母函数(Generating function)详解 — TankyWoo

母函数(Generating function)详解

—— Tanky Woo

在数学中，某个序列的**母函数(Generating function，又称生成函数)**是一种形式幂级数，其每一项的系数可以提供关于这个序列的信息。使用母函数解决问题的方法称为**母函数方法**。

母函数可分为很多种，包括**普通母函数**、**指数母函数**、**L级数**、**贝尔级数**和**狄利克雷级数**。对每个序列都可以写出以上每个类型的一个母函数。构造母函数的目的一般是为了解决某个特定的问题，因此选用何种母函数视乎序列本身的特性和问题的类型。

这里先给出两句话，不懂的可以等看完这篇文章再回过头来看：

> 1. “把组合问题的加法法则和幂级数的乘幂对应起来”
> 2. “母函数的思想很简单 — 就是把离散数列和幂级数一 一对应起来，把离散数列间的相互结合关系对应成为幂级数间的运算关系，最后由幂级数形式来确定离散数列的构造. ”

我们首先来看下这个多项式乘法：

$$(1+a_1x)(1+a_2x)\cdots(1+a_nx)$$
$$=1+(a_1+a_2+\cdots+a_n)x+(a_1a_2+a_1a_3+\cdots+a_{n-1}a_n)x^2$$
$$+\cdots+a_1a_2\cdots a_nx^n$$

母函数图(1)

由此可以看出:

> 1.x的系数是a1,a2,…an 的单个组合的全体。
> 2. x^2的系数是a1,a2,…a2的两个组合的全体。
> ………
> n. x^n的系数是a1,a2,….an的n个组合的全体（只有1个）。

进一步得到：

$$(1+x)^n=1+C(n,1)x+C(n,2)x^2+\cdots+C(n,n)x^n$$

母函数图(2)

母函数的定义

对于序列a0，a1，a2，…构造一函数：

$$G(x)=a_0+a_1x+a_2x^2+\cdots,$$

称函数G(x)是序列a0，a1，a2，...的母函数。

这里先给出2个例子，等会再结合题目分析：

第一种：
有1克、2克、3克、4克的砝码各**一枚**，能称出哪几种重量？每种重量各有几种可能方案？
考虑用母函数来解决这个问题：
我们假设x表示砝码，x的**指数**表示砝码的重量，这样：

> 1个1克的砝码可以用函数1+1*x^1表示，
> 1个2克的砝码可以用函数1+1*x^2表示，
> 1个3克的砝码可以用函数1+1*x^3表示，
> 1个4克的砝码可以用函数1+1*x^4表示，

上面这四个式子懂吗？
我们拿1+x^2来说，前面已经说过，x表示砝码，x的指数表示砝码的重量！初始状态时，这里就是一个质量为2的砝码。
那么前面的**1**表示什么？按照上面的理解，1其实应该写为：1*x^0,即1代表重量为2的砝码数量为0个。
所以这里1+1*x^2 = 1*x^0 + 1*x^2，即表示2克的砝码有两种状态，不取或取，不取则为1*x^0，取则为1*x^2

不知道大家理解没，我们这里结合前面那句话：
"**把组合问题的加法法则和幂级数的乘幂对应起来**"

接着讨论上面的1+x^2，这里x前面的系数有什么意义？
这里的系数表示**状态数**(**方案数**)
1+x^2，也就是1*x^0 + 1*x^2，也就是上面说的不取2克砝码，此时有1种状态；或者取2克砝码，此时也有1种状态。(分析！)

所以，前面说的那句话的意义大家可以理解了吧？
几种砝码的组合可以称重的情况，可以用以上几个函数的乘积表示：
(1+x)(1+x^2)(1+x^3)(1+x^4)
=(1+x+x^2+x^4)(1+x^3+^4+x^7)
=1 + x + x^2 + 2*x^3 + 2*x^4 + 2*x^5 + 2*x^6 + 2*x^7 + x^8 + x^9 + x^10
从上面的函数知道：**可称出从1克到10克，系数便是方案数。（！！！经典！！！）**
例如右端有2^x^5 项，即称出5克的方案有2种：5=3+2=4+1；同样，6=1+2+3=4+2；10=1+2+3+4。
故称出6克的方案数有2种，称出10克的方案数有1种 。

接着上面，接下来是第二种情况：

第二种：
求用1分、2分、3分的邮票贴出不同数值的方案数：
大家把这种情况和第一种比较有何区别？第一种每种是**一个**，而这里每种是**无限**的。

$$G(x) = (1 + x + x^2 + \cdots)(1 + x^2 + x^4 + \cdots)$$
$$\cdot (1 + x^3 + x^6 + \cdots)$$

母函数图(4)

