规划教材

- 详细讲述数据在计算机中的存储结构及操作实现
- 采用C语言作为数据结构和算法的描述工具
- 理论与实践相结合，内容组织以应用为主线
- 免费提供本书PPT电子课件及习题解答

数据结构

张　青　主　编
杨族桥　何中林　副主编

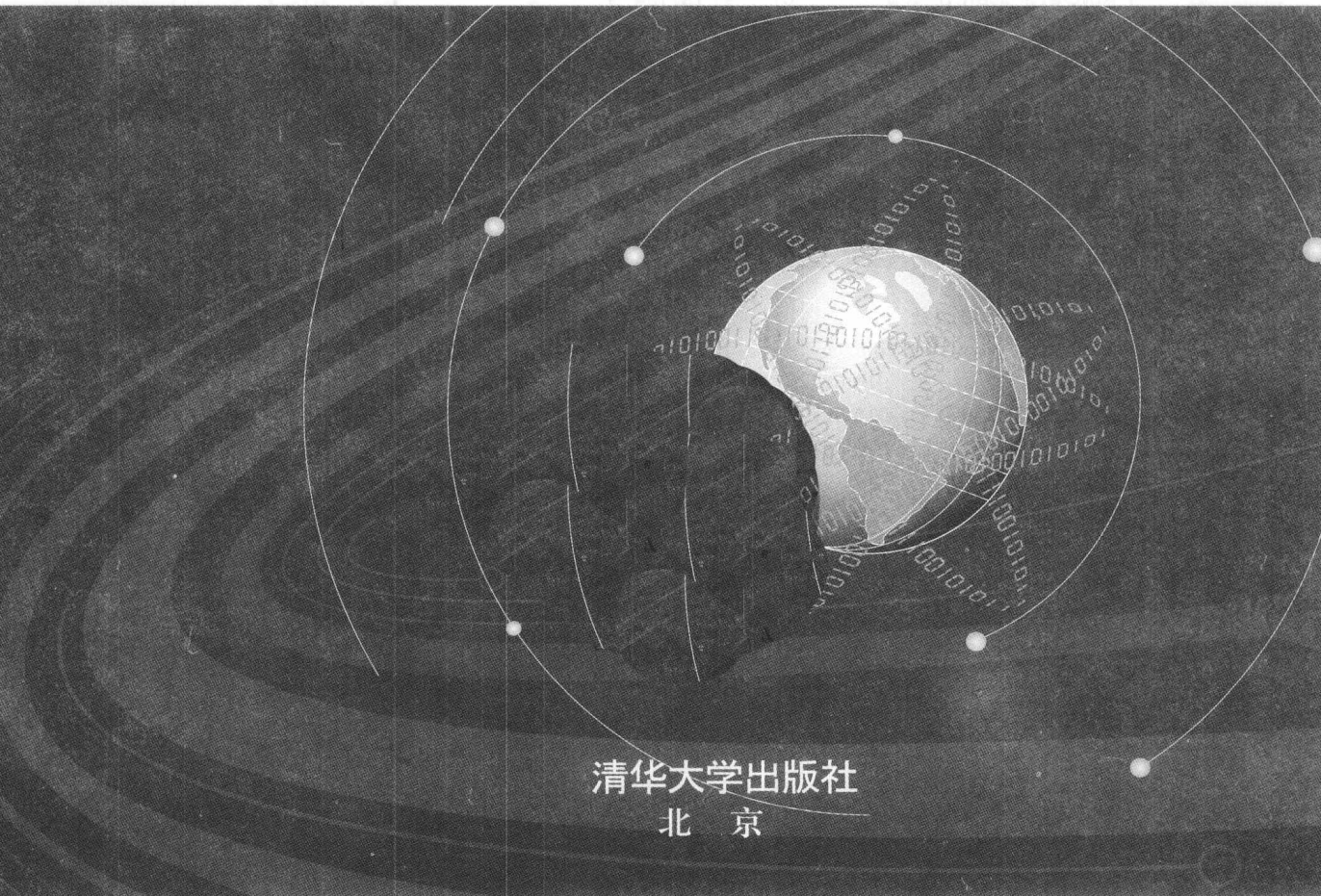


清华大学出版社

高等学校计算机类规划教材

数据结构

张　青　主　编
杨族桥　何中林　副主编



清华大学出版社
北　京

内 容 简 介

本书首先介绍了数据结构的概念及其逻辑结构、存储结构及运算3方面内容涉及的基本概念，然后针对经典的数据结构（即线性表、栈、队列、多维数组、广义表、树和图）的逻辑特征、常用的存储方式及各种基本运算的实现算法作了详细阐述，最后讨论了两种典型运算——排序和查找的各种实现方法。全书采用C语言作为数据结构和算法的描述工具。在一些重点部分，还给出了简单应用举例的完整C语言程序。

