组合数学入门

$$C_n^m = C_{n-1}^{m-1} + C_{n-1}^m$$

$$mC_n^m = nC_{n-1}^{m-1}$$

$$C_n^0 + C_n^1 + C_n^2 + \ldots\ldots + C_n^n = 2^n$$

$$1C_n^1 + 2C_n^2 + 3C_n^3 + \ldots\ldots + nC_n^n = n2^{n-1}$$

$$1^2C_n^1 + 2^2C_n^2 + 3^2C_n^3 + \ldots\ldots + n^2C_n^n = n(n+1)2^{n-2}$$

$$\frac{C_n^1}{1} - \frac{C_n^2}{2} + \frac{C_n^3}{3} + \ldots\ldots + (-1)^{n-1}\frac{C_n^n}{n} = 1 + \frac{1}{2} + \frac{1}{3} + \ldots\ldots + \frac{1}{n}$$

$$(C_n^0)^2 + (C_n^1)^2 + (C_n^2)^2 + \ldots\ldots + (C_n^n)^2 = C_{2n}^n$$

- $C(k,k) + C(k+1, k) + .. C(n, k) = C(n+1, k+1)$

- $C(n,0) + C(n, 1) + .. + C(n, k) = C(n+1, k+1)$

N个球放入M个盒子

- 1球相同，盒相同，可以为空
- 2球相同，盒相同，不能为空
- 3球不同，盒不同，可以为空
- 4球不同，盒不同，不能为空
- 5球不同，盒相同，可以为空
- 6球不同，盒相同，不能为空
- 7球相同，盒不同，可以为空
- 8球相同，盒不同，不能为空

xjoi

3269	放球同不同不空	1 - 2
3270	放球II同不同空	1 - 2
3367	放球3不同不同空	1 - 3
3368	放球4不同同不空	2 - 9
3369	放球5不同同空	2 - 9
3370	放球6不同不同不空	2 - 9
3371	放球7同同空	2 - 9
3372	放球8同同不空	2 - 9

球相同，盒相同，可以为空

- 等价于整数拆分
- dp[i][j]表示将i拆分成小于等于j个整数的方案数
- 1:至少有一个盒子为空
- 2:所有盒子都不为空
- dp[i][j] = dp[i][j - 1] + dp[i - j][j]
- http://www.matrix67.com/blog/archives/6348

球相同，盒相同，不能为空

- 利用之前的结果

- dp[i][j] - dp[i][j-1]

球不同，盒不同，可以为空

- 每个球都有m种不同的选择
- 答案为n^m

球不同盒相同，不能为空

- dp[i][j]:将i个球放入j个盒子里，不为空
- 当前这个球要么放到之前的盒子里，要么新开一个盒子
- dp[i][j] = j * dp[i - 1][j] + dp[i - 1][j - 1]

球不同，盒不同，不能为空

- 先假设盒子都是相同的，就转换成了前一个问题
- 最终答案：dp[n][m] * m!

球不同盒相同，可以为空

- 可以为空就是枚举每一种情况求和
- 最终答案：sigma(dp[n][i])（i=1->m)

球相同，盒不同，不能为空

- 1 2 1 和 1 1 2算不同的分法，用隔板法解决
- c[n - 1][m - 1]

球相同，盒不同，可以为空

- 给每个盒子放上一个球，问题就等价于将n+m个相同的球放到m个不同的盒子里，不为空
- c[m + n - 1][n - 1]

错位排列

- 每个数都不在自己位置上的排列
- 考虑递推，i放到1～i-1的某个位置j，假设j这个数放到i位置上，那么就是剩下的i-2个数进行错排，假如j这个数不放在i位置上，那么j放哪里都可以，就是剩下的i-1个数进行错排
- f[i] = (i - 1) * (f[i - 2] + f[i - 1])
-

圆排列

- 固定一个点，剩下的点一共有(n-1)!种排列

重复排列

- k种不同的球，每种个数分别为ai，n = sigma(ai)
- 排列总数为n!/(a1! * a2! * a3!...ak!)

