# 组合数学

冯巾松
fengjinsong@tongji.edu.cn

# 第1章：排列与组合


1.1 加法法则与乘法法则

1.2 一一对应

1.3 排列与组合

1.4 圆周排列

1.5 排列的生成算法

1.6 允许重复的组合与不相邻的组合

1.7 组合意义的解释

1.8 应用举例

1.9 *Stirling公式

# 1.1 加法法则与乘法法则

## 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B**的事件有m+n个。

**A和B是性质无关的两个事件。**
**m和n之间没有交集的**

例:若从南京到杭州，可以坐飞机，也可以坐火车前往。飞机有3趟，火车有4趟。那么从南京到杭州有多少种走法?

# 1.1 加法法则与乘法法则

## 2、乘法法则:

若具有性质A的事件有m个，具有中文性质B的事件有n个，则具有性质A及B的事件有mn个

例: 若从南京到杭州自驾。南京到上海有3条路可走，从上海到杭州有2条路可走，问从南京经上海到杭州有多少路可走?

# 1.1 加法法则与乘法法则

两法则比较:

1、加法法则:

分类 （每一类都能达到目的）

2、乘法法则:

分步 （单独每一步都不能达到目的）

# 1.1 加法法则与乘法法则

乘法法则是加法法则的一个推论

1、加法法则：集合S划分成$S_1S_2S_3\ldots S_m$，m个子集，且 $S_i \cap S_j = \phi$ $(i \neq j)$ 则 $|S|=|S_1|+|S_2|+\ldots+|S_m|$

2、乘法法则: $S_1=\{(a_1, b_1),(a_1, b_2),\ldots,(a_1, b_n)\}$

$S_2=\{(a_2, b_1),(a_2, b_2),\ldots,(a_2, b_n)\}$

…

$S_m=\{(a_m, b_1),(a_m, b_2),\ldots,(a_m, b_n)\}$

$|S|=|S_1|+|S_2|+\ldots+|S_m|= n+n+\ldots+n = mn$

## 1.1 加法法则与乘法法则

例1-3　长度为n的0, 1符号串的个数有多少?

解：长度为n的字符串形式为：

$a_1a_2\cdots a_n$，　$a_i \in \{0, 1\}$, $i=1, 2, \cdots n$

$a_1$位置上可以有0或1这2种可能。同理，$a_2\cdots a_n$位置上都可以有0或1这2种可能。由乘法法则，长度为n的0, 1符号串的数目是

$$2 \bullet 2 \bullet 2 \ldots 2 = 2^n$$

# 1.1 加法法则与乘法法则

例1-4　人类DNA链的长度为$2.1\times10^{10}$,链上每一位由T,C,A,G四种化合物组成,求人类DNA链的可组成数目。

解：人类DNA链的长度为$2.1\times10^{10}$,根据乘法法则,类DNA链的可组成数目$N = 4^{2.1\times\mathbf{10}^{\mathbf{10}}}$。

# 1.1 加法法则与乘法法则

例1.5：求n元布尔函数$f(x_1, x_2, \ldots, x_n)$的数目？

以n=2为例：$f(x_1, x_2)$有四（$2^2$）种方式：

$f(0,0), f(0,1), f(1,0), f(1,1)$。

1、$f(0,0)=0, f(0,1)=0, f(1,0)=0, f(1,1)=0$。

2、$f(0,0)=0, f(0,1)=0, f(1,0)=0, f(1,1)=1$。

…………

16、$f(0,0)=1, f(0,1)=1, f(1,0)=1, f(1,1)=1$。

对应着长度为$2^2$的字符串，每一位都可以取0或1；

$$\begin{array}{c|ccc} & a_1 & \cdots & a_{2^n} \\ \hline f & 2 & \cdots & 2 = 2^{2^n} \end{array}$$

n变元的布尔函数有：$2^{2^n}$ 个

例1-6 $n=7^3 X 11^2 X 13^4$，求除尽n的整数个数。

解： 能整除n的正整数可以写成如下形式：

$7^x X 11^y X 13^z$,  $0\le x \le 3$, $0\le y \le 2$, $0\le z \le 4$

故能除尽n的整数的个数为

$4 x 3 x 5 = 60$

例1-7 有a,b,c,d,e这5个字母，从中取6个构成一组字符串的个数是多少？要求同时满足：1,第1个和第6个必须是子音b,c,d。2，每一个字符都必有a,e两母音，且a,e不相邻。3，相邻两子音不允许相同。