本书结构清晰、层次分明、深入浅出、通俗易懂、适用面广，可以作为普通高等院校计算机和信息类本科或专科教材，也可以作为其他理工类专业的选修课教材。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。
版权所有，侵权必究。侵权举报电话：010-62782989　13701121933

图书在版编目（CIP）数据

数据结构/张青主编. —北京：清华大学出版社，2011.9

ISBN 978-7-302-26851-2

I. ①数… II. ①张… III. ①数据结构 IV. ①TP311.12

中国版本图书馆CIP数据核字（2011）第187154号

责任编辑：朱英彪
封面设计：张　岩
版式设计：文森时代
责任校对：张彩凤
责任印制：何　芊
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
社　总　机：010-62770175　　**邮　购**：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质　量　反　馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 装 者：北京国马印刷厂
经　　销：全国新华书店
开　　本：185×260　**印　张**：15.5　**字　数**：355千字
版　　次：2011年9月第1版　　**印　　次**：2011年9月第1次印刷
印　　数：1～4000
定　　价：29.80元

产品编号：043689-01

前　言

“数据结构”是计算机及信息类相关专业的专业基础课程和核心课程。随着计算机在各个学科领域内的深入应用，本课程也成为其他很多专业的热门选修课程。“数据结构”所研究的知识内容和技术方法，无论对学习计算机学科的相关课程，还是对从事软件设计和开发工作，都非常重要，是重要的理论基础。

本书主要针对各种数据之间的经典逻辑结构及特点，讨论了数据在计算机中的存储结构及操作实现。通过本书的学习，读者应能够熟练掌握线性表、树、图的逻辑特征，并能根据不同的逻辑关系，采用适当的存储结构，对各种典型运算设计相应的算法；在深入理解和掌握本书内容的基础上，培养自己进行复杂程序设计的能力以及运用基本理论和知识解决实际问题的能力。

本书包含 8 章内容。第 1 章介绍了数据结构和算法等基本概念；第 2 章和第 3 章介绍了 3 种最基本的线性结构，即线性表、栈和队列；第 4 章讨论了数组、广义表及字符串；第 5 章和第 6 章分别讨论了树和图；第 7 章和第 8 章研究了数据处理过程中的两种常用的操作——排序和查找。鉴于目前“C 语言程序设计”已经普遍地成为数据结构的先修课，全书采用 C 语言作为数据结构和算法的描述工具。本书利用数组、结构体和指针等重要数据类型结合 C 语言函数，完成了书中所有基本运算的实现算法，同时，对一些常用的算法和重点的操作，还给出了完整的 C 语言程序，旨在引导学生利用数据结构中的基本运算来解决实际问题。书中所有的算法都经过上机调试通过。

本书在内容选取上符合复合型、应用型人才培养目标的要求，组织编排上体现了理论与实践相结合、突出应用的原则，遵循教学和认知规律。注重课程内容的前后呼应，理清知识的来龙去脉，强调条理性和系统性，兼顾学科的广度和深度。本书结构清晰，层次分明，深入浅出，通俗易懂，适用面广，可以作为普通高等院校计算机和信息类专业本科和专科教材，也可以作为其他理工类专业的选修教材。建议讲授学时为 48～72 学时，教师可以按照教学大纲要求，根据各专业特点，合理安排学时及授课内容。

本书的编者一直从事数据结构的教学和研究工作，曾参与编写多部教材。本书是编者多年教学经验的积累，在重、难点内容的叙述及讲解方法上有一定的特色。本书的编写安排如下：张青教授负责全书的整体策划、统稿及第 1 章的编写；第 2 章由杨族桥编写；第 3 章由朱泽民编写；第 4 章由高汉平编写；第 5 章由肖飞编写；第 6 章由张敏编写；第 7 章由杨旭华编写；第 8 章由涂焱楚编写。王锋教授对本书的编写提出了许多宝贵的意见和建议，在此表示感谢。编写过程中参考了大量相关的著作、教材和网络资料，在此一并表示感谢。