重复组合

- n种不一样的球，每种球的个数是无限的，从中选出k个出来的方案数为
- 问题等价于选k次球，每次都可以选择n种球的方案数，选球不分先后
- 等价于k个相同的球放到n个不同的盒子里，盒子可以为空
- $C(n + k - 1, k) = C(n + k - 1, n - 1)$

斯大林数

- 第一类斯大林数：将n个不同物体排成k个非空环的方案数
- Dp[n][k] = (n - 1) * dp[n - 1][k] + dp[n - 1][k - 1]
- 要么自己当老大成一个环，要么稳一点加入之前的环
- 第二类斯大林数：将n个不同物体分成k个非空集合的方案数
- 等价于放球问题6，n个不同的球放到m个相同的盒子里，不能为空
- dp[n][k] = k * dp[n - 1][k] + dp[n - 1][k - 1]

另一种形式

众所周知，C_n^k表示从n件物品中选出k件的方案数，P_n^k表示从n件物品中选出k个排列的方案数
显然有

$$C_n^k = \frac{P_n^k}{k!}$$

根据定义我们可以得到

$$P_n^k = n * (n-1) * \ldots * (n-k+1)$$

展开后可得

$$P_n^k = S(k,k) * n^k - S(k,k-1) * n^{k-1} \ldots$$

所以有

$$P_n^k = \sum_{i=0}^{k} (-1)^{k-i} * s(k,i) * n^i$$

这个才是第一类斯特林数的定义。
所以第一类斯特林数S就是排列数公式的展开式的系数，也是如上所述的那个东西。

斯特林数降幂技巧

- 求0到n的k次方和

- 两类斯特林数都可以求

一个小结论

：

$$\sum_{i=0}^{k} C_n^i = C_{n+1}^{k+1}$$

证明：因为$C_{n+1}^{k+1} = C_n^k + C_n^{k+1}$，等式两边可以同时把$C_n^k$消掉，然后就又得到了一个相同形式的式子。所以通过归纳法便可以证明这个结论。

求自然数幂和

我们设

$$S_k(n) = \sum_{i=0}^{n} i^k$$

根据第一类斯特林数的定义，有

$$C_n^k = \frac{P_n^k}{k!} = \frac{\sum_{i=0}^{k}(-1)^{k-i} * s(k,i) * n^i}{k!}$$

把式子中的n^k提出来，得

$$n^k = C_n^k * k! - \sum_{i=0}^{k-1}(-1)^{k-i} * s(k,i) * n^i$$

$$S_k(n) = \sum_{i=0}^{n} i^k$$

$$= \sum_{i=0}^{n} (C_i^k * k! - \sum_{j=0}^{k-1} (-1)^{k-j} * s(k,j) * i^j)$$

$$= k! * \sum_{i=0}^{n} C_i^k - \sum_{i=0}^{n} \sum_{j=0}^{k-1} (-1)^{k-j} * s(k,j) * i^j$$

$$= k! * C_{n+1}^{k+1} - \sum_{j=0}^{k-1} (-1)^{k-j} * s(k,j) * S_j(n)$$

$$= \frac{P_{n+1}^{k+1}}{k+1} - \sum_{j=0}^{k-1} (-1)^{k-j} * s(k,j) * S_j(n)$$

于是就可以愉快地$O(k^2)$递推$S_k(n)$啦。边界条件$S_1(n) = \frac{n*(n+1)}{2}$

因为$P_{n+1}^{k+1} = (n+1) * n * (n-1) * \ldots * (n-k+1)$，而连续k+1个数中必然有一个是k+1的倍数，所以在用这种方法求自然数幂和时，由于不存在除法，所以并不用考虑模数是什么。

第二类斯特林数的N的k次表示

$$n^k = \sum_{i=1}^{k} \binom{n}{i} \times S(k,i) \times i!$$

一般求自然数幂和都会用到拉格朗日插值法，但仅当存在逆元的时候能用，给出一种用第二类斯特林数求自然数幂和的方法，时间复杂度是$O(k^2)$而不是$O(k\ log\ k)$的。

