21世纪高职高专规划教材 计算机系列

数据结构实用教程

（C语言版）

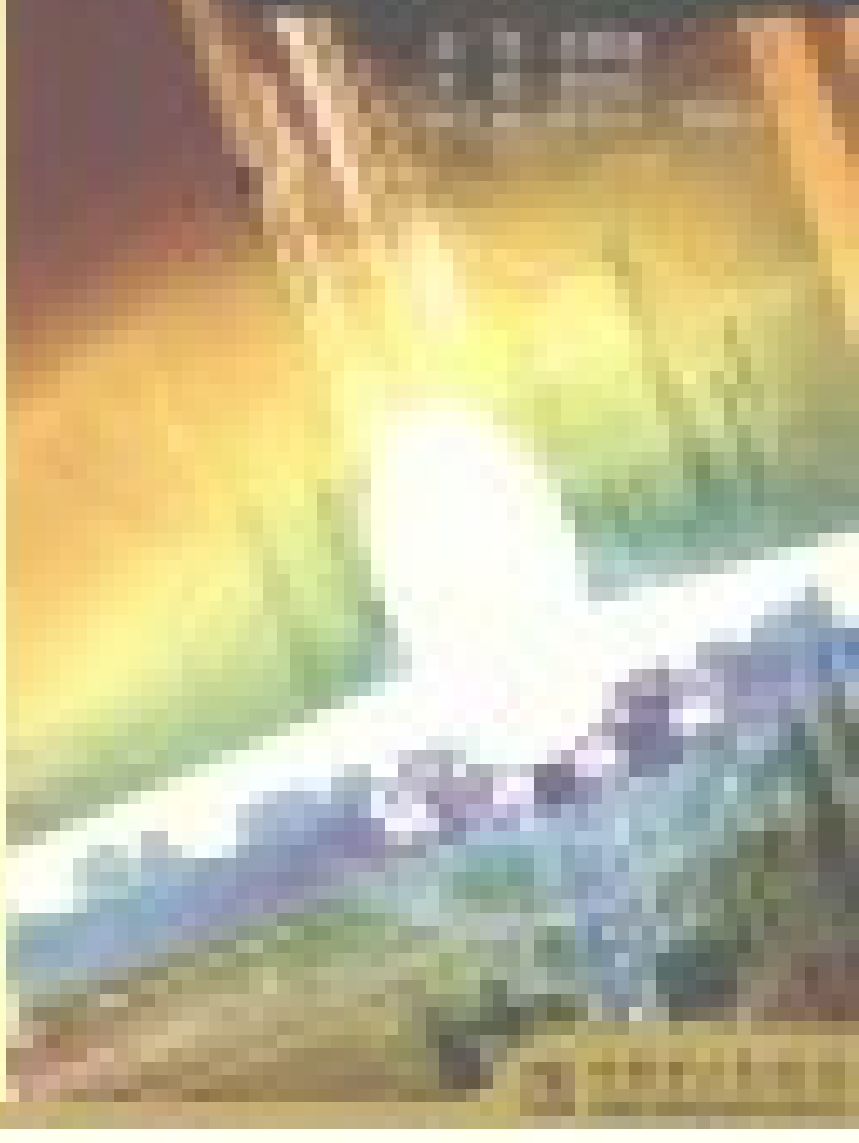
主　审：岳国英
主　编：胡文红
副主编：谭家兴　郑婉华

- 数名一线教师多年教学经验集萃
- 图文并茂、条理清晰、易教易学
- 精心设计上机实验与习题
- 免费提供PPT格式电子教案

中国电力出版社
www.infopower.com.cn

数据结构实用教程

（C语言版）



21世纪高职高专规划教材 计算机系列

数据结构实用教程

（C语言版）

主　审：岳国英
主　编：胡文红
副主编：谭家兴　郑婉华

中国电力出版社
www.infopower.com.cn

内容提要

本书是21世纪高职高专规划教材·计算机系列中的一本。

全书由9个章节和2个附录组成，采用C语言描述，系统地介绍了各种类型的数据结构和查找排序的方法，主要内容包括数据结构概论、线性表、栈和队列、数组与矩阵、树、图、排序、查找以及文件等。每一章都精心设计了习题，并在附录中安排了上机实验和课程设计等内容，做到了学用结合，使读者能够迅速掌握相应知识。为了方便教学，本书配有PPT格式电子教案，免费为任课教师提供。

