



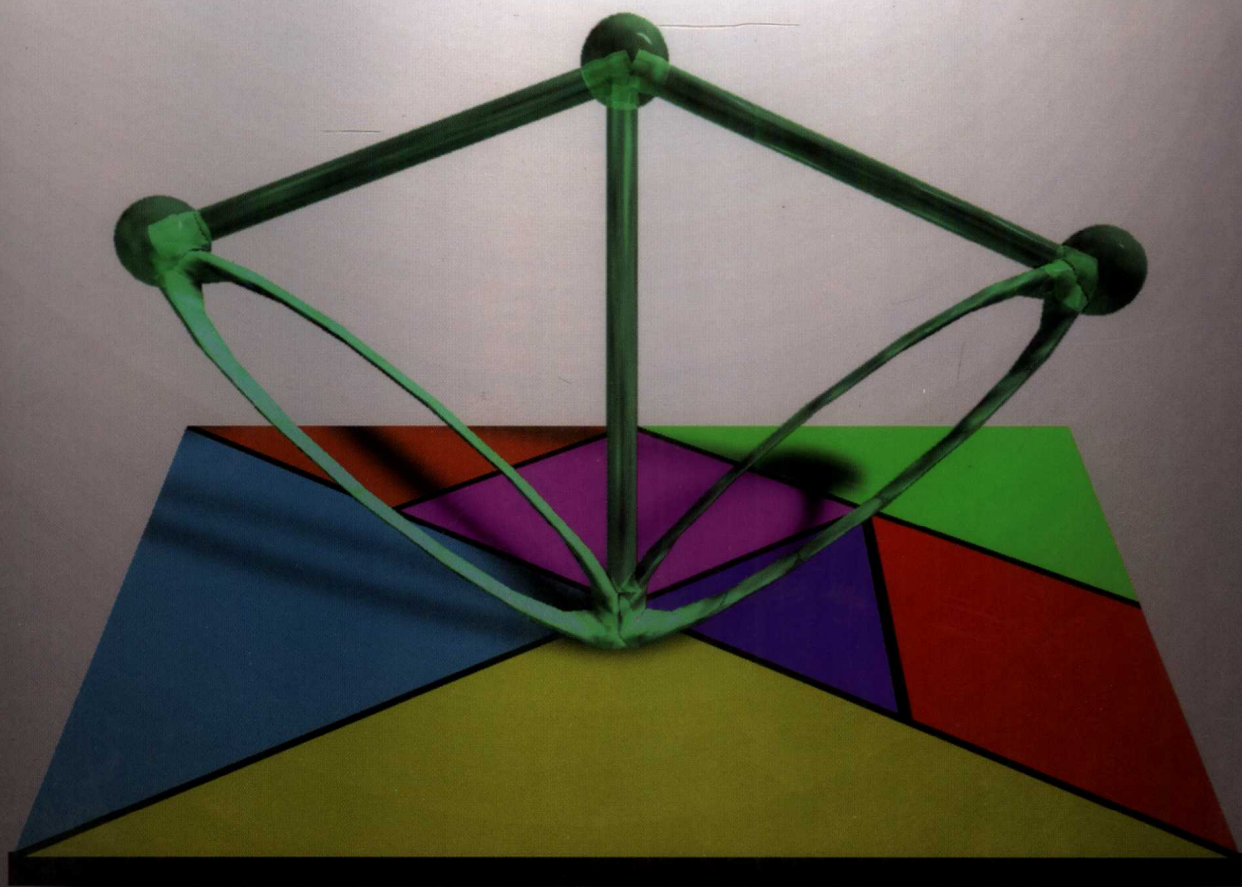
普通高等教育“十一五”国家级规划教材

计算机科学组合学丛书

组合数学

(第4版)

卢开澄 卢华明 编著



清华大学出版社

计算机科学组合学丛书

组 合 数 学

(第 4 版)

卢开澄 卢华明 编著

清 华 大 学 出 版 社
北 京

内 容 简 介

本书是《组合数学》第3版的修订版,全书共分8章,分别是:排列与组合、递推关系与母函数、容斥原理与鸽巢原理、Burnside引理与Polya定理、区组设计、线性规划、编码简介、组合算法简介。丰富的实例及理论和实际相结合是本书一大特点,有利于对问题的深入理解。

本书是计算机系本科生和研究生的教学用书,也可作为数学专业师生的教学参考书。

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 13501256678 13801310933

图书在版编目(CIP)数据

组合数学/卢开澄,卢华明编著. —4版. —北京:清华大学出版社,2006.12

(计算机科学组合学丛书)

ISBN 7-302-13961-X

I. 组… II. ①卢… ②卢… III. 组合数学 IV. O157

中国版本图书馆CIP数据核字(2006)第120643号

责任编辑:张 民 王冰飞

封面设计:卢开澄 傅瑞学

责任校对:白 蕾

责任印制:何 芊

出版发行:清华大学出版社

<http://www.tup.com.cn>

c-service@tup.tsinghua.edu.cn

社总机:010-62770175

投稿咨询:010-62772015

地 址:北京清华大学学研大厦A座

邮 编:100084

邮购热线:010-62786544

客户服务:010-62776969

印装者:山东鸿杰印务有限公司

经 销:全国新华书店

开 本:185×260 印张:20.25 字数:498千字

版 次:2006年12月第4版

印次:2006年12月第1次印刷

书 号:ISBN 7-302-13961-X/TP·8394

印 数:1~5000

定 价:29.00元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。
联系电话:010-62770177 转 3103 产品编号:021816-01