解：符合要求的字符串有以下几种模式：

a,e位于第2和第4位置 × ○ × ○ × × $3*2*3*2*3*2$

a,e位于第2和第5位置 × ○ × × ○ × $3*2*3*2*2*3$

a,e位于第3和第5位置 × × ○ × ○ × $3*2*2*3*2*3$

字符串个数为：$2^3 * 3^3 + 2^3 * 3^3 + 2^3 * 3^3 = 648$个

例1-8有5本不同的日文书，7本不同的英文书，10本不同的中文书。

- 1)取2本不同文字的书有几种可能？

  $5\times7+5\times10+7\times10=155$；

- 2)取2本相同文字的书有几种可能？

  $C(5,2)+C(7,2)+C(10,2)=10+21+45=76$

- 3)任取两本书有几种可能？

  $155+76=231$

## 1.2 一一对应

若问题A的计数比较困难，问题B的计数比较容易，但问题A与问题B一一对应，则对A的计数可转换为对B的计数。

# 1.2 一一对应

例：求$n^2$个人站成一排和站成n排（方阵）的方案数，并比较两种方案数的大小？

解法1：比如9个人站成一排的方案数是9!，

设$a_1a_2a_3a_4a_5a_6a_7a_8a_9$是9个人的一排，

| 可构成一个方阵 | 给定一个方阵 |
|---|---|
| $a_1a_2a_3$ | $b_1b_2b_3$ |
| $a_4a_5a_6$ | $b_4b_5b_6$ |
| $a_7a_8a_9$ | $b_7b_8b_9$ |

也唯一确定一排$b_1b_2b_3b_4b_5b_6b_7b_8b_9$

因此这两种站位方式的方案数一样多，都是9!

# 1.2 一一对应

例：求$n^2$个人站成一排和站成n排（方阵）的方案数，并比较两种方案数的大小？

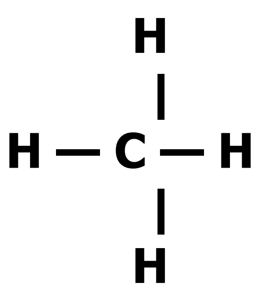
解法2：9个人站成方阵的方案数为：

$C(9,3)3!C(6,3)3!C(3,3)3!$

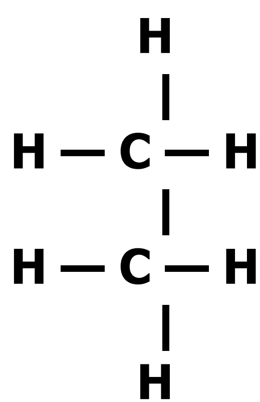
$$\frac{9!}{6!3!}\bullet 3!\bullet\frac{6!}{3!3!}\bullet 3!\bullet 3! = 9!$$

# 1.2 一一对应

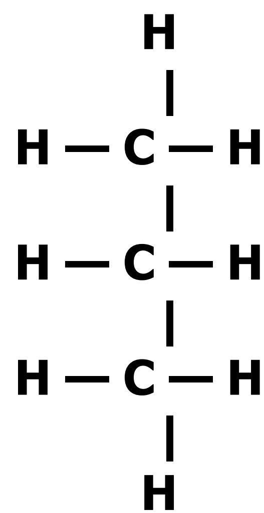
- **例1-10** $C_nH_{2n+2}$是碳氢化合物,随着n的不同有下列不同的支链：



n=1甲烷

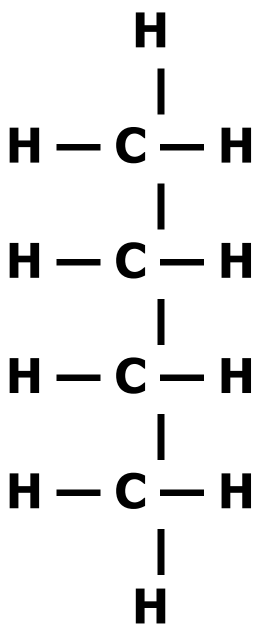


n=2 乙烷

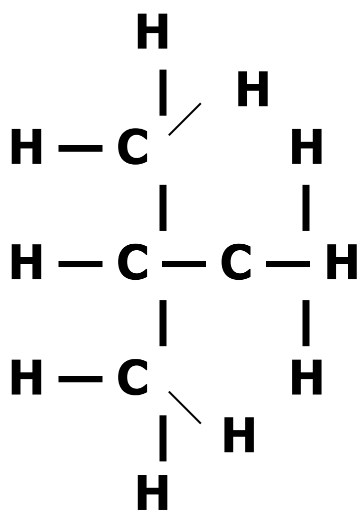


n=3 丙烷

## 1.2 一一对应



n=4 丁烷



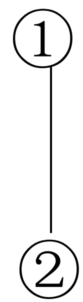
n=4异丁烷

这说明对应$C_nH_{2n+2}$的枝链是有3n+2个顶点的一棵树，其中n个顶点与之关联的边数为4；其它2n+2个顶点是叶子。对于这样结构的每一棵树，就对应有一种特定的化合物。从而可以通过研究具有上述性质的树找到不同的碳氢化合物$C_nH_{2n+2}$.