本书本着理论必需、够用的原则，突出实用性、操作性，加强理论联系实际，语言上通俗易懂，做到了好教易学，以满足目前教学的实际需要。

本书可作为高职高专院校计算机及相关专业学生学习数据结构课程的教材，也可作为计算机科技人员和广大编程爱好者自学的教材或参考书。

图书在版编目（CIP）数据

数据结构实用教程（C语言版）/胡文红等编著. —北京：中国电力出版社，2005

21世纪高职高专规划教材·计算机系列

ISBN 7-5083-2962-7

Ⅰ.数... Ⅱ.胡... Ⅲ.①数据结构－高等学校：技术学校－教材②C语言－程序设计－高等学校：技术学校－教材 Ⅳ.①TP311.12②TP312

中国版本图书馆CIP数据核字（2004）第127867号

丛 书 名：21世纪高职高专规划教材·计算机系列

书　　名：数据结构实用教程（C语言版）

出版发行：中国电力出版社

地　　址：北京市三里河路6号　　邮政编码：100044

电　　话：（010）68358031（总机）　　传　　真：（010）68316497，88518169

本书如有印装质量问题，我社负责退换

服务电话：（010）88515918（总机）　　传　　真：（010）88518169

E-mail：infopower@cepp.com.cn

印　　刷：汇鑫印务有限公司

开本尺寸：185×233　　**印　　张**：11.25　　**字　　数**：260千字

书　　号：ISBN 7-5083-2962-7

版　　次：2005年2月北京第1版

印　　次：2005年2月第1次印刷

印　　数：0001—4000册

定　　价：17.00元

21世纪高职高专规划教材·计算机系列

编 委 会

21 世纪高职高专规划教材参编院校

（排名不分先后）

保定电力职业技术学院
山东电力高等专科学校
黄河水利职业技术学院
湖北水利水电职业技术学院
长江工程职业技术学院
郑州电力高等专科学校
武汉电力职业技术学院
江西电力职业技术学院
浙江水利水电高等专科学校
福建水利电力职业技术学院
广东水利电力职业技术学院
四川水利职业技术学院
兰州电力技术学院
兰州电力学校
南昌水利水电高等专科学校
贵州电力职业技术学院
福建电力职业技术学院
广西电力职业技术学院
内蒙古电力学校
浙江电力职业技术学院
四川电力职业技术学院
石家庄职业技术学院
秦皇岛职业技术学院
唐山工业职业技术学院
唐山科技职业技术学院
天津职业大学
天津大学职教学院
天津理工大学职业技术学院
北京科技大学（管庄校区）
天津渤海职业技术学院
天津轻工职业技术学院
天津中德职业技术学院
天津石油职业技术学院
北京联合大学
太原理工大学
长治职业技术学院
湖南工业职业技术学院
广西工学院职业技术学院
苏州职业大学
南通职业大学
常熟理工学院
常州工学院
徐州工程学院
常州纺织服装职业技术学院
常州轻工职业技术学院
常州信息职业技术学院
连云港职业技术学院
南京工程学院
武汉公交职业技术学院
湖北轻工职业技术学院
武汉职业技术学院
四川工程职业技术学院
四川托普信息技术职业学院
泸州职业技术学院

前　言

“数据结构”是高等院校计算机及相关专业的一门重要的专业基础课。在计算机技术学科的各个领域都要用到各种数据结构。“数据结构”的教学重点是培养学生分析数据和组织数据的能力，并能在实际应用中运用所学知识，编制出正确且有效的程序。

