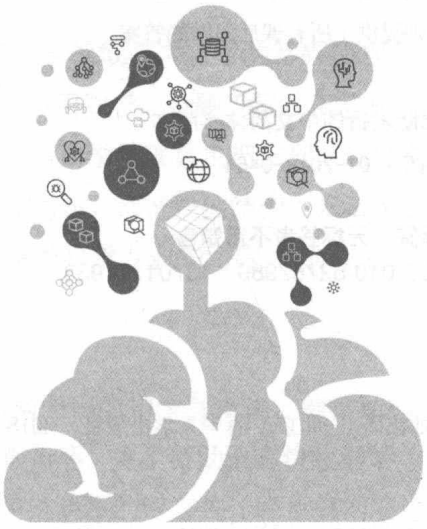
丰富的图文详解
Python范例诠释

图解数据结构
使用Python

- 用丰富的图例来阐述数据结构的复杂理论和算法，有效提高可读性
- 用Python语言实现数据结构中的重要理论，以范例程序说明数据结构的内涵

吴灿铭 著

清華大學出版社



图解数据结构
使用Python

吴灿铭 著

清华大学出版社
北京

内 容 简 介

本书采用丰富的图例来阐述基本概念，并以简洁清晰的语言来诠释重要的理论和算法，同时配合完整的范例程序代码，使读者可以通过“实例+实践”来熟悉数据结构。

本书内容共9章，先从基本的数据结构概念开始介绍，再以Python语言来实现数组、堆栈、链表、队列、树、图、排序、查找等重要的数据结构。在附录A提供了Python语言的快速入门，附录B是使用Python语言实现数据结构程序时调试经验的分享，附录C则提供了所有课后习题的答案。

图书在版编目（CIP）数据

图解数据结构：使用Python/吴灿铭著. —北京：清华大学出版社，2018
ISBN 978-7-302-49532-1

Ⅰ. ①图… Ⅱ. ①吴… Ⅲ. ①数据结构－图解②软件工具－程序设计 Ⅳ. ①TP311.12-64②TP311.561

中国版本图书馆CIP数据核字（2018）第029409号

责任编辑：夏毓彦
封面设计：王　翔
责任校对：闫秀华
责任印制：刘海龙

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦A座　　**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：三河市金元印装有限公司
经　　销：全国新华书店
开　　本：190mm×260mm　　**印　　张：**26.5　　**字　　数：**678千字
版　　次：2018年4月第1版　　**印　　次：**2018年4月第1次印刷
印　　数：1~3000
定　　价：79.00元

产品编号：077878-01

前　言

数据结构一直是计算机科学领域非常重要的基础课程，它是各大专院校的计算机科学、信息科学、信息工程、应用数学、金融工程等信息相关系的必修课程，近年来包括电子工程、通信工程以及一些商学管理系也把它列入选修课程。同时，一些信息相关科系的转系考试和研究生升学考试，都把数据结构列入必考的专业课。由此可知，无论是从考试的角度还是研究信息类科学专业的角度，数据结构都是被高度重视的一门基础+核心课程。

对于第一次接触数据结构课程的初学者来说，数据结构中大量的理论及算法不易理解，常会造成学习障碍与挫折感。为了帮助读者快速理解数据结构，本书采用丰富的图例来阐述基本概念，并以简洁清晰的语言来诠释重要的理论和算法，同时配合完整的范例程序代码，期望通过“实例+实践”来熟悉数据结构。因此，本书是兼具内容和专业的数据结构教学用书。

市面上以 Python 程序设计语言来实践数据结构理论的书比较少，本书则是针对这种情况而编写的。本书提供了完整的程序代码，让学习变得更加轻松。本书先从最基本的数据结构概念开始，再以 Python 语言来实现数组、堆栈、链表、队列、树、图、排序、查找等重要的数据结构。在附录 A 提供了 Python 语言的快速入门，附录 B 则是使用 Python 语言实现数据结构的程序时调试经验的分享。

笔者长期从事信息教育及写作工作，在语句的表达上尽量简洁有力，为了检验大家在各章的学习成果，还特别搜集了大量的习题。

一本好的理论书籍除了内容完备和专业外，更需要有清楚易懂的结构安排和表达方式。在仔细阅读本书之后，相信读者会体会笔者的用心，也希望读者能对计算机专业这门基础+核心的学科有更深、更完整的认识。

改编说明

现在无人不谈“大数据技术”和“人工智能技术”，而商业智能和机器学习等应用的具体开发中又大量使用 Python 这门排名已经上升到第 5 位的程序设计语言。另外，已经有越来越多的大专院校采用 Python 语言来教授计算机程序设计课程，因而用 Python 语言来描述算法和讲述数据结构就成为顺其自然的事情了。

