PEARSON Addison Wesley

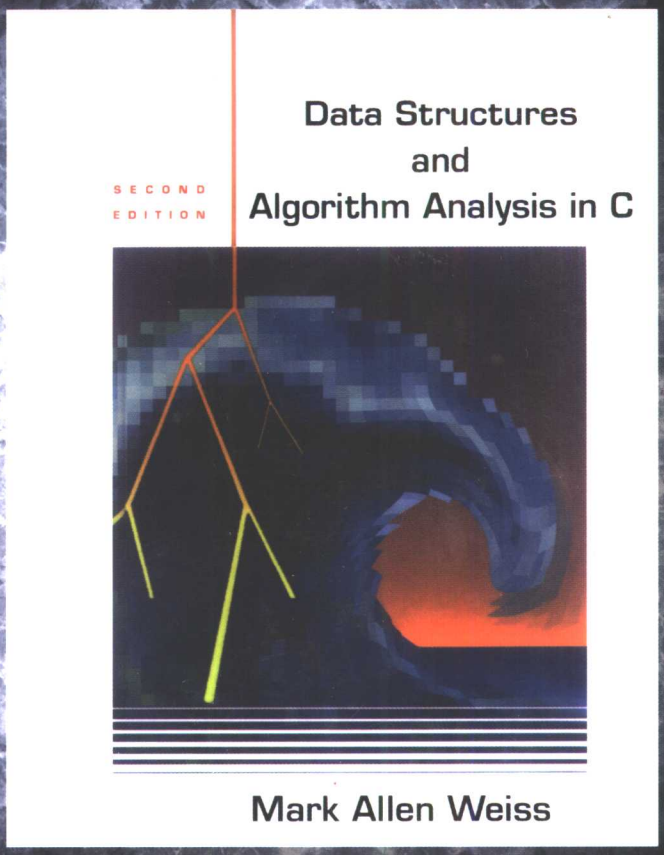
计 算 机 科 学 丛 书

原书第2版

数据结构与算法分析

——C语言描述

（美） Mark Allen Weiss 著　冯舜玺 译



Data Structures and Algorithm Analysis in C

Second Edition

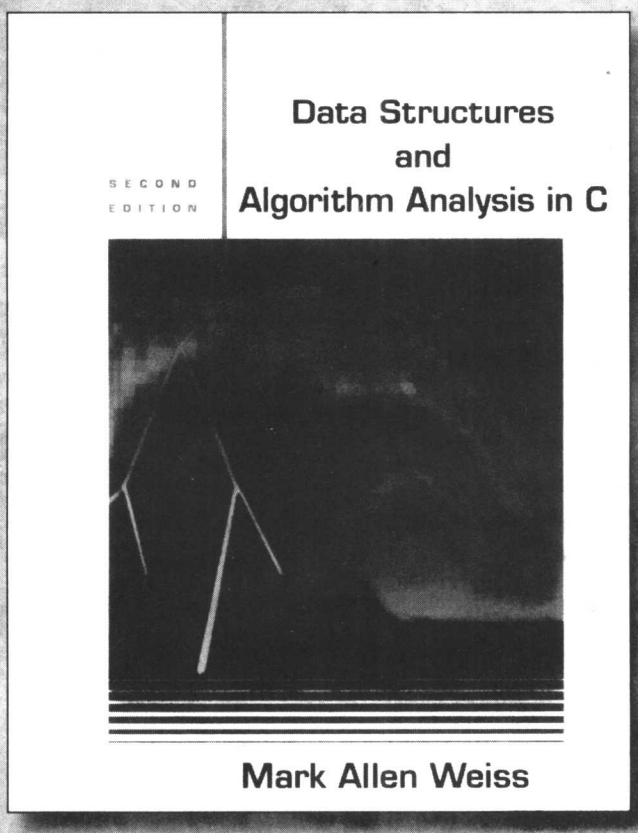
机械工业出版社
China Machine Press

计 算 机 科 学 丛 书

原书第2版

数据结构与算法分析
——C语言描述

（美） Mark Allen Weiss 著　冯舜玺 译



Data Structures and Algorithm Analysis in C
Second Edition

机械工业出版社
China Machine Press

本书是国外数据结构与算法分析方面的标准教材，介绍了数据结构（大量数据的组织方法）以及算法分析（算法运行时间的估算）。本书的编写目标是同时讲授好的程序设计和算法分析技巧，使读者可以开发出具有最高效率的程序。