本书共分为9个章节、2个附录。第1章介绍了数据结构和算法的基本概念，并对算法的分析做了说明；第2～6章分别介绍了线性表、栈、队列、数组、树和图等基本数据结构，讨论了各种结构的逻辑特征、存储方式和基本运算，同时也通过一些实例，讲述了这些结构的应用；第7、8章介绍了几种常用的内排序方法和检索方法，给出了相应的算法并对算法做了简单分析。第9章介绍了文件的基本概念，并根据外存上数据的不同组织方式介绍了几种常见的文件结构。附录A提供了上机实验内容；附录B提供了课程设计内容和指导。各章中的算法均用C语言描述，书中所有程序都上机运行通过。

根据高职高专的特点，本书在文字描述上力求通俗易懂；在算法描述上力求结构清晰。为了帮助理解课程内容，在部分章节中还安排了一些实例，编写了相应的算法，可以直接上机运行。在每章的最后都有小结，并安排了适量的习题。

本书为高职高专计算机及相关专业的专业教材，建议讲授课时为60学时，上机实验课时为20学时。各校可根据自己的实际情况增删课时。由于数据结构是一门理论与实践相结合的课程，因此要求学生在完成理论学习的同时，适当地进行上机实验，切实提高用计算机解决实际问题的能力。

本书的大纲在广泛听取教学第一线的教师的要求和意见的基础上，由胡文红执笔。第1～4章由谭家兴执笔，第5章和附录B由胡文红执笔，第6章由王红霞、陆赛群执笔，第7～9章节和附录A由郑婉华执笔。全书由胡文红修改并统稿，岳国英教授担任主审。

由于作者水平有限且时间仓促，本书中一定还存在不少问题，敬请广大读者批评指正。感谢中国电力出版社对本书的出版给予的支持和鼓励，感谢有关高校给予作者的大力支持。

作　者

2004年11月

目　录

第 1 章　数据结构概论

在学习数据结构之前，首先来了解一下什么是数据结构、学习数据结构的目的以及在数据结构中经常出现的名词术语。这对从总体上理解数据结构这门课的基本特点、把握数据结构的基本概念和后续章节内容的学习会起到“纲举目张”的作用。

1.1　数据结构的基本概念

众所周知，1946 年第一台计算机问世的直接原因是解决弹道学的计算问题，当时计算机的应用只局限于科学和工程计算，其处理对象是纯数值性的问题，也就是通常所说的数值计算。

后来随着计算机的发展和应用范围的拓宽，计算机的处理对象从简单的纯数值性的信息发展到非数值性和具有一定结构的信息。按现代计算机科学的观点，把计算机程序处理的一切数值信息、非数值信息，乃至程序本身统称为数据。

由于数据的表示方法和组织形式直接关系到程序对数据的处理效率，而系统程序和许多应用程序的规模庞大，结构复杂，处理对象又多为非数值性数据，因此单凭程序设计人员的经验和技巧难以设计出效率较高的程序。于是就要求人们对计算机加工的对象进行系统的研究。即研究数据的特性以及数据之间存在的关系，以及如何有效地组织、管理与存储数据，从而提高计算机处理数据的效率。数据结构正是为解决这些问题而进行研究的一门课程。

1.1.1　数据结构

1. 数据结构的概念

对于数据结构的概念，首先举例说明，然后再给出明确的含义。

假定有一个学生通讯录，记录了某学校学生的姓名和相应的住址，现在要写一个算法：当给定任何一位学生的姓名时，计算机能够查出该学生的住址，如果没有该学生，就输出“查找失败”。这样一个算法的设计，将完全依赖于通信录的学生姓名及相应住址的组织形式和存储方式。

如果通讯录的学生姓名的组织形式是任意的，其次序没有规律，那么，当给定一个学生姓名时，只能对通讯录从头开始逐个与给定的学生姓名进行比较，按顺序查对。显然这种方法的耗费时间多，处理效率很低。