第4版序言

电子计算机的出现是20世纪最有影响的一件大事,它改变了整个世界的面貌,人们几乎无处不感到它的存在。哪个领域如果至今还宣称它与计算机线性无关,十之八九它已落后了。电子计算机使各种难题得以解决,但也萌生出更多的相关理论问题,在这种刺激和影响下,组合数学新军突起,一跃而成为最活跃的新数学分支,虽然它所讨论的问题和所使用的工具有的可追溯到二百多年前。有的组合学家将“计算机科学”定义为研究算法的科学,它为组合数学提供了活动的空间和舞台。组合数学(分析)是算法的理论基础,它与算法的关系犹如数学分析与计算方法的关系。作者认为这门课实际上是为学习“算法与复杂性分析”作理论的准备。图论是这个家族的主要成员,由于它已成长壮大,现已独立出去。

组合数学来源于实际,不少的讨论引人入胜。但初学者也往往有犯难的感觉。其实之所以觉得难,是因为还没弄懂,一旦明白了,则会恍然大悟而兴趣盎然。如果说学这门课有什么窍门,那就是从实际情况出发,以规模小的问题,模拟“沙盘推演”,寻找其规律性,然后推广及一般。

作者在实践中常有这样的体会:组合数学欲留给读者以和善可亲的形象,相比板着冷峻的面孔,要困难得多。解决方法是求助于实例。如果说法则是支撑肢体的框架,那么它将由丰富多彩的例子而丰满。本书在这方面,不论质和量都是一个亮点。不少问题饶有趣味,我们也常常为之而上下求索。第4版将依据作者近几年各自在教学实践中的经验,以怎样使读者更易接受作为出发点。对第3版的讲法和内容作了较大的更改,特别是第2章和第6、7、8章,几乎重写了,这部分主要由卢华明执笔。

前面已提到这门课为“算法与复杂性分析”作理论的准备,作者经验认为,计算机专业的本科生和研究生在学习第1~3章后继续学习第6~8章是一个不错的主意,以免有“空返”之憾。其他专业的学生则请酌情处理。

作者
2006年9月

目 录

第 1 章 排列与组合	1
1.1 加法法则与乘法法则	1
1.2 一一对应	5
1.3 排列与组合	8
1.3.1 排列与组合的模型	8
1.3.2 排列与组合问题的举例	9
1.4 圆周排列	14
1.5 排列的生成算法	15
1.5.1 序数法	15
1.5.2 字典序法	17
1.5.3 换位法	18
1.6 允许重复的组合与不相邻的组合	20
1.6.1 允许重复的组合	20
1.6.2 不相邻的组合	21
1.6.3 线性方程的整数解的个数问题	21
1.6.4 组合的生成	21
1.7 组合意义的解释	22
1.8 应用举例	28
1.9 Stirling 公式	35
* 1.9.1 Wallis 公式	35
* 1.9.2 Stirling 公式的证明	37
习题	38
第 2 章 递推关系与母函数	42
2.1 递推关系	42
2.2 母函数	43
2.3 Fibonacci 序列	46
2.3.1 Fibonacci 序列的递推关系	46
2.3.2 若干等式	47
2.4 优选法与 Fibonacci 序列的应用	48
2.4.1 优选法	48
2.4.2 优选法的步骤	50
2.4.3 Fibonacci 的应用	50
2.5 母函数的性质	51

2.6	线性常系数齐次递推关系	54
2.7	关于线性常系数非齐次递推关系	61
2.8	整数的拆分	67
2.9	Ferrers 图像	70
2.10	拆分数估计	73
2.11	指数型母函数	75
2.11.1	问题的提出	75
2.11.2	指数型母函数的定义	76
2.12	广义二项式定理	77
2.13	应用举例	80
2.14	非线性递推关系举例	99
2.14.1	Stirling 数	99
2.14.2	Catalan 数	104
2.14.3	举例	108
2.15	递推关系解法的补充	111
	习题	113
第 3 章	容斥原理与鸽巢原理	119
3.1	De Morgan 定理	119
3.2	容斥定理	120
3.3	容斥原理举例	123
3.4	棋盘多项式与有限制条件的排列	128
3.5	有禁区的排列	131
3.6	广义的容斥原理	133
3.6.1	容斥原理的推广	133
3.6.2	一般公式	134
3.7	广义容斥原理的应用	137
3.8	第二类 Stirling 数的展开式	140
3.9	欧拉函数 $\phi(n)$	141
3.10	n 对夫妻问题	142
3.11	Möbius 反演定理	142
3.12	鸽巢原理	145
3.13	鸽巢原理举例	146
3.14	鸽巢原理的推广	149
3.14.1	推广形式之一	149
3.14.2	应用举例	149
3.14.3	推广形式之二	154
3.15	Ramsey 数	155
3.15.1	Ramsey 问题	155
3.15.2	Ramsey 数	158