本书可作为高级数据结构课程或研究生一年级算法分析课程的教材，使用本书需具有一些中级程序设计知识，还需要离散数学的一些背景知识。

本书版权登记号：图字：01-2001-2200

图书在版编目（CIP）数据

数据结构与算法分析：C 语言描述（原书第 2 版）/（美）维斯著；冯舜玺译. －北京：机械工业出版社，2004.1

（计算机科学丛书）

书名原文：Data Structures and Algorithm Analysis in C: Second Edition

ISBN 7-111-12748-X

Ⅰ. 数… Ⅱ. ①维… ②冯… Ⅲ. ①数据结构 ②算法分析 ③C 语言－程序设计 Ⅳ. TP311.12

中国版本图书馆 CIP 数据核字（2003）第 068447 号

机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码 100037）

责任编辑：蒋 祎

北京牛山世兴印刷厂印刷 · 新华书店北京发行所发行

2004 年 1 月第 1 版第 1 次印刷

787mm×1092mm 1/16 · 25.5 印张

印数：0 001-5 000 册

定价：35.00 元

出版者的话

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭橥了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短、从业人员较少的现状下，美国等发达国家在其计算机科学发展的几十年间积淀的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章图文信息有限公司较早意识到“出版要为教育服务”。自1998年开始，华章公司就将工作重点放在了遴选、移译国外优秀教材上。经过几年的不懈努力，我们与Prentice Hall，Addison-Wesley，McGraw-Hill，Morgan Kaufmann等世界著名出版公司建立了良好的合作关系，从它们现有的数百种教材中甄选出Tanenbaum，Stroustrup，Kernighan，Jim Gray等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及庋藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专诚为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍，为进一步推广与发展打下了坚实的基础。

随着学科建设的初步完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都步入一个新的阶段。为此，华章公司将加大引进教材的力度，在“华章教育”的总规划之下出版三个系列的计算机教材：除“计算机科学丛书”之外，对影印版的教材，则单独开辟出“经典原版书库”；同时，引进全美通行的教学辅导书“Schaum's Outlines”系列组成“全美经典学习指导系列”。为了保证这三套丛书的权威性，同时也为了更好地为学校和老师们服务，华章公司聘请了中国科学院、北京大学、清华大学、国防科技大学、复旦大学、上海交通大学、南京大学、浙江大学、中国科技大学、哈尔滨工业大学、西安交通大学、中国人民大学、北京航空航天大学、北京邮电大学、中山大学、解放军理工大学、郑州大学、湖北工学院、中国国家信息安全测评认证中心等国内重点大学和科研机构在计算机的各个领域的著名学者组成“专家指导委员会”，为我们提供选题意见和出版监督。

这三套丛书是响应教育部提出的使用外版教材的号召，为国内高校的计算机及相关专业

的教学度身订造的。其中许多教材均已为M. I. T.，Stanford，U.C. Berkeley，C. M. U. 等世界名牌大学所采用。不仅涵盖了程序设计、数据结构、操作系统、计算机体系结构、数据库、编译原理、软件工程、图形学、通信与网络、离散数学等国内大学计算机专业普遍开设的核心课程，而且各具特色——有的出自语言设计者之手、有的历经三十年而不衰、有的已被全世界的几百所高校采用。在这些圆熟通博的名师大作的指引之下，读者必将在计算机科学的宫殿中由登堂而入室。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证，但我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。教材的出版只是我们的后续服务的起点。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方法如下：

电子邮件：hzedu@hzbook.com
联系电话：（010）68995264
联系地址：北京市西城区百万庄南街1号
邮政编码：100037

专家指导委员会

（按姓氏笔画顺序）

尤晋元	王　珊	冯博琴	史忠植	史美林
石教英	吕　建	孙玉芳	吴世忠	吴时霖
张立昂	李伟琴	李师贤	李建中	杨冬青
邵维忠	陆丽娜	陆鑫达	陈向群	周伯生
周克定	周傲英	孟小峰	岳丽华	范　明
郑国梁	施伯乐	钟玉琢	唐世渭	袁崇义
高传善	梅　宏	程　旭	程时端	谢希仁
裘宗燕	戴　葵			