由于编者水平有限，书中难免有疏漏或不妥之处，敬请广大同行专家和读者不吝指正，不胜感激。

编　者

2011 年 7 月

目　录

此为试读，需要完整PDF请访问：www.ertongbook.com

第1章 绪 论

本章主要介绍数据结构中的一些常用术语、线性存储结构和链式存储结构等常用数据结构的表示以及C语言描述算法的基础、算法的时间和空间复杂度的分析等内容。

学习要求

1. 重点掌握的核心知识点

(1) 数据结构的基本概念。
(2) 数据的逻辑结构、存储结构。
(3) 算法的概念及特点。
(4) 算法的时间复杂度分析。

2. 一般掌握的知识点

(1) 抽象数据类型（ADT）的定义与使用。
(2) 算法的空间复杂度分析。

1.1 数据结构的概念和基本术语

1.1.1 数据结构简介

信息技术的发展，为计算机提供了更为广阔的应用空间，使其不再局限于科学计算方面的应用，而是广泛应用于控制、管理和数据处理等非数值计算的处理工作中。与之相应，计算机处理的对象也由纯粹的数值数据发展到字符、图、表和文本等一些具有结构的数据，因此非常有必要去研究这些带有结构的数据对象及其相关的操作方法。

数据结构主要研究数据对象的关系及其在计算机中的表示和操作。程序设计中，对不同逻辑结构的数据，应选择适合其特点的存储结构和相应的操作算法。

从提出一个实际问题到计算机解出答案一般需要以下步骤：

（1）从实际问题中抽象出一个数学模型。

（2）设计解此数学模型的算法。

（3）编制程序，进行调试、测试，直到得出正确的结果。

在进行项目开发和程序设计的过程中，许多问题都需要只抽象出一些模型，然后针对这些模型来解决问题。下面来看具体示例。

例 1-1 学生成绩管理。

学生成绩以二维表的形式存入计算机后，对学生信息的处理工作就转化为对表的处理。

如查询学生成绩的操作是在表中查询基本信息，增加学生成绩记录的操作是在表中增加一行基本信息，删除学生成绩记录的操作是删除表中相应的行，修改学生成绩记录的操作是对表中相应行的修改。因此，由计算机实现学生成绩管理问题的抽象模型是：

（1）建立包含每个学生成绩信息的表。

（2）对该表进行查询、插入、删除和修改等操作（也称运算）。

这个二维表格中每一行元素之间的关系是一种线性关系，因此是一种线性表，如表 1.1 所示。

表 1.1　学生成绩表

姓　名	学　号	成　绩
李丹丹	2010026240101	95
马小丽	2010026240102	88
吴志平	2010026240103	93
…	…	…

例 1-2　计算表达式 (pow(1.2,4)+sin(1.125))/((3+8)×12)的值。

当通过键盘输入上述算术表达式后，要求计算机能给出正确的计算结果。为了能够正确地完成计算，必须要求计算机能够识别上述表达式是一个包含加法（+）、减法（-）、乘法（×）、除法（/）和函数的算术表达式，从而达到对输入串的计算类型的识别，进而根据不同的计算类型做相应的处理。因此，需将表达式(pow(1.2,4)+sin(1.125))/((3+8)×12)分成两个独立的子表达式(pow(1.2,4)+sin(1.125))和((3+8)×12)，只要分别计算出了二者的值，那么表达式(pow(1.2,4)+sin(1.125))/((3+8)×12)的值也就很容易计算出来了。上述表达式的计算过程如图 1.1 所示。

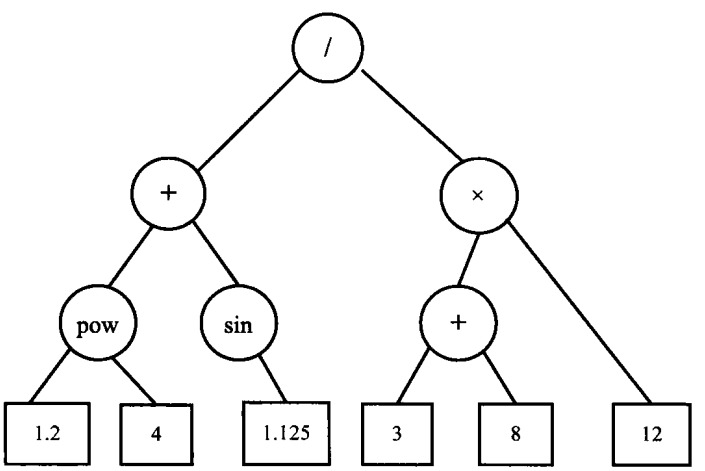


图 1.1　表达式的计算过程

例 1-2 中表达式的计算过程用一种图示的方式表示了出来，计算顺序为先左后右、自底向上逐步进行。而且从形状看，它和一棵倒置树的结构极其相似。计算过程也是类似的，底层计算的结果传给上一层结点，直至到达最上面的结点（称为树根，root），得到计算结

