整数分拆 [编辑]

维基百科，自由的百科全书

本条目需要**精通或熟悉本主题的编者**参与及协助编辑。 *（2013年12月31日）*
请邀请适合的人士改善这篇条目 。 更多的细节与详情请参见条目讨论页 。 您可以关注您所擅长领域的专家关注分类。

一个正整数可以写成一些正整数的和 。 在数论上，跟这些和式有关的问题称为**整数分拆** 、 **整数剖分** 、 **整数分割** 、 **分割数**或**切割数** 。 其中最常见的问题就是给定正整数 n ，求不同数组 $(a_1, a_2, ..., a_k)$ 的数目，符合下面的条件：

1. $a_1 + a_2 + ... + a_k = n$ （ k 的大小不定）
2. $a_1 \geq a_2 \geq ... \geq a_k > 0$
3. 其他附加条件（例如限定「k是偶数」，或「 a_i 不是1就是2」等）

分割函数 p(*n*)是求符合以上第一、二个条件的数组数目。

目录 [隐藏]

拆分数量数列 [编辑]

4可以用5种方法写成和式：4, 3+1, 2+2, 2+1+1, 1+1+1+1。 因此 $p(4) = 5$。

定义 $p(0) = 1$ ，若n为负数则 $p(n) = 0$。

此函数应用于对称多项式及对称群的表示理论等。

分割函数p(n)，n从0开始：

1, 1, 2, 3, 5, 7, 11, 15, 22, 30, 42, 56, 77......(OEIS:A000041)

程式实现 [编辑]

```
#include <iostream>
using namespace std ;

int main () {
   const int len = 121 ;
   int num [ len + 1 ] = { 1 };

   for ( int i = 1 ; i <= len ; ++ i )
      for ( int j = i ; j <= len ; ++ j )
         num [ j ] += num [ j - i ];

   for ( int i = 0 ; i <= len ; i ++ )
      cout << i << ' ' << num [ i ] << endl ;
   return 0 ;
}
```

Ferrers图示与恒等式 [编辑]

每种分割方法都可用**Ferrers图示**表示。

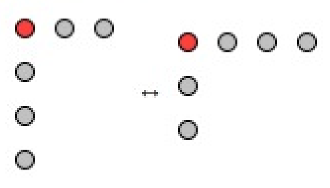
Ferrers图示是将第1行放 a_1 个方格，第2行放 a_2 个方格……第 k 行放 a_k 个方格，来表示整数分割的其中一个方法。

借助Ferrers图示，可以推导出许多恒等式 ：

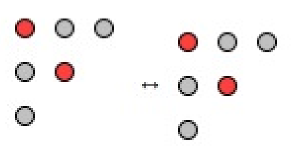
- 给定正整数k和n，n表达成不多于k个正整数之和的方法数目，等于将n分割成任意个不大于k的正整数之和的方法数目。

证明：将表示前者其中一个数组的Ferrers图示沿对角线反射，便得到后者的一个数组。 即两者一一对应，因此其数目相同。

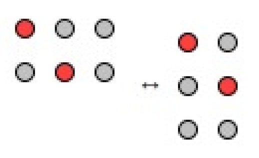
例如k=3，n=6：



6 = 1+1+4 = 1+1+1+3



6 = 1+2+3 = 1+2+3



6 = 2+2+2 = 3+3

此外，

$$6 = 1 + 5 = 1 + 1 + 1 + 1 + 2$$
$$6 = 2 + 4 = 2 + 2 + 1 + 1$$
$$6 = 3 + 3 = 2 + 2 + 2$$
$$6 = 6 = 1 + 1 + 1 + 1 + 1 + 1$$

- 上述恒等式的值亦等于将 $n + k$ 表达成刚好 k 个正整数之和的方法的数目。
- 给定正整数 n 。 将 n 表达成两两相异正整数之和的方法的数目，等于将 n 表达成奇数之和的方法的数目。

例如 $n = 8$ ：

1. 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1
2. 7 + 1
3. 3 + 3 + 1 + 1
4. 5 + 3
5. 5 + 1 + 1 + 1
6. 3 + 1 + 1 + 1 + 1 + 1

1. 8
2. 7 + 1
3. 6 + 2
4. 5 + 3
5. 5 + 2 + 1
6. 4 + 3 + 1

- 将 n 表达成 p 个1和 q 个2之和，这些方法的数目是第 n 个斐波那契数 。
- 将 n 表达成多于1的正整数之和的方法数目是p(*n*) - p(*n* -1)。

生成函数 [编辑]

$p(n)$ 的生成函数是

$$\sum_{n=0}^{\infty} p(n)x^n = \prod_{k=1}^{\infty} \left(\frac{1}{1 - x^k} \right)$$

当|x|<1，右边可写成：

$$(1 + x + x^2 + x^3 + ...)(1 + x^2 + x^4 + x^6 + ...)(1 + x^3 + x^6 + ...)...$$

$p(n)$ 生成函数的倒数为欧拉函数 ，利用五边形数定理可得到以下的展开式：

$$\prod_{k=1}^{\infty}(1 - x^k) = \sum_{i=-\infty}^{\infty}(-1)^i x^{i(3i-1)/2}.$$