习题	161
第 4 章 Burnside 引理与 Pólya 定理	167
4.1 群的概念	167
4.1.1 定义	167
4.1.2 群的基本性质	168
4.2 置换群	170
4.3 循环、奇循环与偶循环	174
4.4 Burnside 引理	178
4.4.1 若干概念	178
4.4.2 重要定理	180
4.4.3 举例说明	183
4.5 Pólya 定理	185
4.6 举例	187
4.7 母函数形式的 Pólya 定理	193
4.8 图的计数	196
4.9 Pólya 定理的若干推广	200
习题	203
第 5 章 区组设计	206
5.1 问题的提出	206
5.2 拉丁方与正交的拉丁方	207
5.2.1 问题的引入	207
5.2.2 正交拉丁方及其性质	208
5.3 域的概念	209
5.4 Galois 域 $GF(p^m)$	211
5.5 正交拉丁方的构造	214
5.6 正交拉丁方的应用举例	216
5.7 均衡不完全的区组设计	217
5.7.1 基本概念	217
5.7.2 (b, v, r, k, λ) -设计	218
5.8 区组设计的构成方法	221
5.9 Steiner 三元素	223
5.10 Kirkman 女生问题	225
习题	226
第 6 章 线性规划	228
6.1 问题的提出	228
6.2 线性规划的问题	230
6.3 凸集	230
6.4 线性规划的几何意义	231

6.5	单纯形法的理论基础	233
6.5.1	松弛变量	233
6.5.2	解的充要条件	234
6.6	单纯形法与单纯形表格	238
6.7	改善的单纯形法	245
6.8	对偶概念	247
6.9	对偶单纯形法	253
	习题	258
第7章	编码简介	260
7.1	基本概念	260
7.2	对称二元信道	261
7.3	纠错码	262
7.3.1	最近邻法则	262
7.3.2	Hamming 不等式	263
7.4	若干简单的编码	264
7.4.1	重复码	264
7.4.2	奇偶校验码	264
7.5	线性码	265
7.5.1	生成矩阵与校验矩阵	265
7.5.2	关于生成矩阵和校验矩阵的定理	268
7.5.3	译码步骤	268
7.6	Hamming 码	269
7.7	BCH 码	270
	习题	273
第8章	组合算法简介	276
8.1	归并排序	276
8.1.1	算法	276
8.1.2	举例	277
8.1.3	复杂性分析	277
8.2	快速排序	278
8.2.1	算法的描述	279
8.2.2	复杂性分析	280
8.3	Ford-Johnson 排序法	281
8.4	排序的复杂性下界	283
8.5	求第 k 个元素	284
8.6	排序网络	286
8.6.1	0-1 原理	287
8.6.2	B_n 网络	287
8.6.3	复杂性分析	289

8.6.4	Batcher 奇偶归并网络	289
8.7	快速傅里叶变换	290
8.7.1	问题的提出	290
8.7.2	预备定理	291
8.7.3	快速算法	292
8.7.4	复杂性分析	294
8.8	DFS 算法	295
8.9	BFS 算法	296
8.10	$\alpha\beta$ 剪技术	297
8.11	状态与图	298
8.12	分支定界法	300
8.12.1	TSM 问题	300
8.12.2	任务安排问题	303
8.13	最短树与 Kruskal 算法	305
8.14	Huffman 树	305
8.15	多段判决	307
8.15.1	问题的提出	307
8.15.2	最佳原理	309
8.15.3	矩阵链积问题	309
8.15.4	图的两点间最短路径	310
习题		311

第 1 章 排列与组合

1.1 加法法则与乘法法则

组合数学的最主要内容是对离散对象的计数,计数时经常要用到两个基本法则,即加法法则和乘法法则.在以后讨论中不特别说明都假定 A 和 B 是性质无关的两类事件.

1. 加法法则

若具有性质 A 的事件有 m 个,具有性质 B 的事件有 n 个,则具有性质 A 或性质 B 的事件有 $m+n$ 个.

例如事件 A 是大于 0 而小于 10 的偶数,即 $A=\{2,4,6,8\}$;事件 B 是大于 0 而小于 10 的奇数,即 $B=\{1,3,5,7,9\}$. 具有性质 A 的事件数 $m=4$,具有性质 B 的事件数 $n=5$. 具有性质 A 和性质 B 的事件,即大于 0 而小于 10 的正整数数目等于 $4+5=9$.

2. 乘法法则

若具有性质 A 的事件有 m 个,具有性质 B 的事件有 n 个,则具有性质 A 及性质 B 的事件有 mn 个.

例 1-1 设某 BASIC 语言限制标识符,最多由三个字符组成,要求第 1 个字符必须是 26 个英文字母中的一个,第 2、3 字符可以是英文字母,也可以是阿拉伯数字 0,1,2,3,4,...,9. 求标识符的数目.