此为试读，需要完整PDF请访问：www.ertongbook.com

果。图 1.1 表示的是一种被称作树的数据结构，它形象地表示了计算结点之间的逻辑关系。

例 1-3　排课问题。

每名学生在入学后都要学习一系列的课程，有的课程需要另一门课程作为先行课。下面以软件专业为例列出课程关系表，如表 1.2 所示。在 12 门课程中，有基础课和专业课，怎样安排才能合理呢？当然要将一门课程安排在其先行课之后，这是一种特殊的排序问题。图 1.2 所描述的数据模型是表 1.2 所列出的课程关系的数据模型，它表示的是一种被称为图的数据结构。

表 1.2　课程关系表

课 程 编 号	课 程 名 称	先　行　课
C_1	程序设计基础	
C_2	离散数学	C_1
C_3	数据结构	C_1，C_2
C_4	汇编语言	C_1
C_5	语言设计和分析	C_3，C_4
C_6	计算机原理	C_{11}
C_7	编译原理	C_3，C_5
C_8	操作系统	C_3，C_6
C_9	高等数学	
C_{10}	线性代数	C_9
C_{11}	普通物理	C_9
C_{12}	数值分析	C_1，C_9，C_{10}

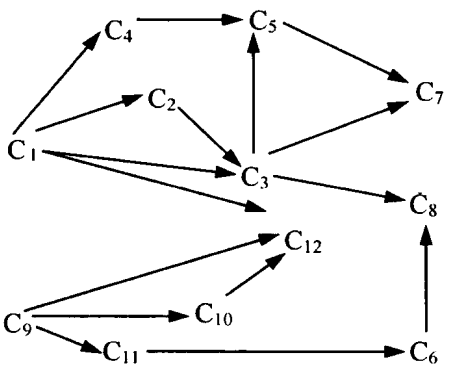


图 1.2　课程关系图

综上 3 个示例可见，描述这类非数值计算问题的数学模型不再是数学方程，而是如表、树、图之类的数据结构。

1.1.2　基本术语

1. 数据（Data）

数据是信息的载体，是所有能输入到计算机中并被计算机识别、存储和处理的符号的

集合。数据就是计算机化的信息，是一种计算机符号的特定表示形式。早期计算机主要应用于数值计算，数据量小且结构简单，数据只包括整型、实型和布尔型，当时程序工作者把主要精力放在程序设计的技巧上，而并不重视组织数据。随着计算机软硬件的发展和应用领域的不断扩大，非数值运算处理在计算机应用领域所占的比例越来越大。现在的数据不单纯是数值，还包含文本、声音和图像等许多非数值数据。多种信息通过编码而被归于数据的范畴，在大量复杂的非数值数据处理过程中，数据的组织显得越来越重要。

2．数据元素（Data Element）

数据元素简称元素，是数据的基本单位，是数据集合的个体，在计算机中通常作为一个整体进行考虑和处理。对于二维表格，每行信息就是它的一个数据元素。每个数据元素由若干个数据项组成，此时的数据元素通常称为记录（Record）。如表 1.1 中的数据，每一个学生的记录就是一个数据元素；对于一个字符串数据来说，每个字符就是它的数据元素；对于一维数组来说，每个下标所对应存储单元的值就是它的数据元素。

3．数据项（Data Item）

数据项是数据不可分割的最小单位。如表 1.1 中学生的姓名、学号和成绩。

4．数据对象（Data Object）

数据对象是性质相同的一组数据元素的集合，是数据的一个子集。如{2,4,6,8,10}是整型数据的一个数据对象，在该数据对象中，每一个数据元素都是性质相同的整数，所以该数据对象是有限个整数数据元素的集合，是整数的一个子集。