第二类斯特林数

众所周知，有

$$i^k=\sum_{j=0}^{k}\left\{ {k \atop j} \right\}i^{\underline{j}}$$

于是乎我们有

$$F(n)=\sum_{i=1}^{n}i^k=\sum_{i=1}^{n}\sum_{j=0}^{k}\left\{ {k \atop j} \right\}i^{\underline{j}}$$

$$F(n)=\sum_{i=1}^{n}\sum_{j=0}^{k}\left\{ {k \atop j} \right\}j!\binom{i}{j}$$

$$F(n)=\sum_{j=0}^{k}\left\{ {k \atop j} \right\}j!\sum_{i=j}^{n}\binom{i}{j}$$

我们又有

$$\sum_{i=j}^{n}\binom{i}{j}=\binom{n+1}{j+1}$$

证明可以考虑组合意义，把n+1个数排成一排，考虑第一个选的数的位置，剩下的数中再选j个，就能得到上面的式子。

当然也可以用$\binom{i}{j}=\binom{i-1}{j-1}+\binom{i-1}{j}$此式子依次展开上式即可。

接着式子就变成了

$$F(n)=\sum_{j=0}^{k}\begin{Bmatrix}k\\j\end{Bmatrix}j!\binom{n+1}{j+1}$$

$$F(n)=\sum_{j=0}^{k}\begin{Bmatrix}k\\j\end{Bmatrix}\frac{(n+1)^{\underline{j+1}}}{j+1}$$

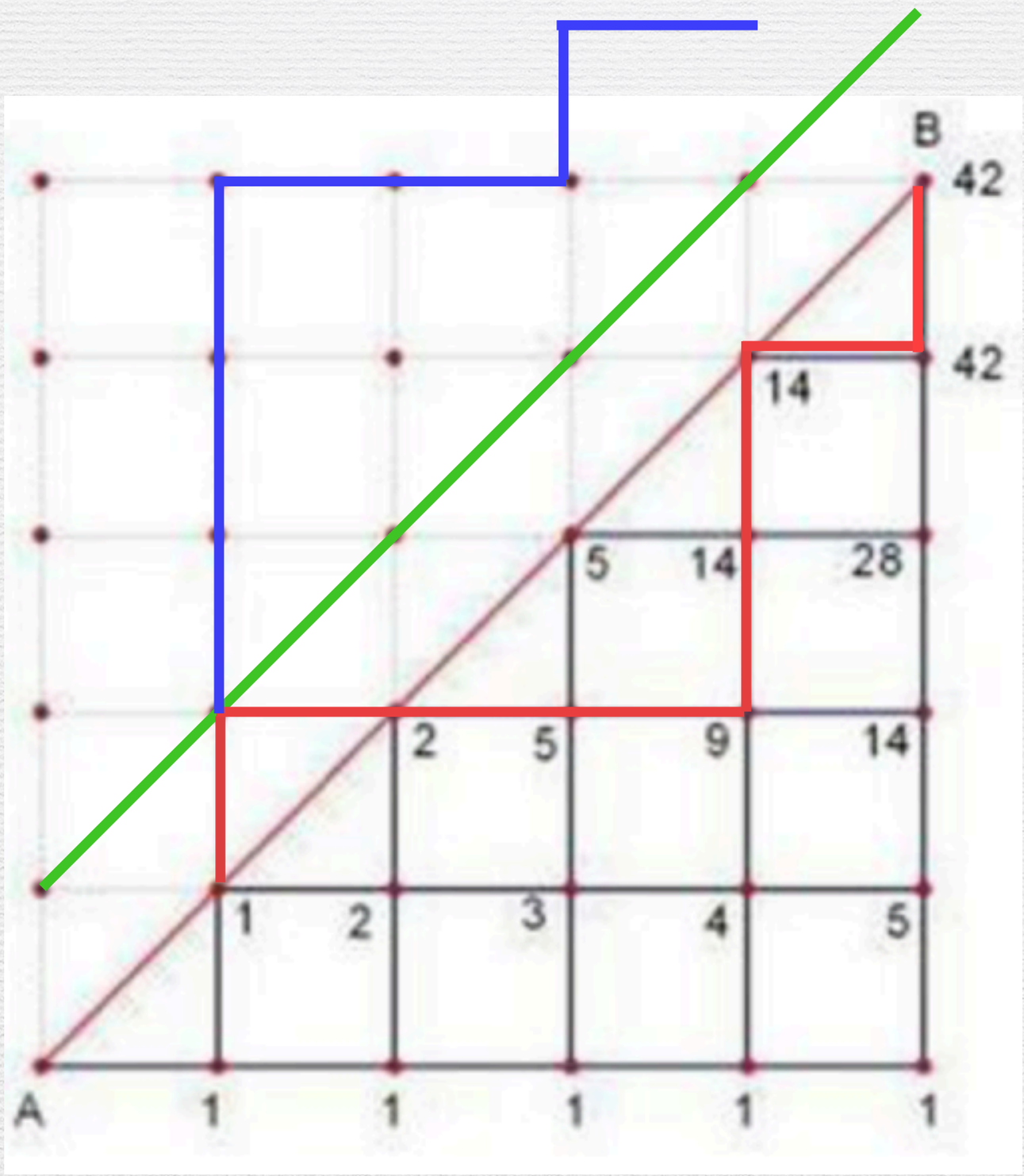
很明显$\frac{(n+1)^{\underline{j+1}}}{j+1}$一定是整数，因此就不需要逆元了，第二类斯特林数用$O(k^2)$计算出来，时间复杂度$O(k^2)$。