译 者 序

随着速度的不断提高和存储容量的持续增长，计算机的功能日益强大，从而处理数据和解决问题的规模和复杂程度与日俱增。这不仅带来了需要认真研究的新课题，而且突出了原有数据结构和算法效率低下的缺点。程序的效率问题不是由于计算机功能的强大而受到冷落，相反地，倒是被人们提到前所未有的重视程度，因为大型问题的解决所涉及到的大容量存储和高速度运算容不得我们对效率有丝毫的忽视。本书正是在阐述数据结构基本概念的同时深入地分析了算法的效率。书中详细介绍了当前流行的论题和新的变化，讨论了算法设计技巧，并在研究算法的性能、效率以及对运行时间分析的基础上考查了一些高级数据结构，从历史的角度和近年的进展对数据结构的活跃领域进行了简要的概括。由于本书选材新颖，方法实用，题例丰富，取舍得当，因此，自从出版以来受到广泛欢迎，已被世界许多知名大学用作教材。

本书的目的是培养学生良好的程序设计技巧和熟练的算法分析能力，使得他们能够开发出高效率的程序。从服务于实践又锻炼学生实际能力出发，书中提供了大部分算法的C程序和伪码例程，但并不是全部。一些程序可从互联网上获得。

承蒙卢开澄教授、陈贤舜先生、温丽芳女士的鼓励，译者有幸将国外几部优秀原著介绍给我国的读者；蒋祎先生认真的工作使本书译文免除不少疏漏和错误；杨海玲女士的监督使翻译工作比预想的顺利。译者在此表示衷心的感谢。译者还愿意借此机会感谢挚友孙华先生，他对本书的翻译工作自始至终给予热心的关怀和无私的帮助。

由于时间及水平所限，书中译文不当之处，统祈学术界同仁及广大读者赐正。

译 者

2003年9月

前　言

目的/目标

本书讨论*数据结构*和*算法分析*。数据结构主要研究组织大量数据的方法，而算法分析则是对算法运行时间的评估。随着计算机的速度越来越快，对于能够处理大量输入数据的程序的需求变得日益急切。可是，由于在输入量很大的时候，程序的低效率现象变得非常明显，因此这又要求对效率问题给予更仔细的关注。通过在实际编程之前对算法的分析，学生可以决定一个特定的解法是否可行。例如，学生在本书中将读到一些特定的问题并看到精心的实现方法是如何把对大量数据的时间限制从 16 年减至不到 1 秒的。因此，若无运行时间的阐释，就不会有算法和数据结构的提出。在某些情况下，对于影响算法实现的运行时间的一些微小细节都需要认真地探究。

一旦解法被确定，程序还是必须要编写的。随着计算机的日益强大，它们必须解决的问题就变得更加巨大和复杂，这就要求开发更加复杂的程序。本书的目的是同时教授学生良好的程序设计技巧和算法分析能力，使得他们能够开发这样的具有最高效率的程序。

本书适合作为高级数据结构(CS7)课程或是研究生第一年算法分析课程的教材。学生应该具有中等程度的程序设计知识，包括像指针和递归这样一些内容，还要具有离散数学的某些知识。

方法

我相信，对于学生重要的是学习如何自己动手编写程序，而不是从书上拷贝程序。但另一方面，讨论现实程序设计问题而不套用样本程序实际上是不可能的。由于这个原因，本书通常提供实现方法的大约一半到四分之三的内容并鼓励学生补足其余的部分。第 12 章是这一版新加的一章，讨论主要侧重于实现细节的一些附加的数据结构。

本书中的算法均以 ANSI C 表示，尽管有些欠缺，但它仍然是最流行的系统程序设计语言。C 代替 Pascal 的使用使得动态分配数组成为可能(可见第 5 章中的“再散列”)。它还在几处地方将代码简化，这通常是因为与(&&)操作走捷径的缘故。

对 C 的大多数批评集中在用它写出的程序代码可读性差的事实上。仅仅少击几次键，却牺牲了程序清晰性，而程序的速度又没有增加。因此，诸如同时赋值以及通过

```
if ( x = y )
```