标识符的构成,可以是一个字符,或两个字符,最多三个字符.

一个字符组成时只能是 26 个英文字母.

两个字符组成时第 1 个字符是 26 个英文字母之一,第 2 个字符可能是 26 个英文字母及 10 个阿拉伯数字,共 36 个. 根据乘法法则两个字符构成的标识符有

$$26 \times 36 = 936$$

三个字符构成的标识符,其中第 1 个字符有 26 个,第 2 个和第 3 个字符有 36 个,故三个字符构成的标识符数为

$$26 \times 36 \times 36 = 33696$$

根据加法法则,最多由三个字符构成的标识符数目为

$$N = 26 + 936 + 33696 = 34658$$

例 1-2 求小于 10000 的正整数中含有数字 1 的数的个数.

小于 10000 的正整数可以看成由 0,1,2,...,8,9 中取四个数构成的. 但 0000 不在正整数范围内,故小于 10000 的正整数数目根据乘法法则应为 $10^4 - 1 = 9999$ 个.

同理四位中不含 1 的数的数目应为

$$9^4 - 1 = 6561 - 1 = 6560$$

故小于 10000 并含有 1 的数的个数为

$$9999 - 6560 = 3439$$

本题解法不是直接去求含有 1 的数的数目,而是求不含 1 的数的数目. 全体减去不含 1

的数剩下的就是含有数 1 的数的个数.

例 1-3 长度为 n 的 0,1 符号串的数目为 2^n .

假定 $a = a_1 a_2 \cdots a_n, a_i \in [0, 1], i = 1, 2, \cdots, n$, 根据乘法法则立即可得 0,1 符号串的个数为 $2^n = \underbrace{2 \cdot 2 \cdot \cdots \cdot 2}_{n\uparrow}$, 例如 $n=3$, 有

000, 001, 010, 011, 100, 101, 110, 111.

或表示为图 1-1.

例 1-4 遗传物质 DNA 是由 T, C, A, G 四种化合物组成的链, 由 T, C, A, G 它们的不同排列顺序确定遗传信息, 称之为遗传编码. 在遗传学上 T 为胸腺嘧啶, C 为胞嘧啶, A 为腺嘌呤, G 为鸟嘌呤.

人类的 DNA 链长度为 2.1×10^{10} . 根据乘法法则, 人类的 DNA 链的数目 N 为

$$N = 4^{2.1 \times 10^{10}} = (4^{2.1})^{10^{10}},$$

但

$$4^{2.1} = 18.379 > 10^{1.26},$$

故

$$N > (10^{10^{1.26}})^{1.26}.$$

例 1-5 n 元布尔函数 $f(x_1, x_2, \cdots, x_n)$ 的数目.

n 个布尔目变元 $x_1, x_2, \cdots, x_n, x_i = (0, 1), i = 1, 2, \cdots, n$, 根据乘法法则可能有 $\underbrace{2 \cdot 2 \cdot \cdots \cdot 2}_{n\text{项}} = 2^n$ 种指派, 令 2^n 个不同指派为

$$a_1, a_2, \cdots, a_{2^n}$$

对每个指派, 布尔函数取值为 (0, 1), 故不同的布尔函数的数目为

$$\underbrace{2 \cdot 2 \cdot \cdots \cdot 2}_{2^n} = 2^{2^n}.$$

以 $n=2$ 为例, (x_1, x_2) 有 (0, 0), (0, 1), (1, 0), (1, 1), 四种指派, 而 $f(x_1, x_2)$ 有 $2^4 = 16$ 种可能. 相同的 (x_1, x_2) , 而函数值不全相同, 表示为不相同的布尔函数, 见表 1-1.

表 1-1

x_1	x_2	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1
$f_i =$		0	$x_1 \wedge x_2$	$x_1 \wedge \bar{x}_2$	x_1	$\bar{x}_1 \wedge x_2$	x_2	$(\bar{x}_1 \vee \bar{x}_2) \wedge (x_1 \vee x_2)$	$x_1 \vee x_2$

x_1	x_2	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1
$f_i =$		$\bar{x}_1 \wedge \bar{x}_2$	$(\bar{x}_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$	$\bar{x}_2$	$x_1 \vee \bar{x}_2$	$\bar{x}_1$	$\bar{x}_1 \vee x_2$	$\bar{x}_1 \vee \bar{x}_2$	1

其中 $\bar{1}=0, \bar{0}=1$, 并满足:

$\vee$	0	1		$\wedge$	0	1
0	0	1	,	0	0	0
1	1	1		1	0	1

例如 $(\bar{x}_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$ 列表计算如表 1-2.

表 1-2

x_1	x_2	$\bar{x}_1$	$\bar{x}_2$	$\bar{x}_1 \vee x_2$	$x_1 \vee \bar{x}_2$	$(\bar{x}_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$	
0	0	1	1	1	1	1	
0	1	1	0	1	0	0	
1	0	0	1	0	1	0	
1	1	0	0	1	1	1	

例 1-6 $n=7^3 \times 11^2 \times 13^4$, 求除尽 n 的整数个数.