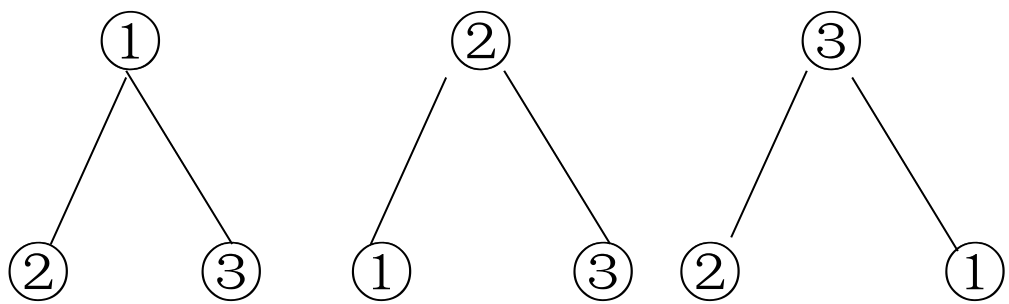
## 1.2 一一对应

例：　n个有标号1, 2, …, n的顶点的树的数目是多少？（n>=2）

n = 2时




n = 3时




n = 4时呢？

# 1.2 一一对应

**定理1-1（Caylay定理） n个有标号1, 2, ..., n的顶点的树的数目等于$n^{n-2}$。（n>=2）**

证：由一一对应关系知：

*  N个顶点树的个数 = 序列$b_1, b_2, b_3, \ldots, b_{n-2}$ {0<b<=n}的个数

* 相对应的个数为$n^{n-2}$

注：一个问题与另一个问题一一对应，则可将一个问题转化成另一个问题来处理。在处理组合技术问题时，常常通过问题的一一对应实现模型转换，以便问题的求解。

# 1.2 一一对应

如何由一棵树构造其对应的序列？

设一棵树的顶点集为A

1、从中找到编号最小的叶子结点，去掉该叶子结点$a_1$及其邻接边$(a_1, b_1)$。

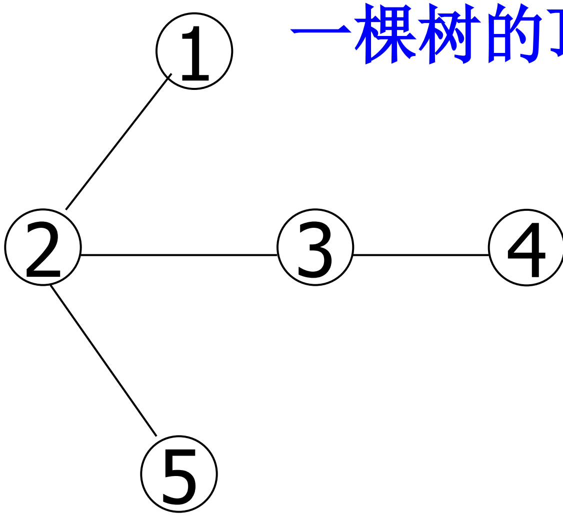
2、重复以上过程。直到剩一条边为止。

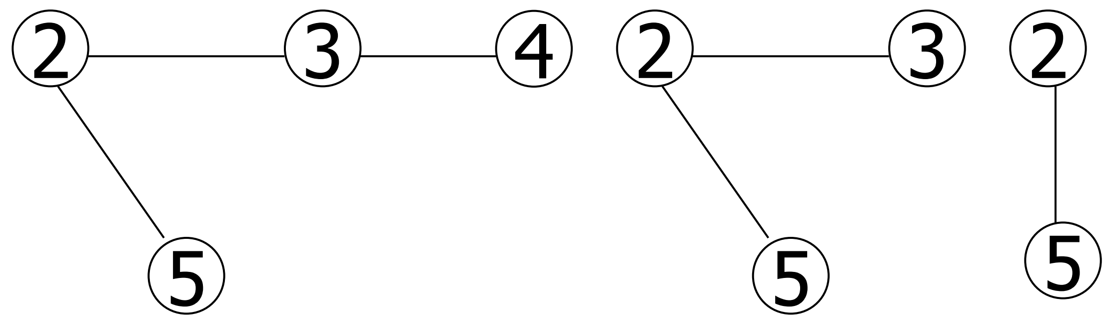
3，最终可以得到一个对应的序列

一个棵对应序列$B=b_1b_2b_3\ldots b_{n-2}$而且是唯一的

# 1.2 一一对应

一棵树的顶点集为（1, 2, 3, 4, 5）





1、找到编号最小的叶子结点是1，去掉该叶子结点及其邻接边(1, 2)

2、找到编号最小的叶子结点是4，去掉该叶子结点及其邻接边(4, 3)

3、找到编号最小的叶子结点是3，去掉该叶子结点及其邻接边(3, 2)

这棵树对应序列（2，3，2）

# 1.2 一一对应

如何由序列构造其对应的树?