测试是否为 0 等技巧一般不在本书中使用。相信本书将证明只要细心练习是可以避免那些难以读懂的代码的。

内容提要

第 1 章包含离散数学和递归的一些复习材料。我相信对递归做到泰然处之的惟一办法是

反复不断地看一些好的用法。因此,除第 5 章外,递归遍及本书每一章的例子之中。

第 2 章处理算法分析。这一章阐述渐进分析和它的主要弱点。这里提供了许多例子,包括对对数运行时间的深入解释。通过直观地把一些简单递归程序转变成迭代程序而对它们进行分析。介绍了更为复杂的分治程序,不过有些分析(求解递归关系)要推迟到第 7 章再详细讨论。

第 3 章包括表、栈和队列。重点放在使用 ADT 对这些数据结构编程,这些数据结构的快速实现,以及介绍它们的某些用途上。课文中几乎没有什么程序(只有些例程),而程序设计作业的许多思想基本上体现在练习之中。

第 4 章讨论树,重点在查找树,包括外部查找树(B-树)。UNIX 文件系统和表达式树是作为例子来介绍的。AVL 树和伸展树只作了介绍而没有分析。程序写出 75%,其余部分留给学生完成。查找树的实现细节见第 12 章。树的另外一些内容,如文件压缩和博弈树,延迟到第 10 章讨论。外部媒体上的数据结构作为几章中的最后论题来讨论。

第 5 章是相对较短的一章,主要讨论散列表。这里进行了某些分析,本章末尾讨论了可扩散列。

第 6 章是关于优先队列的。二叉堆也在这里讲授,还有些附加的材料论述优先队列某些理论上有趣的实现方法。斐波那契堆在第 11 章讨论,配对堆在第 12 章讨论。

第 7 章讨论排序。它特别关注编程细节和分析。讨论并比较所有通用的排序算法。对以下四种算法详细地进行了分析:插入排序、希尔排序、堆排序以及快速排序。堆排序平均情形运行时间的分析对于这一版来说是新的内容。本章末尾讨论了外部排序。

第 8 章讨论不相交集算法并证明其运行时间。这是短且特殊的一章,如果不讨论 Kruskal 算法则该章可跳过。

第 9 章讲授图论算法。图论算法的重要性不仅因为它们在实践中经常用到,而且还因为它们的运行时间强烈地依赖于数据结构的恰当使用。实际上,所有标准算法都是和相应的数据结构、伪代码以及运行时间的分析一起介绍的。为把这些问题放进一本适当的教材中,我们对复杂性理论(包括 NP-完全性和不可判定性)进行了简短的讨论。

第 10 章通过考查一般的问题求解技巧讨论算法设计。这一章添加了大量的实例。这里及后面各章使用的伪代码使得学生更好地理解例子,而避免被实现的细节所困扰。

第 11 章处理摊还分析。对来自第 4 章到第 6 章的三种数据结构以及本章介绍的斐波那契堆进行了分析。

第 12 章是这一版新加的一章,讨论查找树算法、k-d 树和配对堆。不同于其他各章,本章给出了查找树和配对堆完全的仔细的实现。材料的安排使得教师可以把一些内容纳入到其他各章的讨论之中。例如,第 12 章中的自顶向下红黑树可以在(第 4 章的)AVL 树下讨论。

第 1 章到第 9 章为大多数的一学期数据结构课程提供了足够的材料。如果时间允许,那么第 10 章也可以包括进来。研究生的算法分析课程可以使用第 7 章到第 11 章的内容。第 11 章所分析的高级数据结构可以容易地在前面各章中查到。第 9 章中对 NP-完全性的讨论对于这门课来说太过简要,Garey 和 Johnson 的论 NP-完全性的书(有张立昂等翻译的中文译本:计算机和难解性,科学出版社,1987——译者注)可以补充本书的不足。

练习

在每章末尾提供的练习与书中讲授的内容顺序相匹配。最后的一些练习是针对整个一章而不是特定的某一节。难做的练习标以一个星号，更难的练习标有两个星号。

教师可从 Addison-Wesley 出版公司得到包含几乎所有练习答案的解题指南。

参考文献

参考文献位于每章的最后。一般说来，这些参考文献或者是历史性的，代表着书中材料的原始来源，或者阐述对书中给出的结果的扩展和改进。有些文献论述了一些练习的解法。

代码的获得

本书中的程序代码通过匿名 ftp 可在 aw.com 网站得到。这个网站也可以通过 World Wide Web 来访问；其 URL 为 http://www.aw.com/cseng/（从此处继续链接）。该资料的准确位置可能变化。