推荐题目

- 这类题目有待挖掘
- hdu 4625
- 推荐题解
- https://blog.csdn.net/mys_c_k/article/details/79942486

卡特兰数

- 出栈次序问题，进栈顺序为1～n，有多少种出栈顺序
- 合法的括号匹配方案数
- n+1个数连乘，不同的乘法顺序，这个可与凸多边形划分对应
- 将一个凸多边形划分成若干个三角形，划分线不交叉的方案数
- 从网格的左下角走到右上角，不穿过对角线的方案数
- n个点的不同形状二叉树的数量，要走右儿子，一定之前走过了左儿子，往左走可以看成左括号，往右走看成右括号
- 前几项为1 1 2 5 14 42 132 429 1430


B
42
42
14
5
14
28
2
5
9
14
1
2
3
4
5
A
1
1
1
1
1

- 任何一条非法路径一定存在一个中间点在对角线上方，也就是会碰到对角线上方的斜线
- 从这个点开始，将之后部分的折线按照对角线上方的斜线做对称，最终达到的点为$(n-1,n+1)$
- 我们会发现任何一种非法方案都可以对称成$(0,0)$到$(n-1,n+1)$的的一条路径，他们之间存在一一映射关系
- 所以总方案数是$C(2n, n) - C(2n, n + 1)$
- 化简如下

$$
\begin{aligned}
Catalan_n &= C_{2n}^{n} - C_{2n}^{n+1} \\
&= \frac{(2n)!}{n! * n!} - \frac{(2n)!}{(n+1)! * (n-1)!} \\
&= \frac{1}{n+1}(\frac{(2n)! * (n+1)}{n! * n!} - \frac{(2n)!}{n! * (n-1)!}) \\
&= \frac{1}{n+1}(\frac{(2n)! * (n+1)}{n! * n!} - \frac{(2n)! * n}{n! * n!}) \\
&= \frac{1}{n+1} * \frac{(2n)! * (n+1) - (2n)! * n}{n! * n!} \\
&= \frac{1}{n+1} * \frac{(2n)!}{n! * n!} \\
&= \frac{1}{n+1} C_{2n}^{n}
\end{aligned}
$$

卡特兰数的性质

卡特兰数有一些优美的性质，如

通项公式一　$C_n = \dfrac{1}{n+1}\mathrm{C}_{2n}^{n} = \mathrm{C}_{2n}^{n} - \mathrm{C}_{2n}^{n-1}$ ；

通项公式二　$C_n = \dfrac{1}{n+1}\sum\limits_{i=0}^{n}\left(\mathrm{C}_{n}^{i}\right)^2$ ；

递推公式一　$C_{n+1} = \dfrac{2(2n+1)}{n+2}C_n$ ，且 $C_0 = 1$ ；

递推公式二　$C_{n+1} = \sum\limits_{i=0}^{n} C_i C_{n-i}$ ，且 $C_0 = 1$ ；

增长速度　$\Delta C_n \sim \dfrac{4^n}{n^{\frac{3}{2}}\sqrt{\pi}}$.

