

# 全国青少年信息学奥林匹克联赛NOIP2017 上海市模拟赛解题报告

by Tommyr7 朱玮昊

2017年10月22日

T1:

算法一:

我们直接暴力枚举出 $6*6$ 的每一种情况, 然后判断 $a[i]>b[j]$ ,  $a[i]<b[j]$ 的情况数量, 分别讨论两种情况哪个更多, 然后直接输出即可。

需要注意一下相等的情况, `else`的那一句需要小心一下。

时间复杂度 $O(1)$ , 期望得分100分。

T2:

算法一:

我们可以手动特判一下只有三个糖果的时候每天的购买情况, 然后在3天内的购买情况也分别进行讨论, 细节比较多需要小心仔细。

时间复杂度 $O(2^{(n^2)})$ , 期望得分20分。

算法二:

由于一共只会买 $m$ 个糖果, 所以我们可以暴力枚举每一个糖果是在哪一天买的, 然后直接来计算答案。

时间复杂度 $O(n^n)$ , 期望得分40分。

算法三:

我们考虑动态规划 $dp$ ,  $f[i][j]$ 表示前 $i$ 天一共购买了 $j$ 个糖果的最优值。在转移第 $i$ 天的情况是, 我们发现剩下的一维正好是一个0-1背包问题, 于是直接套用经典的0-1背包 $dp$ 即可。

时间复杂度 $O(n^4)$ , 期望得分70分。

算法四:

考虑优化算法三, 每一天的购买情况其实我们是可以预处理的。我们将每一天的糖果进行价格上的从小到大排序。于是我们就可以统计出第 $i$ 天我们买 $j$ 个糖果所需要的最少价格, 然后直接套用 $dp$ 就可以了。

时间复杂度 $O(n^3)$ , 期望得分100分。

T3:

算法一:

我们考虑暴力单点修改以及子树查询, 我们需要统计出每一个数2,3,5,7,11,13的质因子个数, 最后通过快速幂方式算出个数, 约数个数直接将它

们相加后连乘即可。

时间复杂度 $O(qn)$ ，期望得分60分。

算法二：

我们考虑到这个题需要支持树上的单点修改以及子树询问，很容易想到直接硬上树链剖分。（如果你不会树链剖分，直接看算法三）但是由于注意到，需要对每个素因子进行一次维护，所以算法自带6倍常数，所以过不了最后的数据。

时间复杂度 $O(q \log^2 n)$ ，期望得分80分。

算法三：

我们考虑这棵树的DFS序列，于是一棵子树在DFS序列中就成了连续一段区间。于是我们需要对每个素因子维护一个支持单点修改以及支持区间求和的数据结构。直接上线段树或者通过树状数组求前缀和的方式都是可以的。

时间复杂度 $O(q \log n)$ ，期望得分100分。

## 骰子的游戏(sides.pas/c/cpp)

### 【解题报告】

我们直接暴力枚举出 $6*6$ 的每一种情况，然后判断 $a[i]>b[j]$ ,  $a[i]=b[j]$ ,  $a[i]<b[j]$ 的情况数量，分别

## 购物(shopping.pas/c/cpp)

### 【题目描述】

在上海市青少年活动中心的附近，有一家糖果专卖店。（别找了，不存在的）

这家糖果店将会在每天出售一些糖果，它每天都会生产出 $m$ 个糖果，第 $i$ 天的第 $j$ 个糖果价格为 $C[i][j]$ 元。

现在的你想要在接下来的 $n$ 天去糖果店进行选购，你每天可以买多个糖果，也可以选择不买糖果，但是最多买 $m$ 个。（因为最多只生产 $m$ 个）买来糖果以后，你可以选择吃掉糖果或者留着之后再吃。糖果不会过期，你需要保证这 $n$ 天中每天你都能吃到至少一个糖果。

这家店的老板看你经常去光顾这家店，感到非常生气。（因为他不能好好睡觉了）于是他会额外的要求你支付点钱。具体来说，你在某一天购买了 $k$ 个糖果，那么你在这一天需要额外支付 $k^2$ 的费用。