致谢

在本丛书几部著作的准备过程中，本人得到许多朋友的帮助。有些人在本书的其他版本中提到过；谢谢诸位。

对于这一版，我要感谢 Addison-Wesley 的编辑 Carter Shanklin 和 Susan Hartman。Teri Hyde 对此书做出了完美的工作，而 Matthew Harris 和他在 Publication Services 的同事出色地完成了本书最后的定稿任务。

M.A.W.
Miami，Florida
1996 年 7 月

目　录

第1章 引 论

在这一章，我们阐述本书的目的和目标并简要复习离散数学以及程序设计的一些概念。我们将要：

- 看到程序在较大输入情况下的运行性能与在适量输入情况下的运行性能具有同等重要性。
- 总结本书其余部分所需要的基本的数学基础。
- 简要复习递归。

1.1 本书讨论的内容

设有一组 N 个数而要确定其中第 k 个最大者。我们称之为选择问题(selection problem)。大多数学习过一两门程序设计课程的学生写一个解决这种问题的程序不会有什么困难。“显而易见的”解决方法有很多。

该问题的一种解法就是将这 N 个数读进一个数组中，再通过某种简单的算法，比如冒泡排序法，以递减顺序将数组排序，然后返回位置 k 上的元素。

稍微好一点的算法可以先把前 k 个元素读入数组并(以递减的顺序)对其排序。接着，将剩下的元素再逐个读入。当新元素被读到时，如果它小于数组中的第 k 个元素则忽略，否则就将其放到数组中正确的位置上，同时将数组中的一个元素挤出数组。当算法终止时，位于第 k 个位置上的元素作为答案返回。

这两种算法编码都很简单，建议读者试一试。此时我们自然要问：哪个算法更好？哪个算法更重要？还是两个算法都足够好？使用含有一百万个元素的随机文件，在 $k = 500\,000$ 的条件下进行模拟发现，两个算法在合理的时间量内均不能结束；每种算法都需要计算机处理若干天才能算完(虽然最后还是给出了正确的答案)。在第 7 章将讨论另一种算法，该算法将在一秒钟左右给出问题的解。因此，虽然我们提出的两个算法都能算出结果，但是它们不能被认为是好的算法，因为对于第三种算法在合理的时间内能够处理的输入数据量而言，这两种算法是完全不切实际的。

第二个问题是解决一个流行的字谜。输入是由一些字母和单词的二维数组组成。目标是要找出字谜中的单词，这些单词可能是水平、垂直或沿对角线以任何方向放置的。作为例子，图 1-1 所示的字谜由单词 this、two、fat 和 that 组成。单词 this 从第一行第一列的位置即(1, 1)处开始并延伸至(1, 4)；单词 two 从(1, 1)到(3, 1)；fat 从(4, 1)到(2, 3)；而 that 则从(4, 4)到(1, 1)。

	1	2	3	4
1	t	h	i	s
2	w	a	t	s
3	o	a	h	g
4	f	g	d	t

图 1-1 字谜示例

现在至少有两种直观的算法来求解这个问题。对单词表中的每个单词，我们检查每一个有序三元组(行，列，方向)，验证是否有单词存在。这需要大量嵌套的 for 循环，但它基本上是直观的算法。

也可以这样，对于每一个尚未进行到字谜最后的有序四元组(行，列，方向，字符数)我们可以测试所指的单词是否在单词表中。这也导致使用大量嵌套的for循环。如果在任意单

词中的最大字符数已知，那么该算法有可能节省一些时间。

上述两种方法相对来说都不难编码并可求解通常发表于杂志上的许多现实的字谜游戏。这些字谜通常有16行16列以及40个左右的单词。然而，假设我们把字谜变成为只给出谜板(puzzle board)而单词表基本上是一本英语词典，则上面提出的两种解法需要相当可观的时间来解决这个问题，故这两种方法都是不可接受的。不过，这样的问题还是有可能在数秒内解决的，即使单词表很大也可以。

在许多问题当中，一个重要的观念是：写出一个可以工作的程序并不够。如果这个程序在巨大的数据集上运行，那么运行时间就变成了重要的问题。我们将在本书中看到对于大量的输入，如何估计程序的运行时间，尤其是如何在尚未具体编码的情况下比较两个程序的运行时间。我们还将看到彻底改进程序速度以及确定程序瓶颈的方法。这些方法将使我们能够
2 找到需要大力优化的那些代码段。