任给一个序列B $\{b_1, b_2, b_3, \ldots, b_{n-2}\}$

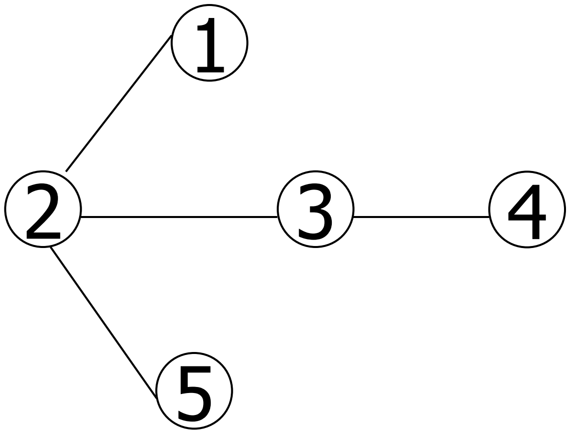
1、从A找到最小的不属于B的元素, 设为$a_1$, 与$b_1$连接, 从A中去掉$a_1$, 从B中去掉$b_1$.

2、重复以上过程只到B为空, A中剩余两个

3、连接剩余的两个顶点。

# 1.2 一一对应

如何由序列构造其对应的树?




树的顶点集合为A=（12345）

这棵树对应序列B（2，3，2）

1、从A找到最小的不属于B的元素是1,与序列B中2连接，从A中去掉1，从B中去掉2.

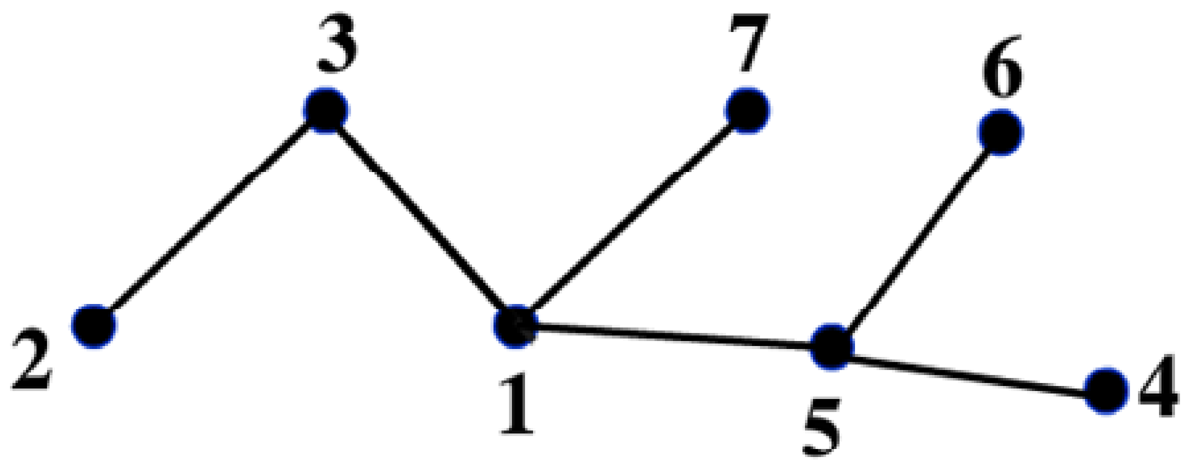
2、从A找到最小的不属于B的元素是4,与序列B中3连接，从A中去掉4，从B中去掉3

3、从A找到最小的不属于B的元素是3,与序列B中2连接，从A中去掉3，从B中去掉2

4、连接剩余的两个顶点2和5。

# 1.2 一一对应

例1-12 给定下列的树，它对应的序列是什么？




可得序列为：3,1,5,5,1。反之从序列3,1,5,5,1也可以构造上述树。

# 1.3：排列与组合

1、排列的定义：设A={$a_1$, $a_2$, …, $a_n$}是n个不同的元素的集合，任取A中r个元素按顺序排成一列，称为从A中取r个的一个排列，r满足0≤r≤n，其排列数记为P(n, r)。

| (1) | (2) | (3) | (…) | (r) |
|---|---|---|---|---|

从n个不同的球中取一个球放在第一个盒子中，

从余下的n-1个球中取一个球放在第二个盒子中，

………………………………

从余下的n-(r-1)个球中取一个放在第r个盒子中。

# 1.3：排列与组合

根据定义有：

1） $P(n,r) = 0, ( r > n)$

2） $P(n,1) = n, ( n \geq 1)$

3) $P(n,r) = n(n-1)(n-2)\ldots(n-r+1) = \frac{n!}{(n-r)!}$

当n=r时，$P(n,n) = n(n-1)(n-2)\ldots2*1 = n!$ 即是n个元素的全排列

推论1：当$n\geq r\geq 2$时，$P(n,r) = nP(n-1,r-1)$

推论2：当$n\geq r\geq 2$时，$P(n,r) = rP(n-1,r-1) +P(n-1,r)$

# 1.3：排列与组合

2、组合的定义：当从n个不同元素中取出r个而不考虑它的顺序时,称为从n中取r个的组合,其数目记为C(n,r)或$\binom{n}{r}$。

公式：从n中取r的组合数记作C(n,r)

从n中取r的排列数是P(n,r)。

组合数与排列数之间的关系:

$$C(n,r) = P(n,r)/r! = n!/[r!(n-r)!]$$

# 1.3：排列与组合

排列可以看作n个不同的元素取r个，放进r个不同的盒子的放法.