“数据结构”毫无疑问是计算机科学既经典又核心的课程之一，只要从事计算机相关的开发工作，系统地学习数据结构是进入这个行业的“开山斧”。数据结构不仅讲授数据的结构以及在计算机内存储和组织数据的方式，它背后真正蕴含的是与之息息相关的算法，精心选择的数据结构配合恰如其分的算法就意味着数据或者信息在计算机内被高效率地存储和处理。算法其实就是数据结构的灵魂，它既神秘又神奇“好玩”，可以说是“聪明人在计算机上的游戏”。

本书是一本综合且全面讲述数据结构及其算法分析的教科书，为了便于高校的教学或者读者自学，作者在描述数据结构原理和算法时文字清晰而严谨，为每个算法及其数据结构提供了演算的详细图解。另外，为了适合在教学中让学生上机实践或者自学者上机“操练”，本书为每个经典的算法都提供了 Python 语言编写的完整范例程序（包含完整的源代码），每个范例程序都经过了测试和调试，可以直接在标准的 Python 解释器中运行，目的就是让本书的学习者以这些范例程序作为参照，迅速掌握数据结构和算法的要点。本书所有范例程序的下载网址如下：

https://pan.baidu.com/s/1jJWK4Lo（注意区分数字和英文字母的大小写）

如果下载有问题，请电子邮件联系 booksaga@126.com，邮件主题为“图解数据结构——使用 Python 源代码”。

学习本书需要有面向对象程序设计语言的基础，如果读者没有学习过任何面向对象的程序设计语言，那么建议读者先学习一下 Python 语言再来学习本书。如果读者已经掌握了 Java、C++、C#等任何一种面向对象的程序设计语言，而没有学习过 Python 语言，只需快速浏览一下附录 A“Python 语言快速入门”，即可开始本书的学习。关

于 Python 运行环境的简单说明也在附录 A 中讲解，当然读者也可以从 https://www.python.org/网站中下载适合自己的计算机及其操作系统的 Python 版本，再安装和设置好 Python 运行环境。

为了方便教学和读者自学，本书每章的最后都提供了丰富的课后习题，同时在整本书的附录 C 也提供了所有课后习题的详细解答，供读者参考对照。

资深架构师 赵军

2018 年 1 月

目　录

第1章 数据结构导论

对于一个有志于从事信息技术（IT）领域的人员来说，数据结构（Data Structure）是一门和计算机硬件与软件都密切相关的学科，它的研究重点是在计算机的程序设计领域中探讨如何在计算机中组织和存储数据并进行高效率的运用，涉及的内容包含算法（Algorithm）、数据存储结构、排序、查找、程序设计概念与哈希函数。

1.1 数据结构的定义

我们可以将数据结构看成是在数据处理过程中的一种分析、存储、组织数据的方法与逻辑，它考虑到了数据之间的特性与相互关系。简单来说，数据结构的定义就是一种程序设计优化的方法论，它不仅讨论到存储的数据，同时也考虑到彼此之间的关系与运算，目的是加快程序的执行速度、减少内存占用的空间。

计算机与数据是息息相关的，计算机具有处理速度快与存储容量大两大特点（见图 1-1），因而在数据处理方面更为举足轻重。数据结构和相关的算法就是数据进入计算机进行处理的一套完整逻辑。在进行程序设计时，对于要存储和处理的一类数据，程序员必须选择一种数据结构来进行这类数据的添加、修改、删除、存储等操作，如果在选择数据结构时做了错误的决定，那么程序执行起来将可能变得非常低效，如果选错了数据类型，那么后果将更加不堪设想。

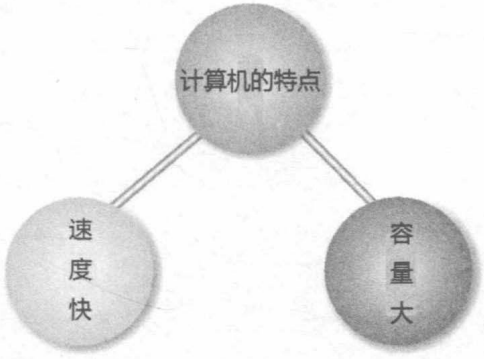


图 1-1　计算机的两大特点

因此，当我们要求计算机解决问题时，必须以计算机所能接受的模式来确认问题，并且要选用适当的算法来处理数据，这就是数据结构讨论的重点。简单地说，数据结构就是对数据与算法的研究。

1.1.1 数据与信息

谈到数据结构，首先就必须了解数据（Data）与信息（Information）。从字义上来看，所谓数据（Data），指的是一种未经处理的原始文字（Word）、数字（Number）、符号（Symbol）或图形（Graph）等，它所表达出来的只是一种没有评估价值的基本元素或表目。例如，姓名或我们常看到的课程表、通讯簿等都可泛称为一种“数据”（Data）。