以展开后的x^4为例，其系数为4，即4拆分成1、2、3之和的拆分方案数为4；
即 ：4=1+1+1+1=1+1+2=1+3=2+2

这里再引出两个概念"整数拆分"和"拆分数"：
所谓**整数拆分**即把整数分解成若干整数的和（相当于把n个无区别的球放到n个无标志的盒子，盒子允许空，也允许放多于一个球）。
整数拆分成若干整数的和，办法不一，不同拆分法的总数叫做**拆分数**。

现在以上面的第二种情况每种种类个数无限为例，给出**模板**：

```
#include <iostream>
using namespace std;
// Author: Tanky Woo
// www.wutianqi.com
const int _max = 10001;
// c1是保存各项质量砝码可以组合的数目
// c2是中间量，保存没一次的情况
int c1[_max], c2[_max];
int main()
{       //int n,i,j,k;
        int nNum;   //
        int i, j, k;

        while(cin >> nNum)
        {
                for(i=0; i<=nNum; ++i)   // ---- ①
                {
                        c1[i] = 1;
                        c2[i] = 0;
                }
                for(i=2; i<=nNum; ++i)   // ----- ②
                {

                        for(j=0; j<=nNum; ++j)   // ----- ③
                                for(k=0; k+j<=nNum; k+=i)  // ---- ④
                                {
                                        c2[j+k] += c1[j];
                                }
                        for(j=0; j<=nNum; ++j)     // ---- ⑤
                        {
                                c1[j] = c2[j];
                                c2[j] = 0;
                        }
                }
                cout << c1[nNum] << endl;
        }
        return 0;
}
```

我们来解释下上面标志的各个地方：(************！！！**重点**！！！************)

> ① 、首先对c1初始化，由第一个表达式(1+x+x^2+..x^n)初始化，把质量从0到n的所有砝码都初始化为1.
> ② 、 i从2到n遍历，这里i就是指第i个表达式，上面给出的第二种母函数关系式里，每一个括号括起来的就是一个表达式。
> ③、j 从0到n遍历，这里j就是(前面i個表达式累乘的表达式)里第j个变量，(这里感谢一下seagg朋友给我指出的错误，大家可以看下留言处的讨论)。如(1+x)(1+x^2)(1+x^3)，j先指示的是1和x的系数，i=2执行完之后变为
>
> （1+x+x^2+x^3）(1+x^3)，这时候j应该指示的是合并后的第一个括号的四个变量的系数。
> ④ 、 k表示的是第j个指数，所以k每次增i（因为第i个表达式的增量是i）。
> ⑤ 、把c2的值赋给c1,而把c2初始化为0，因为c2每次是从一个表达式中开始的。

咱们赶快趁热打铁，来几道题目：
（相应题目解析均在相应的代码里分析）
1. 题目：http://acm.hdu.edu.cn/showproblem.php?pid=1028
代码：http://www.wutianqi.com/?p=587
这题大家看看简单不？把上面的模板理解了，这题就是小Case!
看看这题：
2. 题目：http://acm.hdu.edu.cn/showproblem.php?pid=1398
代码：http://www.wutianqi.com/?p=590
要说和前一题的区别，就只需要改2个地方。 在i遍历表达式时（可以参考我的资料—《母函数详解》），把i<=nNum改成了i*i<=nNum,其次在k遍历指数时把k+=i变成了k+=i*i; Ok,说来说去还是套模板~~~
3. 题目：http://acm.hdu.edu.cn/showproblem.php?pid=1085
代码：http://www.wutianqi.com/?p=592
这题终于变化了一点，但是万变不离其中。
大家好好分析下，结合代码就会懂了。
4. 题目：http://acm.hdu.edu.cn/showproblem.php?pid=1171
代码：http://www.wutianqi.com/?p=594
还有一些题目，大家有时间自己做做：
HDOJ：1709，1028、1709、1085、1171、1398、2069、2152

附：
1.在维基百科里讲到了普通母函數、指數母函數、L級數、貝爾級數和狄利克雷級數：http://zh.wikipedia.org/zh-tw/%E6%AF%8D%E5%87%BD%E6%95%B0
2．Matrix67大牛那有篇文章：什么是生成函数：
http://www.matrix67.com/blog/archives/120
3.大家可以看看杭电的ACM课件的母函数那篇，我这里的图片以及一些内容都引至那。

如果大家有问题或者资料里的内容有错误，可以留言给出，博客:http://www.wutianqi.com/

老版下载地址：

母函数（Generating function）详解（点击下载）
(仅作保留所用，里面有错误，建议看我的最新版本，关注本博客：http://www.wutianqi.com/?p=539)

最后更新：2012.02.24
这次更新变动较大，显示结构更为工整清晰，也修正了一些有歧义的地方。如果大家转载过我以前的版本，可以重新换成这个版本。

Tanky Woo @ 2010-08-02 113 Comments
Category: Algorithms, 我的原创
Tags: 原创, 母函数, 详解