然而，若对学生通讯录的数据进行适当的组织，按学生所在班级来排列，并且再造一个索引表，用这个表来登记每个班级学生姓名在通讯录中起始处的位置。这样一来，查找算法将大为改善。这时，当要查找某一个学生的住址时，可先从索引表中查到该学生所在班级的学生姓名在通讯录中的起始处，而后，就从此起始处开始查找，不必再去查找其他班级学生的姓名了。

显然，由于采用了新的结构思想（即索引结构），使计算机算法得到优化，查找效率得到明显的提高。

如何组织上述学生通讯录中的数据就是一个数据结构问题。不难看出，对于不同的组织形式，需要构造不同的算法，而不同的算法其查找效率又明显不同。因此，计算机算法与数据的结构有着密切关系，算法无不依附于具体的数据结构，而数据结构又直接关系到算法的选择和效率。

此外，当有新学生进校时，通讯录需要添加新学生的姓名和相应的住址；在有学生毕业或转学时，应从通讯录中删除毕业或转学的学生姓名和住址。这就要求在已形成的结构上进行插入（Insert）和删除（Delete）。对于一种具体的结构，如何实现插入和删除？是要把添加的学生姓名和住址插入到学生通讯录的前头，还是插入到末尾，或是中间某个合适的位置上？插入后，对原有的数据是否有影响？有什么样的影响？删除某学生的姓名和住址后，其他的数据（学生的姓名和住址）是否要移动？若需要移动，则应如何移动？这一系列的问题说明，为适应数据的增加和减少的需要，还必须对数据结构给出各种运算和算法。

由此可见，数据结构一般包括以下三方面内容。

（1）逻辑结构（Logical Structure）。数据的逻辑结构是从逻辑关系上描述数据，与数据的存储无关，是独立于计算机的。数据的逻辑结构可以看作是从具体问题抽象出来的数学模型。通俗地讲，数据的逻辑结构就是数据及数据之间的组织形式（即数据之间的逻辑关系）。数据的逻辑结构有两个要素：一是数据元素的集合，通常记为 D；二是 D 上的关系，它反映了 D 中各个数据元素之间的逻辑关系，通常记为 R，即一个数据结构 B 可以表示成

$$B=(D, R)$$

【例 1-1】一年四季按时序组织的数据结构可以表示为

$$B=(D, R)$$
$$D=\{春, 夏, 秋, 冬\}$$
$$R=\{(春, 夏), (夏, 秋), (秋, 冬)\}$$

（2）存储结构（Storage Structure）。数据的存储结构是指数据的逻辑结构在计算机存储空间中的存放形式，它只有通过计算机语言才能实现（亦称为映像）。对机器语言而言，存储结构是具体的。一般只在高级语言的层次上讨论存储结构。存储结构不仅要考虑数据的存储，还要考虑数据之间逻辑关系的体现。

（3）数据的运算。数据的运算是指对数据施加的操作。数据的运算定义在数据的逻辑结构上，每种逻辑结构都有一个运算的集合。最常用的有检索、插入、删除、更新、排序等运算。

2. 数据的逻辑结构分类

在不产生混淆的前提下，通常将数据的逻辑结构简称为数据结构。数据的逻辑结构有两大类：

（1）线性结构。线性结构的逻辑特征是：若结构是非空集，则有且仅有一个开始结点和一个终端结点，并且除开始结点和终端结点外，其余所有结点只有一个直接前趋和一个直接后继。

线性表是一个典型的线性结构。

（2）非线性结构。非线性结构的逻辑特征是：一个结点可能有多个直接前趋和直接后继。

树和图都是非线性结构。

3. 数据的四种基本存储方法

数据的存储结构可用以下四种基本存储方法得到：

（1）顺序存储方法。该方法把逻辑上相邻的结点存储在物理位置上相邻的存储单元里，结点间的逻辑关系由存储单元的邻接关系来体现。由此得到的存储表示称为顺序存储结构（Sequential Storage Structure），通常借助程序语言的数组描述。

（2）链接存储方法。该方法不要求逻辑上相邻的结点在物理位置上也相邻，结点间的逻辑关系由附加的指针字段表示。由此得到的存储表示称为链式存储结构（Linked Storage Structure），通常借助于程序语言的指针类型描述。