5．数据处理（Data Processing）

数据处理是指利用计算机对数据进行储存、检索、插入、删除、统计、排序、输入和输出等的处理过程。

6．数据结构（Data Structure）

数据结构是数据及其相互之间的关系。如表 1.1 中数据是学生记录的集合，数据元素之间的关系就是它们在学生成绩表中的前驱和后继关系，这种关系是一对一的，是一种线性结构。数据元素相互之间的关系称为结构（Structure）。在例 1-3 中描述的数据元素集合是课程的集合，而数据元素之间的关系就是课程之间学习的先后次序关系，这种关系是多对多的，是一种图结构。

数据结构的形式化定义为：

```
Data_Structure=(D,S)
```

其中，D 是数据元素的集合；S 是 D 上关系的有限集。

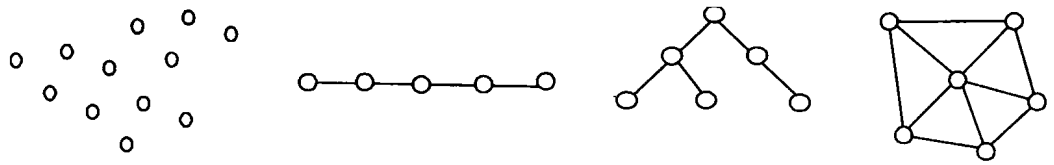
根据数据元素之间关系的不同特性，数据结构通常分为以下 4 类，如图 1.3 所示。

（1）集合结构：结构中的数据元素之间除了同属于一个集合的关系外，别无其他关系。

（2）线性结构：结构中的数据元素之间存在着一对一的线性关系。

（3）树状结构：结构中的数据元素之间存在着一对多的层次关系。

（4）图状结构（或网状结构）：结构中的数据元素之间存在着多对多的任意关系。



（a）集合结构　（b）线性结构　（c）树状结构　（d）网状结构

图 1.3　4 类基本的数据结构

例 1-4　有一个研究小组，其中有一位指导教师和若干学生，指导老师可以带 1～2 名研究生，每名研究生可以带 1～2 名本科生。则研究小组的数据结构可以设计为：

```
R_Group=(P,R)
```

其中，P 表示数据对象；R 表示数据对象中数据元素之间的关系。

P、R 的描述如下：

$P=\{T,G_1,\ldots,G_n,S_{11},\ldots,S_{nm}\}$　$(1\leqslant n\leqslant 2,1\leqslant m\leqslant 2)$

$R=\{R_1,R_2\}$

$R_1=\{<T,G_i>|1\leqslant i\leqslant n,1\leqslant n\leqslant 2\}$

$R_2=\{<G_i,S_{ij}>|1\leqslant i\leqslant n,1\leqslant j\leqslant m,1\leqslant n\leqslant 2,1\leqslant m\leqslant 2\}$

其中，T 表示指导教师；G 表示研究生；S 表示本科生。

例 1-4 中，R_Group 定义的数据结构是对操作对象抽象出来的数学描述，其中，“关系”是描述数据元素之间的逻辑关系，因此称为数据的逻辑结构。与逻辑结构相对应的是数据的物理结构，又称为数据的存储结构，是指数据的逻辑结构在计算机中的映像，即数据结构在计算机中的存储方式。

由于数据结构包括数据元素集合及数据元素之间的关系，所以数据的存储结构也应该包括数据元素的映像和元素间关系的映像两部分内容。数据元素之间的关系在计算机中有两种不同的表示方法——顺序映像和非顺序映像，并由此得到两种不同的存储结构即顺序存储结构和链式存储结构。顺序存储结构是把数据元素存储在一块连续地址空间的内存中，其特点是逻辑上相邻的数据元素在物理上也相邻，数据间的逻辑关系表现在数据元素的存储位置上，实现数据存储结构的方法是使用数组；链式存储结构的特点是借助指示元素存储地址的指针（Pointer）表示数据元素之间的逻辑关系。

任何一个程序都涉及数据的存储结构。因为对于逻辑结构上的同一种运算，其具体的实现方法依存储结构的不同而变化。所以，当描述问题的模型确定之后，首要问题就是确定其存储结构。

7．数据类型（Data Type）

数据类型是一个值的集合和定义在这个值集上的操作集合，是数据的取值范围、数据元素之间的结构以及允许进行的操作的总体描述。数据类型中定义了两部分内容，即某种数据的取值范围和该类型可允许使用的一组运算。高级语言中的整型、实型、字符型就是数据类型的实例。在 C 语言中，整型类型（int）可能的取值范围是-32768～+32767，定义了加（+）、减（–）、乘（×）、除（ / ）和取模（%）等运算。从硬件的角度来看，它们的