组合可以看作n个不同的元素取r个，放进r个相同的盒子的放法.

公式1：$C(n,r)=C(n,n-r)$

从n个元素选出r个元素，就有n-r个元素没有被选出，这说明每个r-组合，就对应了一个(n-r)-组合。

推论：$C(n,r)=C(n-1,r)+C(n-1,r-1)$　　Pascal公式

# 1.3：排列与组合

例1-13:由5面不同颜色的旗帜,20种不同的盆花,排成两端是旗帜,中间放3盆花的形式，问有多少种不同的方案数?

★ ♣ ♣ ♣ ★

解1：按照摆放位置逐个进行选择

$$5\times 20\times 19\times 18\times 4=136800$$

解2：5种颜色的旗帜取两个排列的排列数为

$P(5,2)=5!/3!=5*4=20$

20种不同的花取3种排列的排列数为：

$P(20,3)=20!/17!=20*19*18=6840$

根据乘法法则，共有图案数为：

$6840*20=136800$

# 1.3：排列与组合

例1-14：有男运动员7名，女运动员3名，列队进场。

1）若要求头尾2名运动员必须是男运动员，且女运动员不相邻，问有多少种排列方案？

7！X 6X5X4=5040X120=604800种方案

2）若要求女运动员排在一起，排在队的头尾两端，问有多少种排列方案？

2X3！ X7！ = 60480种方案

3）若只要求女运动员排在一起，问有多少种排列方案？

7！ X 8 X3！ = 181440种方案

# 1.3：排列与组合

例1-15：求2000到7000间的偶数中，由不同数字组成的4位数个数。

解:设所求的数的形式为abcd,

则$2 \le a \le 6$， $d \in \{0,2,4,6,8\}$

**(1) 若$a \in \{2,4,6\}$时，d有4种选择，**

$$3 \times 4 \times P(8,2) = 672$$

**(2)若$a \in \{3,5\}$时，d有5种选择，**

$$2 \times 5 \times P(8,2) = 560$$

共有**672 + 560 = 1232** 个

本题如果先考虑个位d，再考虑千位a，怎么求解？

# 1.3：排列与组合

例：求1000到9999间能有多少各位都不同的奇数?

解:设所求的数的形式为abcd,

则$1 \le a \le 9$，　d $\in \{1,3,5,7,9\}$

1）d$\in \{1,3,5,7,9\}$，　5种选择

2）a有除了d和0都可以，8种选择

3）bc是除了a和d的数字都可以，　P(8,2)种选择

共有5 * 8 * P(8,2)= 5 * 8 * 8 * 7 = 2240 个

本题如果先考虑千位a，再考虑个位d，结果如何?

一般选取约束多的位优先考虑。

# 1.4：圆周排列

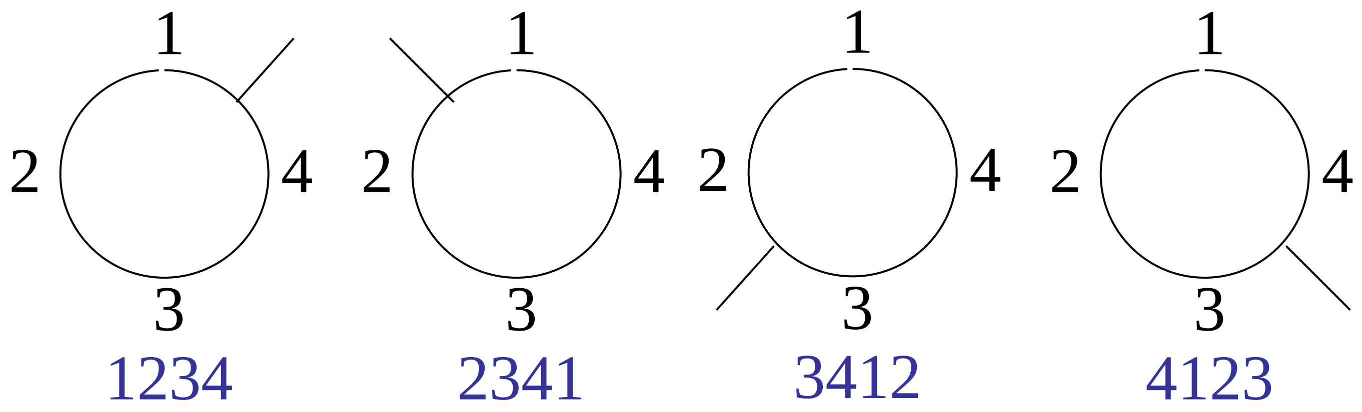
定义：在排列中，如果不横排或竖排，而是将各元素排列在一个圆周上，那么称这种排列方式为**圆周排列（循环排列）**。