（3）索引存储方法。该方法通常在存储结点信息的同时，还建立附加的索引表。索引表由若干索引项组成。若每个结点在索引表中都有一个索引项，则索引项的一般形式是：（关键字、地址）。通过关键字可以查到结点所在的存储位置。

（4）散列存储方法。该方法的基本思想是：根据结点的关键字直接计算出该结点的存储地址。

四种基本存储方法，既可单独使用，也可组合起来对逻辑结构进行存储映像。同一逻辑结构采用不同的存储方法，可以得到不同的存储结构。选择何种存储结构来表示相应的逻辑结构，视具体要求而定，主要考虑运算方便及算法的时间和空间要求。

4. 数据结构的用途

总体来说，“数据结构”在计算机科学中是一门综合性的专业基础课。“数据结构”的研究和应用不仅涉及到计算机硬件（特别是编码理论、存储装置和存取方法等），而且和计算机软件的研究及应用也有着密切的关系，无论是在最简单的计算机操作、应用程序的编写，还是编译程序、操作系统开发及研究，都涉及到数据元素在存储器中的分配问题。在计算机信息检索和程序设计时也必须考虑如何有效地组织数据结构和选择算法，以便使查找和存取数据元素更为方便、快捷。因此可以认为，“数据结构”是介于数学、计算机硬件和计算机软件三者之间的一门核心课程。

1.1.2 常用术语

本节为常用的概念和术语赋予确定的含义，以便与读者取得一致的语言，这些常用的概念和术语将在后续章节中多次出现。

数据（Data）是信息的载体。它能够被计算机识别、存储和加工处理，是计算机程序加工的“原料”。随着计算机应用领域的扩大，数据的范畴包括：整数、实数、字符串、图像和声音等。

数据元素（Data Element）是数据的基本单位，在计算机信息处理中通常作为一个整体来考虑。数据元素也称元素、结点、顶点、记录。一个数据元素可以由若干个**数据项**（也可称为字段、域、属性）组成。数据项是具有独立含义的不可分割的最小标识单位。

数据类型（Data Type）是和数据结构密切相关的一个概念，它最早出现在高级程序语言中，用以刻画操作对象的特性。在用高级程序语言编写的程序中，每个变量、常量或表达式都

有一个它所属的确定的数据类型。类型明确或隐含地规定了程序执行期间变量或表达式所有可能取值的范围，以及在这些值上允许进行的操作。因此数据类型是一个值的集合和定义在这个值集上的一组操作的总称。例如，C 语言中的整型变量，其值集为某个区间上的整数（区间大小依赖于不同的机器），定义在其上的操作有加、减、乘、除和取模等算术运算。

从数据结构三方面的关系来看：数据的逻辑结构、数据的存储结构及数据的运算这三方面是一个整体。孤立地理解一个方面，而不注意它们之间的联系是不可取的。

存储结构是数据结构不可缺少的一个方面，同一逻辑结构的不同存储结构可冠以不同的数据结构名称来标识。例如，线性表是一种逻辑结构：若采用顺序存储方法，可称其为**顺序表**；若采用链式存储方法，则可称其为**链表**；若采用散列存储方法，则可称为**散列表**；若采用索引存储方法，则可称为**索引表**。

而数据的运算也是数据结构不可分割的一个方面。在给定了数据的逻辑结构和存储结构之后，按定义的运算集合及其运算的性质不同，也可得到完全不同的数据结构：若对线性表上的插入、删除运算限制在表的一端进行，则该线性表称之为**栈**；若对插入限制在表的一端、而删除限制在表的另一端进行，则该线性表称之为**队列**。更进一步，若线性表采用顺序表或链表作为存储结构，则对插入和删除运算做了上述限制之后，可分别得到**顺序栈或链栈，顺序队列或链队列**。