将 $p(n)$ 生成函数配合五边形数定理 ，可以得到以下的递归关系式

$$p(n) = \sum_i (-1)^{i-1} p(n - q_i)$$

其中 q_i 是第 i 个广义五边形数 。

与杨氏矩阵的关系 [编辑]

一个杨氏矩阵与一个整数分拆一一对应，也就是说整数分拆的个数等于相应的杨氏矩阵的个数。 如图表示一个10=5+4+1的分拆。 利用杨氏矩阵来表示的分拆更具有直观性，和可处理性，下面是几个例子。

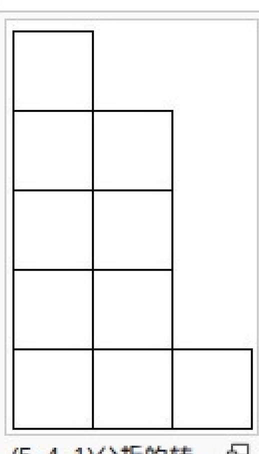
一个(5, 4, 1)分拆表示的杨表

分拆的转置 [编辑]

整数分拆（10=5+4+1）对应的杨氏矩阵沿x=y轴翻转得到新的杨氏矩阵。 它对应分拆为10=3+2+2+2+1。

附加要求的分拆 [编辑]

考虑带有附加条件的分拆。



(5, 4, 1)分拆的转置(3, 2, 2,2,1)

差分拆 [编辑]

考虑满足下面条件分拆

1. $a_1 + a_2 + ... + a_k = n$ （ k 的大小不定）
2. $a_1 > a_2 ... > a_k$

及分拆的每个数都不相等。

生成函数是

$$\sum_{n=0}^{\infty} p(n)x^n = \prod_{k=1}^{\infty} \left(1 + x^k\right)$$

奇分拆 [编辑]

考虑满足下面条件分拆

1. $a_1 + a_2 + ... + a_k = n$ （ k 的大小不定）
2. $a_1 \geq a_2 ... \geq a_k$
3. 要求 $a_i (i = 1, 2, ..., k)$ 为奇数

生成函数是

$$\sum_{n=0}^{\infty} p(n)x^n = \prod_{k=1}^{\infty} \left(\frac{1}{1 - x^{2k-1}} \right).$$

引理 [编辑]

差分拆的个数与奇分拆的个数是一样多的。

可以通过杨表证明。

Rademacher级数 [编辑]

渐近式：

$$p(n) \sim \frac{\exp\left(\pi\sqrt{2n/3}\right)}{4n\sqrt{3}} \text{ as } n \to \infty.$$

这式子是1918年哈代和拉马努金 ，以及1920年JV Uspensky独立发现的。

1937年， Hans Rademacher得出一个更佳的结果：

$$p(n) = \frac{1}{\pi\sqrt{2}} \sum_{k=1}^{\infty} A_k(n)\,\sqrt{k}\,\frac{d}{dn} \left(\frac{\sinh\left(\frac{\pi}{k}\sqrt{\frac{2}{3}\left(n - \frac{1}{24}\right)}\right)}{\sqrt{n - \frac{1}{24}}} \right)$$

其中

$$A_k(n) = \sum_{0 \leq m < k\,;\,(m,k)=1} \exp\left(\pi i s(m,k) - 2\pi i n m / k\right).$$。

$(m, n) = 1$ 表示 m, n 互质时才计算那项。 $s(m, k)$ 表示戴德金和 。 这条公式的证明用上了福特圆 、 法里数列 、 模群 和戴德金η函数 。

Elder定理 [编辑]

在将 n 表示成正整数之和的所有和式之中，任意正整数 r 作为和项出现在这些式子内的次数，跟每条和式中出现 r 次或以上的正整数数目，相同。

当 $r = 1$ 时，此定理又称为Stanley定理。 [1] [2]

以 $n = 5$ 为例：

- 5
- 4+1
- 3+2
- 3+1+1
- 2+2+1
- 2+1+1+1
- 1+1+1+1+1

1. 1的总出现次数：0+1+0+2+1+3+5=12；在每条和式出现1次或以上的数的数目：1+2+2+2+2+2+1 =12
2. 2的总出现次数：0+0+1+0+2+1+0=4；在每条和式出现2次或以上的数的数目：0+0+0+1+1+1+1 =4。

$p_k(n)$ [编辑]

当限定将 n 表示成刚好 k 个正整数之和时，可以表示为 $p_k(n)$。 显然， $p(n) = \sum_{k=1}^{n} p_k(n)$。

- 对于 $n > 1$ ， $p_n(n) = p_1(n) = 1$
- $p_2(n) = \lfloor \frac{n}{2} \rfloor$ (OEIS:A004526)
- $p_3(n)$ = 最接近 $\frac{n^2}{12}$ 的正整数。 (OEIS:A069905)
- $p_{n-1}(n) = 1$
- $p_k(n) = p_{k-1}(n - 1) + p_k(n - k)$