规定相对位置不变算一个排列。

圆周排列与排列不同之处在于圆周排列是头尾相邻。

# 1.4：圆周排列

- 以4个元素为例做圆周排列




在排列中1234, 2341, 3412, 4123被当作四个不同的排列，而在圆排列中这些排列算一个。

# 1.4：圆周排列

将从n中取r个作圆排列的排列数记作Q(n, r)。

从n中取r个作排列，与圆排列相比，重复了r倍；

公式：$Q(n,r)=\frac{P(n,r)}{r}$

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

$$Q(n,n)=\frac{P(n,n)}{n}=\frac{n!}{n(n-n)!}=(n-1)!$$

# 1.4：圆周排列

例：5颗不同的红色珠子，3颗不同的蓝色珠子装在圆板的四周，问有多少种方案？若蓝色珠子不相邻，又有多少种排列方案？蓝色珠子在一起又如何？

解：(1) $Q(8,8)=P(8,8)/8=7!$。

(2) 蓝色珠子不在一起。

首先5颗不同的红色珠子做圆排列，共有 $Q(5,5)=4!$，3颗不同的蓝色珠子依次排在红色珠子中间，第一颗蓝珠有5种排列方案，第二颗蓝珠有4种排列方案，第三颗蓝珠有3种排列方案，所以一共是 $4!\times5\times4\times3$

(3) 蓝色珠子在一起，可以当成一个整体，加上5颗红珠做圆排列，共有 $Q(6,6)$，这3颗蓝珠自己还需要做全排列。所以一共是 $Q(6,6)3!=5!3!$。

# 1.4：圆周排列

例1-26：5对夫妇出席一宴会，围一圆桌而坐，试问有几种不同的方案？若要求每对夫妻相邻，又有多少种方案?

解： 1）座位无限制

Q(10, 10) = P(10, 10)/10 = 10!/10 = 9!种方式。

2）夫妇相邻而坐

首先可以将一对夫妇作为一个元素来看待，共有Q(5, 5)=P(5, 5)/5=24。

但夫妇可以交换坐位，5对夫妇共有$2^5$种方式。

根据乘法法则：若夫妻相邻而坐，共有

$24 \times 2^5 = 24 \times 32 = 768$种方式。

*

## 1.5 排列的生成算法

- 问题：从已知排列出发，生成新的排列
- 目的：提高效率，减少开销
- 任务：寻找好的算法
  - 序数法
  - 字典序法
  - 邻位对换法

# 1.5 排列的生成算法 – 序数法

- 对于一个自然数m，利用十进制来表示为 $(a_l, a_{l-1}, \ldots, a_0)$，其中

$m= a_l 10^l + a_{l-1} 10^{l-1} + \ldots + a_0 10^0 \quad 0 \le a_i \le 9$

  - 例如 $312 = 3*10^2+1*10^1+2*10^0$

- 对于一个自然数m，利用二进制来表示为 $(a_l, a_{l-1}, \ldots, a_0)$，其中

$m= a_l 2^l + a_{l-1} 2^{l-1} + \ldots + a_0 2^0 \quad a_i \in \{0，1\}$

  - 例如 $312 = 100111000 = 1*2^8+0*2^7+0*2^6+1*2^5+1*2^4+1*2^3+0*2^2+0*2^1+0*2^0$

## 1.5 排列的生成算法 – 序数法

- 以{(n-1)!, (n-2)!, …, 2!, 1!}为基数来表示自然数n!=n(n-1)!=(n-1+1)(n-1)!
 =(n-1)(n-1)!+(n-1)!=(n-1)(n-1)!+(n-1)(n-2)!
 =(n-1)(n-1)!+(n-2+1)(n-2)!=…
 =(n-1)(n-1)!+(n-2)(n-2)!+…+2*2!+1*1!+1
- 对于自然数n!-1的表示(n-1,n-2,…,2,1):

n!-1=(n-1)(n-1)!+(n-2)(n-2)!+…+2*2!+1*1!= $\sum_{k=1}^{n-1} kk!$

- 对于自然数1的表示(0,0,…,0,1);

  1=0(n-1)!+0(n-2)!+…+0*2!+1*1!
- 对于自然数0的表示(0,0,…,0,0);