1.2 算法及算法评价

1.2.1 算法

由于数据的运算是通过算法描述的，所以对算法的讨论是学习数据结构的重要内容。

一般地说，算法可以被认为是用来解决一个计算问题的方法或者是一系列将输入转换为输出的计算步骤。更严格的定义，算法是由若干条指令组成的有穷序列，它具有下列几个重要特性：

（1）输入：具有 0 个或多个输入的外界量，它们是算法开始前对算法给予的最初量。

（2）输出：至少产生一个输出，它们是同输入有某种关系的结果量。

（3）有穷性：一个算法必须在执行有穷步之后结束，且每一步的执行次数必须是有限的。

（4）确定性：每条指令的含义都必须明确，无二义性。并且，在任何条件下，算法只有惟一的一条执行路径，即对于相同的输入只能得到相同的输出。

（5）可行性：算法中描述的操作都是可以通过已经实现的基本运算执行有限次来实现的。

在一个算法中，有些指令可能是重复执行的，因而指令的执行次数可能远远大于算法中的指令条数。由有穷性可知，对于任何输入，一个算法在执行了有限条指令后一定要终止；又由可行性知道，一个算法描述的操作必须在有限时间内实现。因此，一个程序如果对任何输入都不会陷入无限，并能产生同输入有某种关系的量，则它就是一个算法。

另外，算法与程序有相同的地方，但也有区别，见表 1-1。

表 1-1　算法与程序的比较

	算　法	程　序
相同点	（1）输入　（2）输出　（4）确定性　（5）可行性	
不同点	有穷性	不一定满足有穷性，如操作系统
	不一定可执行	指令（或语句）必须可执行

1.2.2　算法的描述

一个算法可以用自然语言、计算机程序语言或其他语言来描述，惟一的要求是该描述必须精确地说明计算过程。

一般而言，描述算法最合适的语言是介于自然语言和程序语言之间的伪语言。它的控制结构往往类似于 Pascal、C 等程序语言，但其中可使用任何表达能力强的方法使算法表达更加清晰和简洁，而不至于陷入具体的程序语言的某些细节。

从易于上机验证算法和提高实际程序设计能力考虑，本书采用 C 语言描述算法。

1.2.3　算法的评价

1. 评价算法好坏的标准

求解同一计算问题可能有许多不同的算法，究竟如何来评价这些算法的好坏以便从中选出较好的算法呢？

选用的算法首先应该是“正确”的。此外，主要考虑如下三点：

（1）执行算法所耗费的时间。

（2）执行算法所耗费的存储空间，其中主要考虑辅助存储空间。

（3）算法应易于理解、易于编码、易于调试等等。

2. 算法性能选择

一个占存储空间小、运行时间短、其他性能也好的算法是很难做到的。原因是上述要求有时相互抵触：要节约算法的执行时间往往要以牺牲更多的空间为代价；而为了节省空间可能要耗费更多的计算时间。因此只能根据具体情况有所侧重：若该程序使用次数较少，则力求算法简明易懂；若对于反复多次使用的程序，应尽可能选用快速的算法；若待解决的问题数据量极大，机器的存储空间较小，则相应算法主要考虑如何节省空间。

3. 算法的时间性能分析

算法执行的时间需要通过该算法编制的程序在计算机上运行所耗费的时间来度量。而度量一个程序的执行时间通常有两种方法。

（1）事后统计的方法。因为计算机内部都有计时器，所以不同算法的程序可以通过一组或若干组相同的统计数据来区分优劣，但该方法有两个缺陷：一是必须先运行依据算法编制的程序；二是所得时间的统计量依赖于计算机的硬件、软件等环境因素，容易掩盖算法本身的优劣，因而人们通常使用另外一种称为事前分析估计的方法。

（2）事前分析估计的方法。该方法撇开计算机硬件和软件的有关因素，只从算法本身的规模来分析评估。一个算法所耗费的时间等于算法中每条语句的执行时间之和，而每条语句的执行时间等于语句的重复执行次数（即频度）与该语句执行一次所需时间的乘积。

