# POI XIX

## StageI

### Festival

题意简述：

有 n 个数，给出两种制约条件：a) X 比 Y 小 1； b) X 不比 Y 大。求不同的数的个数的最大值。

解法：

首先用 a 类条件建图 ,可以发现同一个联通块里面的数之间的关系是确定的。设最小的数是 x，最大的数是 x+a，那么我们可以用(x,a)来表示一个联通块。

那么对于某个 b 类条件(i,j)，假设 i 所属的联通块是(x,a)且 i=x+a’，j 属于(y,b)且 j=y+b’；我们可以得出 x 和 y 之间的一个约束关系：y-x>=a’-b’。

我们再把 a 类条件中的联通块缩成点，用 b 类条件的边来连接它们。易证：不属于同一强连通分量的点互不影响。于是我们只需考虑一个**强连通分量**。

另外，对于(x,a)，我们可以把它看做一条左端点在 x，长度为 a+1 的线段；原问题转化为求最大覆盖长度。那么对于一个强连通分量，求出一个合法解后，答案就是 **max(xj+aj)-min(xi)+1** !! 为什么呢？如果“线段”之间有空隙呢？

假设存在空隙，那么当前会有两堆（或者更多）分开的“线段”；这样的话，对于不属于同一堆的两条线段(xi,ai)和(xj,aj)，就有 xi+ai<xj 即 xj-xi>=ai+1；注意到约束关系的产生公式：**y-x>=a’-b’**，a’-b’<=a! 因为属于同一强连通分量，所以 ai+1 应该小于等于 ai。矛盾。所以中间没有空隙！

求 max(xj+aj)-min(xi)+1 的话，从每个点开始求最短路即可。

（这题想了我 3h+。。。好弱）

## Letters

题意简述：

给出两个小写字母组成的字符串 a 和 b，每次你可以交换 a 的相邻的字母，最少交换多少次使得 a 与 b 相同。

题解：

从左到右扫描，如果相同的话不换；不相同的话找最近的换；BIT。

## Distance

题意简述：

定义 D(a,b)为使 a 变为 b 的最少操作次数。每次操作可以使 a 乘以或者除上

一个质数。给出 n 个数 A[] 。对于每个数 i，求出最小的 j（不等于 i），使得 D(A[i],A[j])最小。

题解：

注意到 D(a,b)=F(a/d)+F(b/d)，d=gcd(a,b)，F(x)为质因数个数之和。那么我们设 G(y)为通过除法到达 y 的操作数最少、次少的数。对于每个数 x 我们枚举它的约数 y，用 F(x/y)去更新 G(y)即可。然后求 A[i]的最小 j 时同理。

## Rendezvous

题意简述：

给出一堆仙人掌（每个点出度为 1），每次询问(a,b)，要求你返回(x,y)， 在 max(x,y)最小的情况下 min(x,y)最小；且满足：有两个人分别在点 a 和点 b，他们只能沿着有向边的方向走 x 步和 y 步，最后到达同一地方。

题解：

预处理后计算。（这也算题解么。。。）

## Well

题意简述：

给出 A[1..n]和 m，要求 B[1..n]满足 B[i]<=A[i]且 sigma(A[i]-B[i])<=m，而且至少有一个 B[j]为 0；满足这些条件后 B[1..n]应该尽可能地“平滑”。平滑度=max|B[i]-B[i-1]|。平滑度越小，越平滑。

题解：

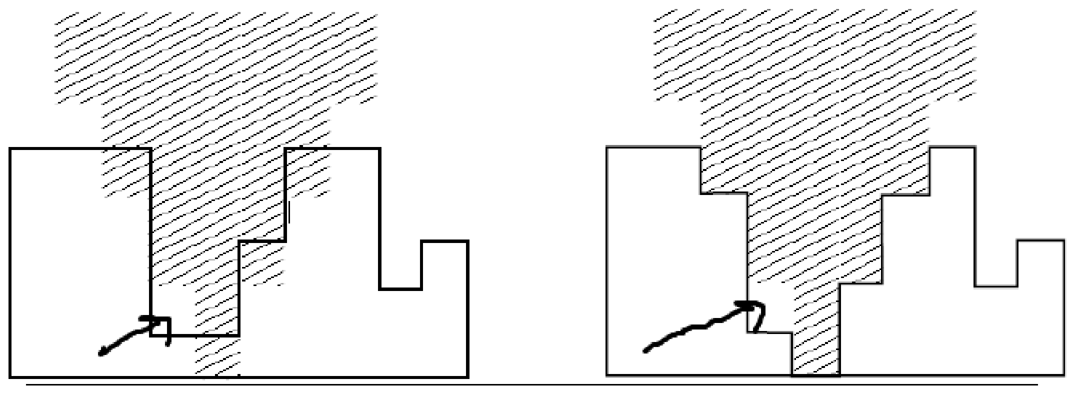
首先二份平滑度 z。我们先不考虑 B[j]=0；如何使得 A 足够平滑？以下代码足矣：

```
Rep(i,1,n-1) if (a[i+1]>a[i]+z)  a[i+1]=a[i]+z;
Repd(i,n,2) if (a[i-1]>a[i]+z) a[i-1]=a[i]+z;
```