实现涉及字、字节、位和位运算等；从用户的观点来看，并不需要了解整数在计算机内是如何表示、运算细节是如何实现的，只需要了解整数运算的外部运算特性，就可运用高级语言进行程序设计。引入数据类型的目的，从硬件的角度，是将其作为解释计算机内存中信息含义的一种手段；对使用数据类型的用户来说，则实现了信息隐蔽，将一切用户不必关心的细节封装在数据类型中。如两整数求和问题，用户只需注重其数学求和的抽象特性，而不必关心加法运算涉及的内部位运算实现。

高级程序设计语言中的数据类型可分为两大类：一类是原子类型。原子类型的值是不可分解的，如 C 语言中的整型、实型、枚举值、逻辑和字符等基本类型；另一类是结构类型。结构类型的值是按某种结构由若干成分组成的，因此是可以分解的，其成分可以是原子类型，也可以是结构类型。如数组、结构体类型的值由若干分量组成，每个分量可以是整数，也可以是数组等。

8．抽象数据类型（Abstract Data Type，ADT）

抽象数据类型是定义好逻辑关系的数据类型以及该类型之上的一组操作的总称。抽象的本质是抽取事物的本质特性，忽视非本质的细节。计算机中使用的是二进制数，汇编语言中则可给出各种数据的十进制表示，如 98.65、9.6E3 等，它们是二进制数据的抽象，用户在编程时可以直接使用，不必考虑实现细节。在高级语言中，则给出更高一级的数据抽象，出现了数据类型，如整型、实型、字符型等。到抽象数据类型出现，可以进一步定义更高级的数据抽象，如各种表、队、栈、树、图、窗口和管理器等，这种数据抽象的层次为用户提供了更有利的手段，使得用户可以从抽象的概念出发，从整体考虑，然后自顶向下逐步展开，最后得到所需结果。可以这样看，高级语言中提供整型、实型、字符型、记录、文件和指针等多种简单数据类型，可以利用这些类型构造出像栈、队列、树和图等复杂的抽象数据类型。

抽象数据类型的定义取决于客观存在的一组逻辑特性，而与它在计算机内如何表示和实现无关，即无论其内部结构如何变化，一个抽象数据类型定义了一个数据对象、数据对象中各元素间的结构关系，以及一组处理数据的操作。抽象数据类型通常是指由用户定义的用以表示应用问题的数据模型，通常由基本的数据类型组成，并包括一组相关服务操作。

抽象数据类型包括定义和实现两方面，其中定义是独立于实现的。定义仅给出一个抽象数据类型的逻辑特性，不必考虑如何在计算机中实现。抽象数据类型的特征是使用与实现分离，实现封装和信息隐蔽，也就是说，在抽象数据类型设计时，类型的定义与其实现分离。另一方面，抽象数据类型的含义更广，不仅限于各种不同的计算机处理器中已定义并实现的数据类型，还包括设计软件系统时用户自定义的复杂数据类型。所定义的数据类型的抽象层次越高，含有该抽象数据类型的软件复用程度就越高。

抽象数据类型是计算机科学中最重要的概念之一，它集中体现了程序设计中的一些最基本的原则：分解、抽象和信息隐藏。严格地说，可以用代数系统形式定义一个抽象数据类型，可以把抽象数据类型看成是定义了一组运算的数学模型。

抽象数据类型不仅发展了数据抽象的概念，而且将数据抽象和过程抽象结合起来。一

个抽象数据类型确定了一个模型，却将模型的实现细节隐藏起来。它定义了一组运算，却将运算的实现过程隐藏起来。数学模型→抽象数据类型→数据结构，反映了信息结构转换的 3 个重要阶段。

ADT 的定义如下：

```
ADT<抽象数据类型名> {
        Data: <数据对象的定义>
    Relations: <结构关系的定义>
    Operations: <基本操作的定义>
}   ADT<抽象数据类型名>
```

其中，数据对象和结构关系的定义采用数学符号和自然语言描述，而基本操作的定义格式为 C 语言基本函数的定义形式，如下所示：

```
函数类型    函数名([参数列表])
{
        函数体
}
```