当数据经过处理（例如以特定的方式系统地整理、归纳甚至进行分析）后，就成为“信息”（Information）了。这样处理的过程称为“数据处理”（Data Processing），如图 1-2 所示。

从严谨的角度来形容“数据处理”，就是用人力或机器设备对数据进行系统的整理，如记录、排序、合并、计算、统计等，以使原始的数据符合需求，成为有用的信息。

大家可能会有疑问：“数据和信息的角色是否绝对一成不变呢？”这倒也不一定，同一份文件可能在某种情况下为数据，而在另一种情况下则为信息。例如，战争中某场战役的死伤人数报告对百姓而言可能只是一份不痛不痒的“数据”，不过对于指挥官而言，这份报告则是非常重要的“信息”。

图 1-2　数据处理过程的示意图

1.1.2　数据的特性

通常按照计算机中所存储和使用的对象，我们可将数据分为两大类：一类为数值数据（Numeric Data），例如由 0, 1, 2, 3, …, 9 所组成，可用运算符（Operator）来进行运算的数据；另一类为字符数据（Alphanumeric Data），例如+、*以及 A、B、C 等非数值数据（Non-Numeric Data）。不过，若按照数据在计算机程序设计语言中的存在层次来分，则可以分为以下 3 种类型：

- **基本数据类型（Primitive Data Type）**

不能以其他类型来定义的基本数据类型，或称为标量数据类型（Scalar Data Type），几乎所有的程序设计语言都会为标量数据类型提供一组基本数据类型，例如 Python 语言中的基本数据类型就包括整型、浮点型、布尔（bool）类型和字符类型。

- **结构数据类型（Structured Data Type）**

结构数据类型也称为虚拟数据类型（Virtual Data Type），是一种比基本数据类型更高一级的数据类型，例如字符串（string）、数组（array）、指针（pointer）、列表（list）、文件（file）等。

- **抽象数据类型（Abstract Data Type，ADT）**

我们可以将一种数据类型看成是一种值的集合，以及在这些值上所进行的运算及其所代表的属性组成的集合。“抽象数据类型”比结构数据类型更高级，是指一个数学模型以及定义在此数学模型上的一组数学运算或操作。也就是说，ADT 在计算机中用于表示一种“信息隐藏”（Information Hiding）的程序设计思想以及信息之间的某一种特定的关系模式。例如，堆栈（Stack）就是一种典型的数据抽象类型，它具有后进先出（Last In，First Out）的数据操作方式。

1.1.3　数据结构的应用

计算机的主要工作就是把数据（Data）经过某种运算处理转换为实用的信息（Information）。例如一个学生的语文成绩是 90 分，我们可以说这是一项成绩的数据，不过无法判断它具备什么意义。

如果经过某些处理（如排序）可以知道这个学生语文成绩在班上的名次，也就清楚了这个班中学生语文成绩大致如何，因为有了具体的含义，所以成为一种信息，而排序是数据结构的一种应用。下面我们将介绍一些数据结构的常见应用。

树形结构

树形结构是一种相当重要的非线性数据结构，广泛运用在人类社会的族谱、机关的组织结构、计算机的操作系统结构、平面绘图应用、游戏设计等方面。例如，在年轻人喜爱的大型网络游戏中需要获取某些物体所在的地形信息，如果程序依次从构成地形的模型三角面寻找，往往会耗费许多运行时间，非常低效。因此，程序员一般会使用树形结构中的二叉空间分割树（BSP Tree）、四叉树（Quadtree）、八叉树（Octree）等来代表分割场景的数据，如图 1-3 和图 1-4 所示。

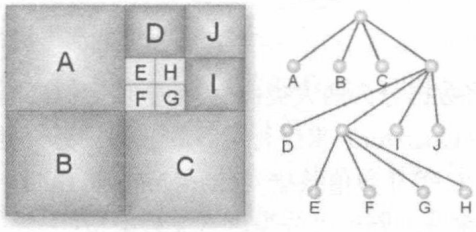


图 1-3　四叉树示意图

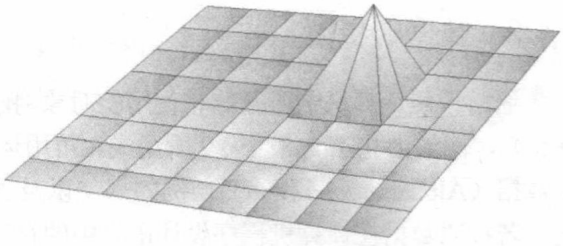


图 1-4　地形与四叉树的对应关系

最短路径

最短路径是指在众多不同的路径中距离最短或者所花费成本最少的路径。寻找最短路径最常见的应用是公共交通运输系统的规划或网络的架设，如都市公交系统、铁路运输系统、通信网络系统等，如图 1-5 所示。