显然能除尽 n 的整数是

$$7^{l_1} \times 11^{l_2} \times 13^{l_3}, \quad 0 \leq l_1 \leq 3, 0 \leq l_2 \leq 2, 0 \leq l_3 \leq 4.$$

根据乘法法则, 能除尽 n 的数的数目为

$$4 \times 3 \times 5 = 60$$

例 1-7 有 a, b, c, d, e 这 5 个字, 从中取 6 个构成一组字符串. 要求: (1) 第 1 个和第 6 个必须是子音 b, c, d ; (2) 每一字符串都必有 a, e 两个母音, 且 a, e 不相邻; (3) 相邻两子音不允许相同. 求字符串的数目.

根据要求, 两个母音 a, e 有以下 3 种格式: (1) 即 a, e 位于第 2 和第 4 位, 或记为 (2, 4) 格式; (2) (2, 5) 格式; (3) (3, 5) 格式.

$$\begin{aligned} (2, 4): & \quad \cdot \quad \bigcirc \quad \cdot \quad \bigcirc \quad \cdot \quad \cdot \\ (2, 5): & \quad \cdot \quad \bigcirc \quad \cdot \quad \cdot \quad \bigcirc \quad \cdot \\ (3, 5): & \quad \cdot \quad \cdot \quad \bigcirc \quad \cdot \quad \bigcirc \quad \cdot \end{aligned}$$

其中 $\cdot$ 表示子音, $\bigcirc$ 表示母音. 单个子音有 3 种选择, 一对子音字符只有 3×2 种选择.

(2, 4) 格式有 $3 \times 2 \times 3 \times 2 \times 3 \times 2 = 3^3 \times 2^3$ 种方案,

(2, 5) 格式有 $3 \times 2 \times 3 \times 2 \times 2 \times 3 = 3^3 \times 2^3$ 种方案,

(3, 5) 格式有 $3 \times 2 \times 2 \times 3 \times 3 \times 2 = 3^3 \times 2^3$ 种方案.

根据加法法则所求字符串的数目 n 有

$$n = 3 \times 3^3 \times 2^3 = 3^4 \times 2^3 = 81 \times 8 = 648$$

例 1-8 有日文书 5 册, 英文书 7 册, 中文书 10 册, 若从中取两册不同文字的书有几种可能? 若取两册是相同文字的又有多少种方案? 若取两本不论是什么文字的又有几种方案?

(1) 取两本不同文字的书, 根据乘法法则有:

① 英、日各一册的有 $n_1 = 5 \times 7 = 35$ 种可能;

② 中、日各一册的有 $n_2 = 10 \times 5 = 50$ 种可能;

③ 中、英各一册的有 $n_3 = 10 \times 7 = 70$ 种可能.

再根据加法法则取两册不同文字的书的方案数

$$N_1 = 35 + 50 + 70 = 155 \text{ 种}$$

(2) 取两册书是相同文字的方案数分别有:

① 同为中文: $n'_1 = (10 \times 9) / 2 = 45$.

第一本有 10 种可能,第二本有 9 种可能,但第一本为 a ,第二本为 b ,和第一本为 b ,第二本为 a 情况相同,重复计算一次,故除以 2.

② 同为英文: $n'_2 = 7 \times 6 / 2 = 21$,

③ 同为日文: $n'_3 = 5 \times 4 / 2 = 10$.

根据加法法则取两册,为同文字的方案数

$$N_2 = 45 + 21 + 10 = 76 \text{ 种}$$

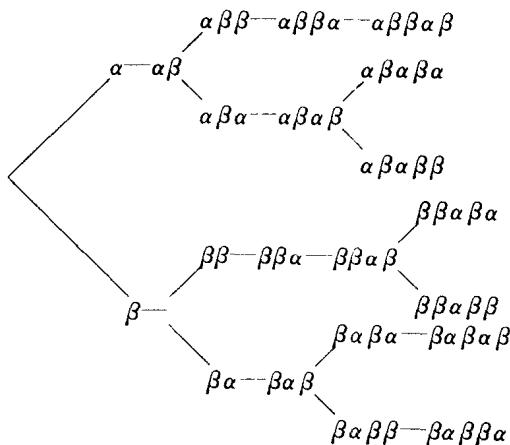
(3) 不考虑文字相同与否, 根据加法法则

$$N_3 = 155 + 76 = 231 \text{ 种,}$$

或从 22 本书取两册有 $22 \times 21 / 2 = 231$ 种.

例 1-9 由 26 个英文字母构成长度为 5 的字符串, 要求 (1) 6 个母音 a, e, i, o, u, y 不相邻; (2) 其余 20 个子音不存在 3 个相邻; (3) 相邻的子音不相同. 试求有多少个这样的字符串.