那么问题来了，你最少需要多少钱才能达成自己的目的呢？

### 【输入格式】

输入文件名为shopping.in。

第一行两个正整数 $n$ 和 $m$ ，分别表示天数以及糖果店每天生产的糖果数量。

接下来 $n$ 行（第2行到第 $n+1$ 行），每行 $m$ 个正整数，第 $x+1$ 行的第 $y$ 个正整数表示第 $x$ 天的第 $y$ 个糖果的费用。

### 【输出格式】

输出文件名为shopping.out。

输出只有一个正整数，表示你需要支付的最小费用。

### 【样例输入1】

```
3 2
1 1
100 100
10000 10000
```

### 【样例输出1】

```
107
```

### 【样例输入2】

```
5 5
1 2 3 4 5
2 3 4 5 1
3 4 5 1 2
4 5 1 2 3
5 1 2 3 4
```

### 【样例输出2】

```
10
```

**【数据规模与约定】**

对于前20%的数据， $1 \leq n, m \leq 3$ 。

对于前40%的数据， $1 \leq n, m \leq 8$ 。

对于前50%的数据， $1 \leq n, m \leq 10$ 。

对于另10%（不与前50%相交）的数据， $1 \leq n, m \leq 75$ ， $m$ 个糖果的价格每天都不变。

对于前70%的数据， $1 \leq n, m \leq 75$ 。

对于100%的数据， $1 \leq n, m \leq 300$ ，所有输入的数均 $\leq 10^6$ 。

## 随机树(rand.pas/c/cpp)

### 【题目描述】

平日里写hash的时候，总有某些选手由于脸黑而导致惨遭卡模数，然后一些恶意卡模数的出题人也因此身败名裂。为了防止被卡，我们用一种高级的随机方式来代替原来的线性随机生成，也就是所谓的随机树！

现在有一棵编号为 $0 \sim n-1$ 的有根树，其中0是树的根。每个节点初始有一个值 $T_i$ 。现在要求支持一下两种操作：

1. 给出两个正整数 $u$ 和 $x$ ，我们将 $T_u$ 的值乘以 $x$ ，我们将这种操作称为SEED操作。

2. 给出一个正整数 $i$ ，询问 $S_i$ 以及它一共有多少个正约数。其中 $S_i$ 表示以 $i$ 为根的子树所有点的权值的乘积，我们将这种操作称为RAND操作。

容易发现，这样得到的答案还是很随机的。（其实不是）

你需要回答每一次的询问，由于一个数的约数个数可能非常多，这个数也可以非常大，你只需要把答案对 $1e9+7$ 取模就可以了。

### 【输入格式】

输入文件名为rand.in。

第一行一个正整数 $n$ ，表示节点个数。

接下来 $n-1$ 行，每行两个正整数 $u$ 和 $v$ ，表示 $u$ 是 $v$ 的父节点。

接下来一行 $n$ 个正整数，分别表示每个节点的初始权值 $T_i$ 。

接下来一行一个正整数 $q$ ，表示操作的个数。

接下来 $q$ 行，每行是以下两种情况之一：

1. SEED  $u$   $x$

表示将 $u$ 节点的权值乘以 $x$ 。

2. RAND  $i$

表示询问 $S_i$ 以及它一共有多少个正约数。

### 【输出格式】

输出文件名为rand.out。

每一行两个整数，对应一个RAND操作，你需要输出所求的权值以及它的正约数个数，答案对于 $1e9+7$ 取模即可。

### 【样例输入】

```
8
0 1
0 2
1 3
2 4
2 5
3 6
3 7
7 3 10 8 12 14 40 15
3
```

RAND 1  
SEED 1 13  
RAND 1

**【样例输出】**

14400 63  
187200 126

**【数据规模与约定】**

对于20%的数据， $1 \leq n, q \leq 10$ 。

对于40%的数据， $1 \leq n, q \leq 100$ 。

对于60%的数据， $1 \leq n, q \leq 2000$ 。

对于80%的数据， $1 \leq n, q \leq 50000$ 。

对于100%的数据， $1 \leq n, q \leq 100000$ 。

另外请注意，所有读入的数一定满足 $1 \leq x \leq 10^9$ 。

同时，数据保证在任意时刻，每个点的权值不可能拥有超过13的素因子，也就是说，每个数的素因子最多只有2，3，5，7，11，13这六种可能。