# 1.5 排列的生成算法 – 序数法

对于任意一个整数m，$0 \le m \le n!-1$

其中$m=a_1 (n-1)+a_2 (n-2)!+\ldots+a_{n-1}*1!$　　$0 \le a_i \le n-i$

$(a_1, a_2, \ldots, a_{n-1})$序列共有$n!$个，与$0 \sim n!-1$间的整数一一对应。

$(a_1, a_2, \ldots, a_{n-1})$序列与$n!$个排列也有一一对应关系，故称之为中介数。

- 由 $(a_1, a_2, \ldots, a_{n-1}) \Rightarrow$ 整数m

$m=a_1 (n-1)! +a_2 (n-2)!+\ldots+ a_{n-2}*2! +a_{n-1}*1!$

- 由整数m $\Rightarrow$ $(a_1, a_2, \ldots, a_{n-1})$
  - m除以2，得到余数为$a_{n-1}$，商为

$$m_1=a_1 \frac{(n-1)!}{2}+a_2 \frac{(n-2)!}{2}+\ldots+ a_{n-3}\frac{3!}{2} +a_{n-2}$$

  - m除以3，得到余数为$a_{n-1}$，商为

$$m_2=a_1 \frac{(n-1)!}{2*3}+a_2 \frac{(n-2)!}{2*3}+\ldots+ a_{n-4}\frac{4!}{2*3} +a_{n-3}$$

  - …
  - $m_{n-3}$除以n-1，得到余数为$a_2$，商为$a_1$

$$m=a_1 (n-1)! + a_2 (n-2)! + \ldots + a_{n-2}*2! + a_{n-1}*1!$$

| **序号数m (0~4!)** | 中介数($a_1, a_2, a_3$)（递增进位） |
|---|---|
| 0 | (0,0,0) |
| 1 | (0,0,1) |
| 2 | (0,1,0) |
| 3 | (0,1,1) |
| 4 | (0,2,0) |
| 5 | (0,2,1) |
| 6 | (1,0,0) |
| 7 | (1,0,1) |
| … | |

- 由 $(a_1, a_2, \ldots, a_{n-1})$ 确定整数m（递减进位）

$$m=\sum_{i=1}^{n-1} a_i \frac{n!}{(n-i+1)!}$$

$$=a_1+a_2\frac{n!}{(n-1)!}+a_3\frac{n!}{(n-2)!}+\ldots+a_{n-1}\frac{n!}{2!}$$

$$=a_1+a_2 n+a_3\, n(n-1)+\cdots+\ a_{n-1}\ \, n(n-1)\cdots 3$$

$$=(\ldots((a_{n-1}\ .\ 3+a_{n-2})\ .\ 4\ +a_{n-3})\ .\ 5\ +\cdots+a_2)\ .\ n+a_1$$

- 由整数m确定$(a_1, a_2, \ldots, a_{n-1})$ （递减进位）
  - m除以n，得到余数为$a_1$，商为$m_1$

    $$m_1 = a_2 + a_3(n-1) + \ldots + a_{n-1}(n-1)\cdots 3$$
  - $m_1$除以n-1，得到余数为$a_2$，商为$m_2$

    $$m_2 = a_3 + a_4(n-2) + \ldots + a_{n-1}(n-2)(n-3)\cdots 3$$
  - …
  - $m_{n-3}$除以3，得到余数为$a_{n-2}$，商为$a_{n-1}$

    $$m_{n-3} = a_{n-2} + a_{n-1} * 3$$

$m = a_1 (n-1)! + a_2 (n-2)! + \ldots + a_{n-2} * 2! + a_{n-1} * 1!$

| 序号数m (0~4!) | 中介数($a_3, a_2, a_1$)（递减进位） |
|---|---|
| 0 | (0,0,0) |
| 1 | (0,0,1) |
| 2 | (0,0,2) |
| 3 | (0,0,3) |
| 4 | (0,1,0) |
| 5 | (0,1,1) |
| 6 | (0,1,2) |
| 7 | (0,1.3) |
| … | |

# $(a_1, a_2, \ldots, a_{n-1})$与n-排列间的一一对应关系

- 集合{1,2,…,n}有n!个排列，设$i_1 i_2 \ldots i_n$是其中的一个排列
  - 若$k<l$，但$i_k>i_l$，则$(i_k, i_l)$是一对逆序
  - 排列31524有4个逆序(3, 1), (3, 2), (5, 2), (5, 4)
  - 排列12…n是唯一没有逆序的排列
- 对于排列$i_1 i_2 \ldots i_n$，令$a_j$表示与j相关的逆序数，即排列中先于j但大于j的整数的个数
- 排列$i_1 i_2 \ldots i_n$与逆序列$(a_1, a_2, \ldots, a_{n-1})$一一对应
  - 例：31524的逆序列为（1, 2, 0, 1）