用 α 表示 a, e, i, o, u, y 的集合, β 表示其余 20 个子音的集合, 根据要求, 这样的字符串其结构可表示为:



根据乘法法则有:

(1) $\alpha\beta\beta\alpha\beta$ 型

$$n_1 = 6 \times 20 \times 19 \times 6 \times 20 = 6^2 \times 20^2 \times 19;$$

(2) $\alpha\beta\alpha\beta\alpha$ 型

$$n_2 = 6 \times 20 \times 6 \times 20 \times 6 = 6^3 \times 20^2;$$

(3) $\alpha\beta\alpha\beta\beta$ 型

$$n_3 = 6 \times 20 \times 6 \times 20 \times 19 = 6^2 \times 20^2 \times 19.$$

(4) $\beta\beta\alpha\beta\alpha$ 型

$$n_4 = 20 \times 19 \times 6 \times 20 \times 6 = 6^2 \times 20^2 \times 19;$$

(5) $\beta\beta\alpha\beta\beta$ 型

$$n_5 = 20 \times 19 \times 6 \times 20 \times 19 = 6 \times (20 \times 19)^2;$$

(6) $\beta\alpha\beta\alpha\beta$ 型

$$n_6 = 20 \times 6 \times 20 \times 6 \times 20 = 6^2 \times 20^3;$$

(7) $\beta\alpha\beta\beta\alpha$ 型

$$n_7 = 20 \times 6 \times 20 \times 19 \times 6 = 6^2 \times 20^2 \times 19.$$

根据加法法则

$$\begin{aligned} n &= n_1 + n_2 + n_3 + n_4 + n_5 + n_6 + n_7 \\ &= 6 \times 20^2 [6 \times 19 + 6^2 + 6 \times 19 + 6 \times 19 + 19^2 + 6 \times 20 + 6 \times 19] \\ &= 6 \times 20^2 [24 \times 19 + 6^2 + 6 \times 20] \\ &= 6 \times 20^2 \times 973 = 2335200. \end{aligned}$$

1.2 一一对应

一一对应是计数时常用的一种技巧,若性质 A 的计数比较困难,性质 B 的计数比较容易,但性质 A 和性质 B 一一对应,则对 A 的计数可转换为对性质 B 的计数.

最有趣而且直观的一个例子,比如有 100 位乒乓球选手通过淘汰赛,最后产生一名冠军,先分 50 对比赛,第一轮结束留下 50 名胜利者.第二轮将 50 名第一轮胜出的选手分成 25 对进行比赛,25 名胜利者参加第三轮比赛,分 12 组,其中一人直接参加第四轮比赛.第四轮开始时 13 名选手,分 6 对比赛,一人直接进入第五轮,第五轮有 7 人,分 3 组,同样有一个直接进入第六轮;第六轮有四人,分两组,其中胜者参加最后的决赛.现统计共进行多少对比赛,总比赛台数 $50 + 25 + 12 + 6 + 3 + 2 + 1 = 99$.

其实比赛的台数和每一场比赛淘汰一名选手一一对应,100 名选手要选出一名单打冠军,必须淘汰 99 名,故必须进行 99 台比赛.1000 位选手要选出一名单打冠军,不必犹豫回答要进行 999 台比赛.

例 1-10 碳氢化合物 C_nH_{2n+2} , 随 n 的不同有如下不同的支链,见图 1-2.

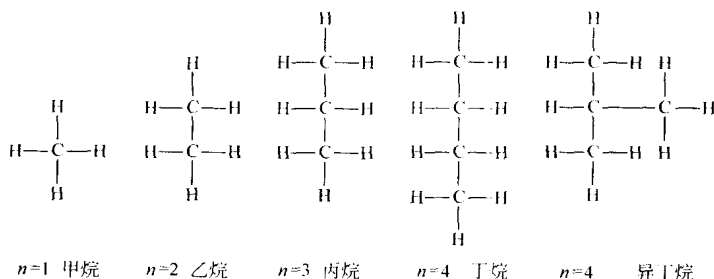


图 1-2

这说明 C_nH_{2n+2} 的每一支链和一棵有 $3n+2$ 个顶点的树一一对应,要求树叶的数目为 $2n+2$,内点的数目为 n ,内点的线度为 4. 比如图 1-2 的 $n=4$,这样的树有两个,一个对应丁烷,一个对应异丁烷.构造碳氢化合物 C_nH_{2n+2} 的问题可转换为图论问题.只要计算符合上述条件的树的数目,便可确定对应的不同化合物的数目,而且列举满足条件的树的结构便可提供对应化合物支链的构造.例如 $n=3$ 的 C_3H_8 ,图 1-3 有两个图,从图论的观点看是同构,

他们的相关位置一样,即他们点和边的连接关系是一致的.

定理 1-1(Cayley) 过 n 个有标志顶点的树的数目等于 n^{n-2} .

此定理说明用 $n-1$ 条边将 n 个已知的顶点连接起来的连通图的个数为 n^{n-2} . 也可以这样理解,将 n 个城市连接起来的树状公路网络有 n^{n-2} 种可能方案. 所谓树状,指的是用 $n-1$ 条边将 n 个顶点构成一个连通图. 当然,建造一个树状公路网将 n 个城市连接起来,应求其中长度最短,造价最省的一种,或效益最大的一种. Cayley 定理只是说明可能方案的数目.

