Nim博弈

题意

有n堆石子，第i堆有A(i)颗石子。两人依次从中拿取，规定每次只能从一堆中取若干根，可将一堆全取走，但不可不取，最后取完者为负，求必胜的方法。

思路

这道题是nim博弈的模板题

题目1：有**n**堆石子，第**i**堆有**A(i)**颗石子。两人依次从中拿取，规定每次只能从一堆中取若干根，可将一堆全取走，但不可不取，最后取完者为胜，求必胜的方法。

令C=A(1) xor A(2) xor A(3) xor ... xor A(n)，若C>0，则记为利己态，用S表示，若C=0，则记为利他态，用T表示。

【定理**1**】对于一个**S**态，一定能从一堆石子中取出若干个，使其成为**T**态。

【证明】既然是S态，则此时C>0，我们要使得C变为0。

设C转化为二进制后，最高位的1是第p位。那么一定存在一个A(t)的二进制最高位的1是第p位。（显然，不然C的第p位不可能是1）

然后，把第t堆石子的个数变为x=A(t) xor C。因为A(t)和C的二进制最高位的1是同一位。那么异或之后这一位就变成了0。那么x一定小于A(t)。

此时的C'=A(1) xor A(2) xor ... xor A(t) xor C xor A(t+1) xor ... xor A(n)。把C带进去，得到

C'=A(1) xor A(2) xor ... xor A(n) xor A(1) xor A(2) xor ... xor A(n)。显然C'=0。

所以，只要在第t堆石子中取出A(t)-x颗石子，就把S态变为了T态。

【定理**2**】对于一个**T**态，从任意一堆取任意个石子出来，都会变为**S**态。

【证明】用反证法。设此时第i堆的石子数变成了A(i')。此时C=0。如果C'>0，那么命题就成立了。

假设C'=0。则C'=A(1) xor A(2) xor ... xor A(i') xor... xor A(n)=0。

因为C=0。所以C xor C'=0。则A(1) xor A(2) xor ... xor A(i) xor... xor A(n) xor A(1) xor A(2) xor ... xor A(i') xor... xor A(n)=0。

A(i) xor A(i')=0。A(i)=A(i')明显不对了。所以命题得证。

得到了这两个定理之后，我们可以发现，任何一个**S**态，我们都可以通过自己的控制将它转化成**T**态。而对方怎么做都是将**T**态再转回**S**态，很被动。所以只要先手是**S**态，总是可以根据定理**1**得到的策略获胜。

题目**2**：有**n**堆石子，第**i**堆有**A(i)**颗石子。两人依次从中拿取，规定每次只能从一堆中取若干根，可将一堆全取走，但不可不取，最后取完者为负，求必胜的方法。

再来定义几个状态。一堆石子里只有一个石子，记为孤单堆。否则记为充裕堆。

在T态中，如果充裕堆的个数大于等于2，记为T2态，1个充裕堆，记为T1态，没有充裕堆，记为T0态。

S0、S1、S2同理。

【定理**3**】在**S0**态中，若孤单堆的个数为奇数。那必输。**T0**态必赢。
【证明】也就是奇数个石子，每次取一个。很明显先去的人必输。

【定理**4**】在**S1**态中，方法正确就必胜。
【证明】如果孤单堆的个数是奇数，那么就把充裕堆取完；如果是偶数，就把充裕堆取的只剩1根。这样留下的就是奇数个孤单堆，对方先手。由定理3得，对方必输，己方必胜。

【定理**5**】**S2**态不可一次变为**T0**态。
【证明】显然，充裕堆不可能一次从2堆以上变为0。

【定理**6**】**S2**态可一次变为**T2**态。
【证明】由定理1得S态可以一次变为T态，而且不会一次取完整堆，那么充裕堆的个数是不会变的，由定理5得S2态不能一次变为T0态，那么转化的T态是T2态。

【定理**7**】**T2**态只能转变为**S1**或**S2**态。
【证明】由定理2得，T态一次只能变为S态。由于充裕堆数不会变为0。所以是S1或S2态。

【定理**8**】在**S2**态中，只要方法正确，就必胜。
【证明】由定理6得，先转化为T2态。由定理7，对方只能再转化回S1或S2态。由定理4，己方必胜。

【定理**9**】**T2**态必输。
【证明】同证明8。

我们得到了几个必胜态：**S2**，**S1**，**T0**。必输态：**T2**，**T1**，**S0**。

比较一下两题：
第一题的过程：S2-T2-S2-T2-.....-T2-S1-T0-S0-T0-...-S0-T0（全0）
第二题的过程：S2-T2-S2-T2-.....-T2-S1-S0-T0-S0-...-S0-T0（全0）
我们可以发现前面的过程是一样的。关键在于得到了S1态之后，怎样抉择使自己获胜。而这个是自己可以掌握的。
因此，我们只需要把T2态留给对方，迟早他会转化成S1态。己方就必胜。

来源： http://dzy493941464.is-programmer.com/posts/39629.html

代码

```
#include<bits/stdc++.h>
using namespace std;
typedef long long LL;
int main()
{
    int _;
    cin>>_;
    while(_--)
    {
        int n;
        cin>>n;
        int flag = 0;
        int x;
        int sum = 0;
```

```
        for(int i=1;i<=n;i++)
        {
            cin>>x;
            sum = sum^x;
            if(x!=1)
            flag = 1;
        }
        if((sum==0 && flag==0)||(sum!=0&&flag!=0))
        {
            cout<<"John"<<endl;
        }
        else
        {
            cout<<"Brother"<<endl;
        }
    }
}
```