# 排列$i_1 i_2 \ldots i_n$确定逆序列$(a_1, a_2, \ldots, a_{n-1})$

- 排列$i_1 i_2 \ldots i_n$推出逆序列$(a_1, a_2, \ldots, a_{n-1})$
  - 从排列$i_1 i_2 \ldots i_n$中找到1的位置，1的左边比1大的整数的个数即为$a_1$
  - 从排列$i_1 i_2 \ldots i_n$中找到2的位置，2的左边比2大的整数的个数即为$a_2$
  - …
  - 从排列$i_1 i_2 \ldots i_n$中找到n-1的位置，n-1的左边比n-1大的整数的个数即为$a_{n-1}$
- 例：31524的逆序列为（1, 2, 0, 1）

# 逆序列$(a_1, a_2, \ldots, a_{n-1})$确定排列$i_1i_2\ldots i_n$

- 由逆序列$(a_1, a_2, \ldots, a_{n-1})$确定排列$i_1i_2\ldots i_n$
  - 确定n个空位；
  - 确定1的位置：从左向右数$a_1$个空格，然后下一个空格填上1；
  - 确定2的位置：从左向右数$a_2$个空格，然后下一个空格填上2；
  - …
  - 确定n-1的位置：从左向右数$a_{n-1}$个空格，然后下一个空格填上n-1；
  - 最后一个空格填上n
- 由逆序列（1, 2, 0, 1）确定31524

# 用序数法（递增进位）生成n!个全排列

- 例n = 4, n! = 24个排列

| 序数<br>**m [0~4!-1]** | 逆序列（递增进位）<br>$(a_1, a_2, a_3)$ | 排列<br>$i_1i_2i_3i_4$ |
|---|---|---|
| 0 | (0,0,0) | 1234 |
| 1 | (0,0,1) | 1243 |
| 2 | (0,1,0) | 1324 |
| 3 | (0,1,1) | 1423 |
| 4 | (0,2,0) | 1342 |
| 5 | (0,2,1) | 1432 |
| … | … | … |

# 用序数法（递减进位）生成n!个全排列

- 例n = 4, n! = 24个排列

| 序数<br>**m [0~4!-1]** | 逆序列（递减进位）<br>（$a_3, a_2, a_1$） | 排列<br>$i_1i_2i_3i_4$ |
|---|---|---|
| 0 | (0,0,0) | 1234 |
| 1 | (0,0,1) | 2134 |
| 2 | (0,0,2) | 2314 |
| 3 | (0,0,3) | 2341 |
| 4 | (0,1,0) | 1324 |
| 5 | (0,1,1) | 3124 |
| … | … | … |

# 1.5 排列的生成算法 – 字典序法

- 规定两个全排列的大小：从左到右逐个比较对应的字符的大小
- 例：字符集{1,2,3}，较小的数字较先，这样暗自点序生成的全排列是：123,132,213,231,312,321.

# 字典序法生成排列的算法

- 第一个排列是1,2,3,…,n;
- 最后一个排列是n,n-1,…,2,1;
- 已知一个排列$i_1 i_2 \ldots i_k \ldots i_n$，求下一个排列
  - 从右向左找到第一个右邻比自己大的数$i_k$。

$i_k < i_{k+1}$，但$i_{k+1} > i_{k+2}$ ,$i_{k+2} > i_{k+3}$ ,…, $i_{n-1} > i_n$

  - 在$i_{k+1} i_{k+2} \ldots i_n$中找到比$i_k$大的数，在这些数中找到最右的数$i_j$
  - $i_k$和$i_j$交换位置，形成一个新序列$j_1 j_2 \ldots j_n$
  - 将新序列中$j_{k+1} j_{k+2} \ldots j_n$顺序逆转。

# 字典序法举例

- 求839647521的下一个排列。
  - 从右向左一个一个比较，1<2不满足，2<5不满足，5<7不满足， 7不大于4满足了（$i_k$=4）
  - 在4右边（7521）找比4大的数（7和5），多于一个取最右边的那个，本例是5（$i_j$=5）
  - 数字4和数字5交换位置得一新的序列839657421
  - 新序列的后缀部分（7421）进行反转，本例7421反转成1247
  - 所求的下一个排列就是839651247

# 1.5 排列的生成算法 – 算法分析

- 在递增进位法中，中介数的最低位是逢2进1，进位频繁，这是一个缺点。
- 在递减进位法中中介数进位不频繁，求下一个排列在不进位的情况下容易。
- 这就给了启发，能不能设计一种算法
  - 下一个排列总是上一个排列某相邻两位对换得到。
  - 递减进位法的换位是单向的，从右向左
  - 而将要结束的邻位对换法的换位是双向的。

# 1.5 排列的生成算法 – 邻位互换法

- 一个排列的逆序个数是奇还是偶，称排列为期排列和偶排列
- 算法描述如下：
  - 对[1 n-1]的每一个偶排列，n从右到左插入n个空档（包括两端），生成1 to n的n个排列。
  - 对[1 n-1]的每一个奇排列，n从左到右插入n个空档（包括两端），生成1 to n的n个排列。