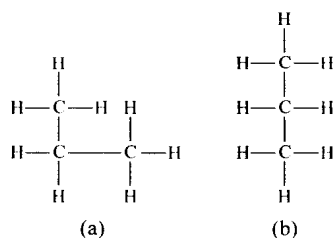


图 1-3

Cayley 定理的证明方法有许多,下面采用最聪明的一一对应法. 不失一般性,假定已知的 n 个顶点标志为 $1, 2, \dots, n$.

假定 T 是其中一棵树,树叶中有标号最小者,设为 a_1 , a_1 的邻接点为 b_1 ,从图中消去 a_1 点和边 (a_1, b_1) . b_1 点便成为消去后余下的树 T_1 的顶点. 在余下的树 T_1 中寻找标号最小的树叶,设为 a_2 , a_2 的邻接点为 b_2 ,从 T_1 中消去 a_2 及边 (a_2, b_2) . 如此步骤继续 $n-2$ 次,直到最后剩下一条边为止. 于是一棵树 T 对应一序列

$$b_1, b_2, \dots, b_{n-2}$$

$b_1, b_2, \dots, b_{n-2}$ 是 1 到 n 中的数,并且允许重复.

反过来从 $b_1 b_2 \dots b_{n-2}$ 可以恢复树 T 本身,因为消去的是树叶中标号最小的,而且它和 b_i 是邻接的. 即给出一序列

$$b_1, b_2, \dots, b_{n-2}$$

其中 $1 \leq b_i \leq n, i=1, 2, \dots, n-2$. 可恢复与之对应的树,方法如下:有两个序列,一个是

$$1, 2, \dots, n \quad (1-1)$$

另一个是

$$b_1, b_2, \dots, b_{n-2} \quad (1-2)$$

在序列(1-1)中找出第一个不出现在序列(1-2) $b_1, b_2, \dots, b_{n-2}$ 中的数,这个数显然便是 a_1 , 同时形成边 (a_1, b_1) , 并从(1-1)中消去 a_1 , 从(1-2)中消去 b_1 . 在余下的(1-1), (1-2)序列中继续以上的步骤 $n-2$ 次,直到序列(1-2)成为空集为止. 这时序列(1-1)剩下两个数 a_k, b_k . 边 (a_k, b_k) 是树 T 的最后一条边.

例 1-11 $n=6$, 由图 1-4 根据上述的步骤可得序列 $b_1=b_2=b_3=b_4=1$, 即图 1-4 对应序列 1 1 1 1. 这个过程很容易,请读者自己来做做看. 这是一个极端的例子,反过来已知

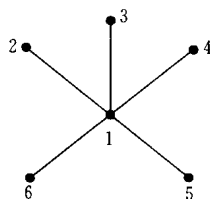


图 1-4

恢复树 T .

$$\begin{cases} (1, 2, 3, 4, 5, 6) & \text{序列(1)} \\ (1, 1, 1, 1), & \text{序列(2)} \end{cases}$$

序列(1)中第一个不在序列(2)中出现的数是 2. 故 $(2, 1)$ 是树的一条边. 从序列(1)中消去 2, 从序列(2)中消去第 1 个数 1 并继续以上步骤.

$$\begin{cases} (1, 3, 4, 5, 6) \\ (1, 1, 1) \end{cases} \Rightarrow \begin{cases} (1, 4, 5, 6) \\ (1, 1) \end{cases} \quad (3, 1) \text{ 是树的一条边.}$$

$$\begin{cases} (1, 4, 5, 6) \\ (1, 1) \end{cases} \Rightarrow \begin{cases} (1, 5, 6) \\ (1) \end{cases} \quad (4, 1) \text{ 是 } T \text{ 树的一条边,}$$

$$\begin{cases} (1, 5, 6) \\ (1) \end{cases} \Rightarrow \begin{cases} (1, 6) \\ \phi \end{cases} \quad (5, 1) \text{ 是树的边. } (1, 6) \text{ 是 } T \text{ 的最后一边.}$$

例 1-12 以图 1-5 为例说明以上的步骤.

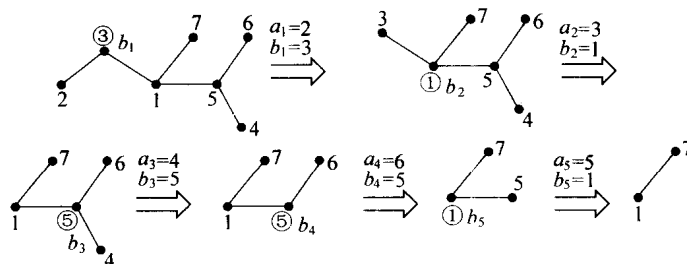


图 1-5

得 $(b_1, b_2, b_3, b_4, b_5) = (3, 1, 5, 5, 1), n = 7$.