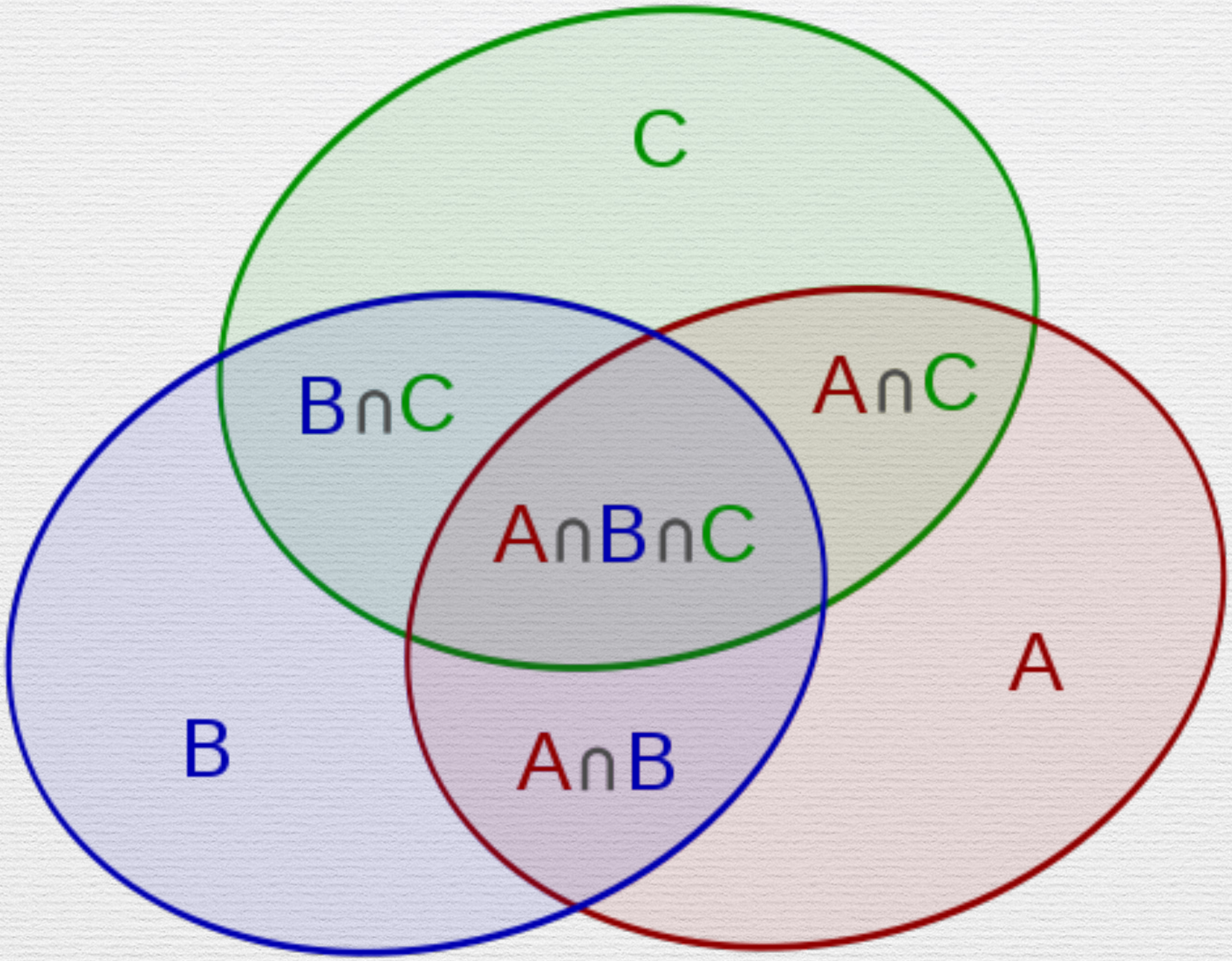
五边形数定理

- 可以快速求整数拆分

- 参考资料

- https://blog.csdn.net/acdreamers/article/details/12259815

容斥原理inclusion and exclusion


C
BnC
AnC
AnBnC
A
B
AnB

In its general form, the principle of inclusion–exclusion states that for finite sets A_1, ..., A_n, one has the identity

$$\left|\bigcup_{i=1}^{n} A_i\right| = \sum_{i=1}^{n} |A_i| - \sum_{1 \le i < j \le n} |A_i \cap A_j| + \cdots$$
$$\cdots + \sum_{1 \le i < j < k \le n} |A_i \cap A_j \cap A_k| - \cdots + (-1)^{n-1} |A_1 \cap \cdots \cap A_n| .$$

This can be compactly written as

$$\left|\bigcup_{i=1}^{n} A_i\right| = \sum_{k=1}^{n} (-1)^{k+1} \left(\sum_{1 \le i_1 < \cdots < i_k \le n} |A_{i_1} \cap \cdots \cap A_{i_k}| \right)$$

or

$$\left|\bigcup_{i=1}^{n} A_i\right| = \sum_{\emptyset \neq J \subseteq \{1,2,\ldots,n\}} (-1)^{|J|-1} \left|\bigcap_{j \in J} A_j\right|.$$

- 说的直白一点就是，所有一个图形的交减去所有两两组合的交，加上所有三三组合的交。。。
- 为什么是对的?
- 我们要证明什么?
- 韦恩图中被覆盖次数相同的区域在容斥的计算过程中恰好会被加到一次

- 所以我们需要计算容斥系数
- 假设被覆盖k次的面积之和为sk
- 在所有一个图形的组合里面，这块面积会被加到k次，因为它存在于k个集合中