参数表中的参数有两种：一种参数只为操作提供待处理数据，又称值参；另一种参数既能为操作提供待处理数据，又能返回操作结果，也称变量参数。操作前提是描述操作执行之前数据结构和参数应满足的条件；操作结果是描述操作执行之后数据结构的变化状况和应该返回的结果。抽象数据类型可用现有计算机语言中已有的数据类型，即固有数据类型来表示和实现。用标准 C 语言表示和实现抽象数据类型描述时，主要包括以下两个方面。

（1）通过结构体将 int、float 等固有类型组合到一起，构成一个结构类型，用 typedef 为该类型或该类型指针重命名。

（2）用 C 语言函数实现各操作。

1.1.3　顺序存储结构

数据常用的存储结构有顺序存储结构和链式存储结构两种。一般来说，可采取其中的一种方式，也可采用二者相结合的方式。

顺序存储结构是指逻辑上相邻的数据元素，其结点的物理位置也相邻，数据元素之间的关系由结点的邻接关系体现。计算机的内存单元是一维结构，这种存储结构非常方便实现。

表 1.3 给出了学生基本信息的逻辑结构，该结构是线性的，各个学生记录前后是相邻的。利用顺序存储结构存储学生基本数据时，可以为学生基本信息表分配一块连续的内存单元。若内存地址单元从 2200 开始，表的第 1 个数据元素就可从 2200 单元开始存放，若一个学生记录需占用 100 个单元，则第 1 个元素就占用了 2200～2299 连续的 100 个单元，从 2300 开始存放第 2 个元素，从 2400 开始存放第 3 个元素……依此类推，直至将表中的元素存放完，如表 1.3 所示。

表 1.3　存储单元

2200	李思维	201026240101	女	湖北黄冈
2300	蒋航	201026240102	男	山东临沂
2400	周凡	201026240103	女	湖北罗田
2500	李勇	201026240104	男	江西南昌
2600	丁宇	201026240105	男	湖北公安
…	…	…	…	…

顺序存储结构的主要特点如下。

（1）结点中只存放数据元素本身的信息，无附加内容。

（2）数据元素的存取操作速度较快。

（3）插入、删除数据元素实现起来较慢，关于该问题将在第 2 章线性表的插入和删除中讨论。

（4）空间利用率低。顺序存储是一种静态结构，连续的存储空间一旦分配完毕，其大小就难以改变。因此，当表中元素个数难以估计时，分配的空间大小也就难以确定。预分配的空间太大，会造成浪费；空间太小，又可能产生溢出现象。

1.1.4　链式存储结构

在链式存储结构中，逻辑上相邻的数据元素，其结点的物理位置不一定相邻，因此，结点之间是否邻接并不能反映数据元素的逻辑顺序。在这种情况下，存储结构要反映数据元素在逻辑结构中的关系，只有在结点中增加信息来表示。增加的信息是一个指针，前一个结点的指针指向后一个结点，多个结点的指针一起形成一个链，因此称这种存储结构为链式存储结构。链式存储结构既可用于实现线性数据结构，也可用于实现非线性数据结构。对于非线性数据结构来说，每个数据元素在逻辑上可能与多个数据元素相邻，而计算机内存的一维地址结构限制了每一个结点只能与前、后各一个结点相邻，结点的相邻反映不出一个元素与多个元素的相邻关系，树、图等就是这种非线性数据结构。在链式存储结构的每个结点中，数据域可分为两类：数据域和指针域。前者用于存放数据元素本身的信息；后者用于存放指针信息。表 1.1 所示的学生成绩表用链式存储结构实现时，如图 1.4 所示。

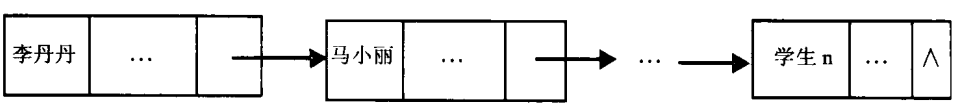


图 1.4　学生成绩表的链式存储结构

图 1.4 中结点包含数据元素域和指针域，指针域的值是所指结点的首地址，通过箭头来表示。如果一个结点的指针域指向另一结点，表示这两个结点的数据元素在逻辑结构中是相邻的。

链式存储结构的特点体现在以下几方面。

（1）结点中除存放数据元素本身的信息外，还存放附加的指针信息。

此为试读，需要完整PDF请访问：www.ertongbook.com