图 1-5　许多公共运输系统的规划都会用到最短路径

例如，全球定位系统（Global Positioning System，GPS）就是通过卫星与地面接收器实现传递地理位置信息、计算路程、语音导航与电子地图等功能。目前有许多汽车与手机都安装了 GPS 用于定位与路况查询。其中路程的计算就是以最短路径的理论作为程序设计的依据，为旅行者提供不同的路径选择方案，增加驾驶者选择行车路线的弹性。

查找理论

所谓“搜索引擎”（Searching Engine），是一种自动从因特网的众多网站中查找信息，再经过一定的整理后提供给用户进行查询的系统，例如百度、谷歌（Google）、搜狗等。

搜索引擎的信息来源主要有两种，一种是用户或网站管理员主动登录，另一种是编写程序主动搜索网络上的信息，例如，百度的 Web Spider 和谷歌的 Spider 程序会主动通过网站上的超链接爬行到另一个网站，并收集该网站上的信息，然后收录到数据库中。

用户在进行查找时，内部的程序设计就必须依赖不同的查找理论来进行，信息会由上而下列出，如果数据笔数过多，就分数页摆放，列出的方式则依照搜索引擎自行判断用户查找时最有可能得到的结果来摆放。

用户使用搜索引擎的搜索功能时，搜索程序就是利用不同的搜索算法进行信息的查找，查到的结果信息会由上而下列出，如果结果信息过多，就会分数页列出，是依照搜索引擎自行判断用户搜索时最有可能得到的结果来按序列出的。

注意，Search 这个英文单词在翻译到不同的中文语境时，因为历史习惯的问题，在网络应用中习惯翻译成“搜索”，如搜索引擎；而在数据结构的算法描述中，一般习惯翻译成“查找”，例如查找理论、查找算法；在计算机科学中，“搜索”和“查找”大部分情况下可以互换使用，只是有时不按照习惯翻译比较别扭而已。

1.2 算法

数据结构与算法是程序设计实践中最基本的内涵。程序能否快速而高效地完成预定的任务，取决于是否选对了数据结构，而程序是否能清楚而正确地把问题解决，则取决于算法。所以大家可以认为“数据结构加上算法等于可执行程序”，如图 1-6 所示。

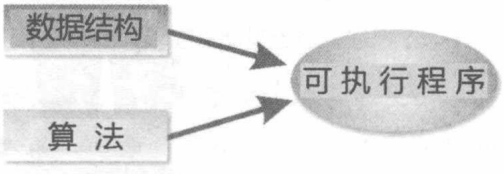


图 1-6　数据结构加上算法等于可执行程序

在韦氏辞典中算法定义为：“在有限步骤内解决数学问题的程序”。如果运用在计算机领域中，我们也可以把算法定义成：“为了解决某项工作或某个问题，所需要的有限数量的机械性或重复性指令与计算步骤”。其实日常生活中有许多工作都可以用算法来描述，例如员工的工作报告、宠物的饲养过程、学生的课程表等。

算法的条件

认识了算法的定义之后，我们再来说明一下算法所必须符合的 5 个条件，如图 1-7 和表 1-1 所示。

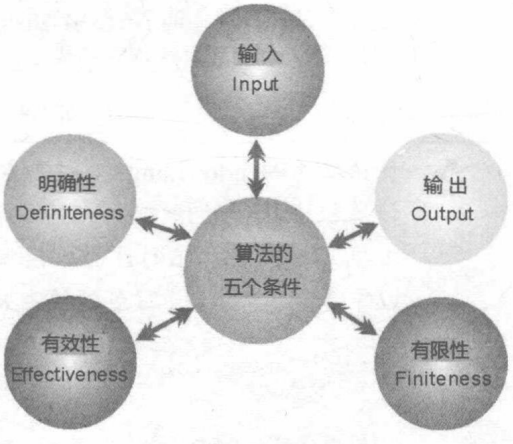


图 1-7　算法必须符合的 5 个条件

表 1-1　算法必须符合的 5 个条件

算法的特性	内容与说明
输入（Input）	0 个或多个输入数据，这些输入必须有清楚的描述或定义
输出（Output）	至少会有一个输出结果，不可以没有输出结果
明确性（Definiteness）	每一个指令或步骤必须是简洁明确的
有限性（Finiteness）	在有限步骤后一定会结束，不会产生无限循环
有效性（Effectiveness）	步骤清楚且可行，能让用户用纸笔计算而求出答案