而且求出来的 A' 是代价最小的！

这里就不证明了。。。有兴趣可以看波兰文题解。。（这也叫兴趣？！）（反正我是用 google 翻译成英文然后看懂了。。。）合法性显然，代价最小可以用归纳法来证。

那么如果让其中一个 B[j]变为 0 呢？看图！



假如我们让 k 变为 0，那么我们只需要减去以 k 为中心的“倒金字塔”里面

的数就好了！

可是，就像箭头所指的一样，万一缺了一块怎么办？不合法啊？

只要我们之前已经把平滑度减少到 z 以下，那么必定不会出现这种情况！这个也可以证明。。请看。。。。

可是这样子还是 O(n^2)的啊？当然可以优化到 O(n)的，我们只要维护右轮廓和左轮廓的变化就好了。。具体请自己思考^_^

# StageII

## Tour_de_Byteotia：

题意简述：

给出一个 n 个顶点的无向图，要求删去最少的边，使得编号小于等于 k 的点都不在环上。

题解：

有一个优美的解法 我们把编号小于等于 k 的点和与之相连的边视为断开的，然后对新图求边双连通分量。然后把边双连通分量进行缩点，把断开的边恢复——那么原图中所有的非树边都必须删去了。

这题 MLE 把我弄死了 T_T

## Vouchers

题意简述：

水题。。。

题解：

免了吧。。。

## Cloakroom

题意简述：

给出 n 种物品(Ai,Bi,Ci)，和 q 个询问(Li,Ri,Ki)。对于每个询问，选出若干种物品使得: Aj<=Li, Ri<=Bj ,sigma(Cj)=Ki。N<=1000, K<=10^6 ,q<=10^6。

题解：

想了很久都想不出好方法，结果。。。。。

O(n^2*K/32)就可以过了。。。

因为有两个限制条件，我们用离线去掉第一个条件，然后维护 n 个背包就好了。。

注意 unsigned int 的使用：位运算只能同一类型之间进行，否则会挂；特判 0 和~0U 才可以卡过。。。

## A Horrible Poem

题意简述：

给出一个字符串 S，每次询问(i，j)，要求 S(i..j)这个子串的最短循环节。

题解：

注意到对于串 S[1..n]如果有长度为 L 的循环节，那么 S[L+1..n]=S[1,n-L]

对于询问(i，j)，我们设 L=j-i+1。分解 L 显然可以得到 O(n^1.5)的算法，然而这是不够的。

先设 L=p1^a1*p2^a2*…*pm^am。

再设 bi=max(j, j<=ai 并且 L/(pi^j)是循环节)。

那么答案就是 p1^b1*p2^b2*…*pm^bm。

证明不难；只需证若 x’ * y 和 y’ * x 是合法循环节，满足 y’ | y 且 x’ | x，x 与 y 互质，则 x’ * y’ 亦是循环节即可。证明如下：

因为 x’ y 是合法循环节，那么我们只考虑 S 开头的长度为 x ‘y 的一段；那么这一段的循环节长度与 y’ x 有关，可以列出：

(i+(y’ x)*k)mod(x’ y)=i .那么：

(y’ x)*k = (x’ y)*j， 可以得出：

k=j*(x’ /x)*(y/y’ )， 那么：

k 的最小值就是(y/y’ )，也就是说:

新的循环节长度是(x’ y)/(y/y’ )=x’ y’ ! !

证了这个，结合归纳法，算法的正确性就确定了。

## Fibonacci_Representation

题意简述：

询问最少用多少个 Fibonacci 数加减得出一个数 n。

题解：

这尼玛太水了吧。。。。。。。

假设 Fi<=n<Fi+1,那么 G(n)=1+min(G(n-Fi), G((Fi+1)-n))

直接记忆化爆搜。。。

# StageIII

## Squarks

题意简述：

给出 n 个数 A[1..n]的两两之和 B[1..(n-1)*n/2]，求出所有可能的 A[1..n]

题解：

这题这么水我做的时候居然只有 13AC。。。

显然确定最小数后，可以在 O(n^2logn 内)确定其他数。于是只关心最小数的取值。

假设 A[]和 B[]已排序，那么显然 B[1]=A[1]+A[2]，B[2]=A[1]+A[3]。但是 B[3]不一定是 A[2]+A[3]（否则就能确定 A[1]）；观察到 B[3]...B[n+1]都有可能，于是枚举即可。

## Bidding

题意简述：

交互式博弈题。游戏规则如下：

有一个栈和一个壶，和很多很多球，并给定 n。一开始壶内有一个球，栈内没有。每次操作，每个人可以选择：壶内的球加倍；壶内的球加两倍（乘以 3）；把壶内所有球移入栈内，壶内再出现一个球。如果轮到某个玩家操作时，壶与栈的球的总数大于等于 n，那么他就输了。