1.2 数学知识复习

这一节列出一些需要记住或是能够推导出的基本公式，复习基本的证明方法。

1.2.1 指数

$$X^A X^B = X^{A+B}$$

$$\frac{X^A}{X^B} = X^{A-B}$$

$$(X^A)^B = X^{AB}$$

$$X^N + X^N = 2X^N \neq X^{2N}$$

$$2^N + 2^N = 2^{N+1}$$

1.2.2 对数

在计算机科学中，除非有特别的声明，所有的对数都是以2为底的。

定义：$X^A = B$，当且仅当 $\log_X B = A$

由该定义可以得到几个方便的等式。

定理 1.1

$$\log_A B = \frac{\log_C B}{\log_C A};\ C > 0$$

证明：

令 $X = \log_C B$，$Y = \log_C A$，以及 $Z = \log_A B$。此时由对数的定义得：$C^X = B$，$C^Y = A$ 以及 $A^Z = B$。联合这三个等式则产生 $(C^Y)^Z = C^X = B$。因此，$X = YZ$，这意味着 $Z = X/Y$，定理得证。

定理 1.2

$$\log AB = \log A + \log B$$

证明：

令 $X = \log A$，$Y = \log B$，以及 $Z = \log AB$。此时由于假设默认的底为2，$2^X = A$，$2^Y = B$ 及 $2^Z = AB$。联合最后的三个等式则有 $2^X 2^Y = 2^Z = AB$。因此 $X + Y = Z$，这就证明了该定理。

3 其他一些有用的公式如下，它们都能够用类似的方法推导。

$\log A/B = \log A - \log B$

$\log(A^B) = B \log A$

$\log X < X$（对所有的 $X>0$ 成立。）

$\log 1 = 0$，$\log 2 = 1$，$\log 1\,024 = 10$，$\log 1\,048\,576 = 20$。

1.2.3 级数

最容易记忆的公式是

$$\sum_{i=0}^{N} 2^i = 2^{N+1} - 1$$

和

$$\sum_{i=0}^{N} A^i = \frac{A^{N+1} - 1}{A - 1}$$

在第二个公式中，如果 $0<A<1$，则

$$\sum_{i=0}^{N} A^i \leqslant \frac{1}{1 - A}$$

当 N 趋于∞时该和趋向于 $1/(1-A)$，这些公式是“几何级数”公式。

我们可以用下面的方法推导关于 $\sum_{i=0}^{\infty} A^i (0 < A < 1)$ 的公式。令 S 表示和。此时

$$S = 1 + A + A^2 + A^3 + A^4 + A^5 + \cdots\cdots$$

于是

$$AS = A + A^2 + A^3 + A^4 + A^5 + \cdots\cdots$$

如果我们将这两个等式相减（这种运算只能对收敛级数进行），等号右边所有的项相消，只留下 1：

$$S - AS = 1$$

这就是说

$$S = \frac{1}{1 - A}$$

可以用相同的方法计算 $\sum_{i=1}^{\infty} i/2^i$，它是一个经常出现的和。我们写成

$$S = \frac{1}{2} + \frac{2}{2^2} + \frac{3}{2^3} + \frac{4}{2^4} + \frac{5}{2^5} + \cdots\cdots$$

用 2 乘之得到

$$2S = 1 + \frac{2}{2} + \frac{3}{2^2} + \frac{4}{2^3} + \frac{5}{2^4} + \frac{6}{2^5} + \cdots\cdots$$

4

将这两个方程相减得到

$$S = 1 + \frac{1}{2} + \frac{1}{2^2} + \frac{1}{2^3} + \frac{1}{2^4} + \frac{1}{2^5} + \cdots\cdots$$

因此，$S=2$。

分析中另一种常用类型的级数是算术级数。任何这样的级数都可以通过基本公式计算其值。

$$\sum_{i=1}^{N} i = \frac{N(N+1)}{2} \approx \frac{N^2}{2}$$

例如，为求出和 $2+5+8+\cdots+(3k-1)$，将其改写为 $3(1+2+3+\cdots+k)-(1+1+1+\cdots+1)$，显然，它就是 $3k(k+1)/2-k$。另一种记忆的方法则是将第一项与最后一项相加