【例 1-2】两个数相乘的算法。

程序	频度
`for(i=0;i<n;i++)`	`n+1`
`for(j=0;j<n;j++)`	`n(n+1)`
`  t=i*j;`	n^2

第一个语句 for 的循环控制变量 i 由 0 增加到 $n-1$ 需执行 n 次，加上测试 $i \geqslant n$ 成立后才会终止，又需执行 1 次，故该语句共执行 $n+1$ 次，即第一个语句的频度为 $n+1$。

第二个语句 for 的循环控制变量 j 由 0 增加到 $n-1$，同样测试 $j \geqslant n$ 成立后才会终止，该语句执行 $n+1$ 次，加之它是第一个语句 for 的循环体内的语句，要循环 n 次，故第二个语句的频度为 $n(n+1)$。

第三个语句 $t=i*j$ 的频度为 n^2。

由此可得出两个数相乘算法所耗费的时间（所有语句的频度之和）为

$$T(n)=2n^2+2n+1$$

或者说算法的耗费时间 $T(n)$是 n 的函数。显然，整个算法的执行时间与该基本操作（乘法）重复执行次数 n^2 成正比，记作

$$T(n)=O(n^2)$$

一般情况下，算法中基本操作重复执行次数是问题规模 n 的某个函数 $f(n)$，算法的时间度量称为算法的时间复杂度，记作

$$T(n)=O(f(n))$$

它表示随问题规模 n 的增大，算法执行时间的增长率和 $f(n)$的增长率相同，即 $T(n)$与 $f(n)$是 n 的同阶量。

【例 1-3】交换 i 和 j 的内容。

```
temp=i;
i=j;
j=temp;
```

以上三条单个语句的频度均为 1，该程序段的执行时间是一个与问题规模 n 无关的常数。算法的时间复杂度为常数阶，记作 $T(n)=O(1)$。如果算法的执行时间不随着问题规模 n 的增加而增长，即使算法中有上千条语句，其执行时间也不会是一个较大的常数。此类算法的时间复杂度都是 $O(1)$。

【例 1-4】变量计数。

① `x=1;`

```
② for(i=1;i<=n;i++)
③   for(j=1;j<=i;j++)
④     x++;
```

该程序段中频度最大的语句是④，内循环语句③的执行次数虽然与问题规模 n 没有直接关系，但是却与外循环语句②的变量 i 取值有关，而外循环变量 i 的执行次数又直接与 n 有关，因此可以分析语句④的执行次数为

$$1+2+3+\cdots+n=\frac{n(n+1)}{2}=\frac{n^2}{2}+\frac{n}{2}$$

则该程序段的时间复杂度为 $T(n)=O(n^2)$。

小　结

本章简要地介绍了数据、数据结构等基本概念；阐述了数据结构所包含的三个方面的内容，即数据的逻辑结构、数据的存储结构和数据的运算；讨论了线性结构和非线性结构的逻辑特征，以及数据存储的四种基本方法。希望读者学习这些内容和例子后，能对数据结构的基本要领具有初步的认识和理解。

算法和数据结构密切相关，不可分割，本章引进了算法、算法的时间复杂度等概念，以及分析时间复杂度的简易方法。

习　题

1.1　简述下列概念：数据、数据元素、数据类型、数据结构、逻辑结构、存储结构、线性结构、非线性结构。

1.2　什么是算法？评价一个算法好坏的标准是什么？

1.3　什么是算法的时间复杂度？

1.4　设 n 为大于 1 的正整数，将下列程序段带有@符号的语句的执行时间表示为 n 的函数，并利用大“O”表示法得出时间复杂度。

```
(1) i=1; k=0;
    while(i<n)
    {@   k=k+10*i;
         i++;
    }

(2) x=0;
    for(i=1;i<=n;i++)
    for(j=1;j<=n;j++)
```

```
        @   x=x+1;
(3)  i=0; k=0;
     do
     {@   k=k+10*i;
          i++;
     }while(i<n);

(4)  x=0;
     for(i=1;i<=n;i++)
     for(j=1;j<=i;j++)
     for(k=1;k<=j;k++)
          @   x=x+1;
```