题解：

我勒个去这题交了我 20 多次啊有木有！！！！！！！！！

直接记忆化搜索就可以了。。。

我的 WA+RE 轨迹：

把自己的操作上报后忘记在自己这里更新当前的信息 ——>

因为递归爆栈，RE ——>

BFS,TLE——>

用 map, RE——>

改成二维数组 dp——>AC

而且因为是交互式题。。。难以调试。。。

（好吧不该用景明老师的颜艺的。。。村正好虐心 T_T）

## Salaries

题意简述：

在一棵有根树上，每个点都有自己的不同权值，在[1,n]之间；且父亲的权值比儿子的大。现在给出了某些点的权值，其他的点权值未知。问还可以确定哪些点的权值。题目保证，如果给出某个点的权值，则其父亲的权值也会被给出。

题解：

一开始没看到红字……于是悲剧了。。。

看到后简单很多，但是也有不少细节。

我们从小到大地扫描权值 c：如果 c 不存在，那么加入队列 Q 中；如果 x 这个点的权值是 c，那么我们开始判断：

如果当前应该被填权值的点的个数小于队列的大小（即可用权值的数量），那么把 x 加入队列 A 中；

否则队列 Q 中的所有权值都应该填入队列 A 中的点上。我们从最大的可用权值 c’ 开始扫描，如果 A 队列中只有一个节点，或者 c’ 比除了 A 的队尾的点外（也就是 x）的所有点的权值要大，那么进入下一步判断（否则跳出）：如果 x 有且仅有一个未确定儿子 y，那么 y 必定是 C’ ，递归 y；否则也跳出。

最后要把 Q 和 A 清空。（总之靠自己 YY）

## Leveling Ground：

题意简述：

给出 n 个数 A[]，和 a、b。每次操作可以给一个区间都+a，-a，+b，-b。求最小的操作次数使得 A[]都变为 0。

题解：

首先求 d=gcd(a,b)判断无解神马的。然后 a,b,A[]都除以 d。

设 B[i]=A[i]-A[i-1](1<=i<=n+1)。那么区间操作给[l,r]加上 x 相当于 B[l]+=x, B[r+1]-=x， 再设 B[i]+a*Xi+b*Yi=0， 那么目标就是最小化：Sigma(|Xi|)+Sigma(|Yi|)， 并且满足 Sigma(Xi)=Sigma(Yi)=0（因为每加一次就减一次，每减一次就加一次）。

我们先不管约束条件，对于 i 求出|Xi|+|Yi|最小的可行解。不失一般性，假设此时的 Sigma(Xi)>=0，那么因为 Sigma(Xi)*a+Sigma(Yi)*b=0，所以 Sigma(Yi)<=0 并且 Sigma(Xi)是 b 的倍数！如果我们调整 i 的二元组(Xi,Yi)，每次可以给 Xi 加上（减去）b，Yi 减去（加上）a，而且能够使 Sigma(Xi)加上（减去）b。而这次操作对答案的影响则容易计算。

于是我们每次选出一个对答案影响最小的 i 进行调整，直至 Sigma(Xi)=0 为止。可以使用堆来维护。

对于 a*Xi+b*Yi=1，怎么求最小的|Xi|+|Yi|？

一句话，要么|Xi|<b，要么|Yi|<a。

## Minimalist Security

题意简述：

给出一幅有向图，每个点 x 有权值 p(x)。对于边(x,y)，有 p(x)+p(y)>=b(x,y)。现在要求每个点一个新的权值 z(x)，满足 p(x)>=z(x)，且对于边(x,y)有 z(x)+z(y)=b(x,y)。求 sigma(p(x)-z(x))的最大值和最小值。

题解：

对于一个联通块，确定某一点为基点，那么其他点和 sigma(p(x)-z(x))都能用其表示出来了，也能得到基点的取值范围。

## Warehouse Store

题意简述：

有一间 xx 仓库，第 i 天早上到货 Ai；同一天中午会有订单，要求买货 Bi（不可延迟）。求最多能接受的订单数。

题解：

按 Bi 从小到大处理，能接受就接受，用了线段树。貌似用堆也可以。

# Prefixuffix

题意简述：

给出字符串 S[1..n]，求最大的 L，满足：

1） S[1..L]与 S[n-L+1,n]循环同构

2） 这两个串不能重叠

题解：

膜拜 oimaster+acmonster 神犇……

所谓循环同构，其实就是两次去掉相同的前后缀， 比如：

Ababbabababbaab

(L 最大为 6，红字为第一次去掉的前后缀，蓝字为第二次去掉的前后缀)

设红字的长度为 L，那么再设 F(L)表示蓝字的最长长度，易证:

F(L)>=F(L-1)-2

但这样子是没办法递推的。。

我们变形:

F(L-1)<=F(L)+2 就可以了。。。