- 在所有两两组合的交里面，这块面积会被减去C(k, 2)次，因为这块面积会出现在所有k个集合的2组合中
- 在三三组合的交里面，会被加上C(k, 3)次
- 所以总的累加次数为T=C(k,1)-C(k,2) +.. +(-1)^(k-1) * C(k,k)
- (1-x)^k = C(k,0) - C(k,1)*x + C(k, 2) * x^2 …+(-1)^k * C(k, k) * x^k
- 令x为1，0 = 1 - T
- 所以T = 1

容斥的应用

- 求x1+x2+..+x6=20 (0<=xi<=8)的整数解的数量
- 由0到9的数字组成排列，要求第一个数大于1，最后一个数小于8的排列数量
- 长度为n的由数字0，1，2组成的数列，要求每种数字至少出现一次，有多少种
- 给定整数n,r，求区间[1,r]中与n互质的数的个数
- 给出n个数，从中选出4个数，使得gcd为1，一共有多少选法

这里列出了一些可以用容斥原理解决的习题。

· UVA #10325 **"The Lottery"** [难度：简单]

· UVA #11806 **"Cheerleaders"** [难度：简单]

· TopCoder SRM 477 **"CarelessSecretary"** [难度：简单]

· TopCoder TCHS 16 **"Divisibility"** [难度：简单]

· SPOJ #6285 NGM2 **"Another Game With Numbers"** [难度：简单]

· TopCoder SRM 382 **"CharmingTicketsEasy"** [难度：中等]

· TopCoder SRM 390 **"SetOfPatterns"** [难度：中等]

· TopCoder SRM 176 **"Deranged"** [难度：中等]

· TopCoder SRM 457 **"TheHexagonsDivOne"** [难度：中等]

· SPOJ #4191 MSKYCODE **"Sky Code"** [难度：中等]

· SPOJ #4168 SQFREE **"Square-free integers"** [难度：中等]

· CodeChef **"Count Relations"** [难度：中等]

母函数

- 待续