反过来从 $(3, 1, 5, 5, 1)$ 恢复树 T ,

$$\begin{aligned} & \begin{cases} (1, 2, 3, 4, 5, 6, 7) \\ ((3), 1, 5, 5, 1) \end{cases} \Rightarrow \begin{cases} (1, 3, 4, 5, 6, 7) \\ ((1), 5, 5, 1) \end{cases} \Rightarrow \begin{cases} (1, 4, 5, 6, 7) \\ ((5), 5, 1) \end{cases} \Rightarrow \begin{cases} (1, 5, 6, 7) \\ ((5), 1) \end{cases} \Rightarrow \\ & \begin{cases} (1, 5, 7) \\ ((1)) \end{cases} \Rightarrow \begin{cases} (1, 7) \\ \phi \end{cases} \end{aligned}$$

图 1-6 中假定已知各顶点的标号, 有标志 * 的边是新加的边.

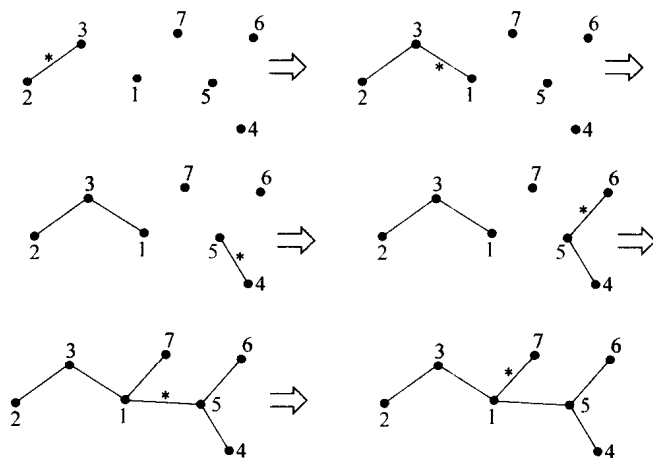


图 1-6

上面的讨论解决了由过 n 个已知标号的顶点的树, 可推出序列 $b_1, b_2, \dots, b_{n-2}, 1 \leq b_i \leq n, i = 1, 2, \dots, n-2$. 反过来从已知序列: $b_1, b_2, \dots, b_{n-2}$, 其中 $1 \leq b_i \leq n, i = 1, 2, \dots, n-2$, 可推出一个过 n 个已知标号的顶点的树, 即过 n 个已知标号的顶点的树和序列 $b_1 b_2 \dots b_{n-2}$ 一一对应, 根据乘法法则可得, 过 n 个有标号的顶点的树的数目, 由于 $1 \leq b_i \leq n, i = 1, 2, \dots, n-2$, 故为 n^{n-2} 个.

Cayley 定理的证明过程实际上是提供了构造过 n 个有标号顶点的树的方法.

1.3 排列与组合

1.3.1 排列与组合的模型

从 n 个元素中任取 r 个元素一组,若不考虑他们的顺序时,则称为从 n 中取 r 的组合,它的方案数以 $C(n, r)$ 或 $\binom{n}{r}$ 表示.

n 个元素中取 r 个按顺序排成一列,称为从 n 中取 r 的排列,其排列的方案数以 $P(n, r)$ 表示.

例如从 $\{A, B, C, D\}$ 中取三个为一组,可有

$$(A, B, C), (A, B, D), (A, C, D), (B, C, D)$$

四个组,故 $C(4, 3) = 4$.

将 (A, B, C) 进行排列可有 $ABC, ACB, BAC, BCA, CAB, CBA$;

将 (A, B, D) 进行排列可有 $ABD, ADB, BAD, BDA, DAB, DBA$;

将 (A, C, D) 进行排列可有 $ACD, ADC, CAD, CDA, DAC, DCA$;

将 (B, C, D) 进行排列可有 $BCD, BDC, CBD, CDB, DBC, DCB$.

故 $P(4, 3) = 24$.

从 n 中取 r 个排列的模型:可以看作是 r 个盒是有标志的, n 个球也是有区别的,取 r 个球放进盒子,且无一空盒.

$$\boxed{(1)} \quad \boxed{(2)} \quad \boxed{(3)} \quad \cdots \quad \boxed{(r)}$$

不失一般性,记球的标志为 $1, 2, \dots, n$. 取一球放进第 1 个盒子,有 n 种选择可能. 再从余下的 $n-1$ 个球中取一球放进第 2 个盒子,有 $n-1$ 种选择可能,以此类推,最后第 r 次从 $n-r+1$ 个球中取 1 球放进第 r 个盒子,有 $n-r+1$ 种可能选择,根据乘法法则有

$$\begin{aligned} P(n, r) &= n(n-1)(n-2)\cdots(n-r+1) \\ &= \frac{n!}{(n-r)!} \end{aligned}$$

取 n 个有标志的球的全体进行排列,称为 n 的全排列,其排列数为

$$n!$$

这里补充定义

$$0! = 1.$$

从 n 个元素中取 r 个进行组合的模:例如 n 个球是有标志的, r 个盒子是无区别的,从 n 个球中取 r 个装进盒子,每盒一球,无一空盒.

排列与组合的模型的区别在于盒子,排列的盒子有区别,组合的盒子无区别. 放进 r 个盒子的球的全排列为 $r!$,这就是组合的重复度,故

$$C(n, r) = \frac{P(n, r)}{r!} = \frac{n!}{r!(n-r)!}$$