

## 夜空 (night)

### 【题目背景】

传说在很久以前，小怪兽 Nexus 作恶多端，大法师便将其封印于夜空之中。为了完成封印，大法师施展法术重排了星辰，令夜空呈现出特定的星象。

据说这道封印一直留存至今，再无人知晓它昔日的全貌。

### 【题目描述】

小 H 在阅读古籍时发现，夜空中的星辰可以抽象为一个非负整数序列。远古时期的星辰序列为  $A = [a_1, \dots, a_n]$ ，而如今所观测到的序列则为  $B = [b_1, \dots, b_m]$ 。

古籍中还记载了大法师重排星辰时使用的两种法术。具体而言，对于当前长度不小于 2 的星辰序列，大法师可以施展以下两种法术之一：

1. 删除序列最左侧的两个元素，并在最右侧插入它们的异或和；
2. 删除序列最右侧的两个元素，并在最左侧插入它们的异或和。

可以发现，每施展一次法术，星辰序列的长度都会恰好减少 1。小 H 猜测，或许大法师当年仅使用了这两种法术，恰好施展  $n - m$  次，就将最初的序列  $A$  变成了如今的序列  $B$ 。你需要帮助他判断这是否可能；若可能，还需找出一组具体的施法方案。

### 【输入格式】

从文件 *night.in* 中读入数据。

本题包含多组测试数据。

输入的第一行包含两个非负整数  $c, t$ ，分别表示测试点编号与测试数据组数。 $c = 0$  表示该测试点为样例。

接下来依次输入每组测试数据，对于每组测试数据：

- 第一行包含两个正整数  $n, m$ 。
- 第二行包含  $n$  个非负整数  $a_1, \dots, a_n$ 。
- 第三行包含  $m$  个非负整数  $b_1, \dots, b_m$ 。

### 【输出格式】

输出到文件 *night.out* 中。

对于每组测试数据：

- 第一行输出一个字符串 Yes 或 No，表示大法师是否可能仅施展了这两种法术就将序列  $A$  变成了序列  $B$ 。
- 若可能，则第二行输出  $n - m$  个  $\{1, 2\}$  中的正整数，分别表示大法师每次施展的法术种类。

正确回答上述第一项即可获得部分分数，具体评分规则请参见【评分方式】。

**【样例 1 输入】**

```
1 0 5
2 2 2
3 1 2
4 1 2
5 3 2
6 3 4 2
7 6 3
8 5 3
9 2 3 4 5 6
10 6 1 1
11 6 1
12 1 2 3 4 5 6
13 0
14 7 2
15 3 3 5 6 1 2 8
16 11 3
```

**【样例 1 输出】**

```
1 Yes
2
3 Yes
4 2
5 Yes
6 1 1
7 No
8 Yes
9 2 1 2 1 1
```

**【样例 1 解释】**

该样例共包含五组测试数据。

对于第一组测试数据，序列  $A$  与序列  $B$  相同，因此无需施展任何法术。

对于第二组测试数据，施展一次第 2 种法术，删除了序列  $A$  最右侧的 4 和 2，并在最左侧插入  $4 \text{ xor } 2 = 6$ ，得到序列  $B$ 。

对于第三组测试数据,

- 施展一次第 1 种法术, 删除了序列  $A$  最左侧的 2 和 3, 并在最右侧插入  $2 \text{ xor } 3 = 1$ , 得到序列  $[4, 5, 6, 1]$ ;
- 再次施展一次第 1 种法术, 删除了最左侧的 4 和 5, 并在最右侧插入  $4 \text{ xor } 5 = 1$ , 得到序列  $B$ 。

对于第四组测试数据, 可以证明, 仅施展这两种法术无法将序列  $A$  变成序列  $B$ 。

### 【样例 2】

见选手目录下的 *night/night2.in* 与 *night/night2.ans*。  
该样例满足测试点 1, 2 的约束条件。

### 【样例 3】

见选手目录下的 *night/night3.in* 与 *night/night3.ans*。  
该样例满足测试点 4 的约束条件。

### 【样例 4】

见选手目录下的 *night/night4.in* 与 *night/night4.ans*。  
该样例满足测试点 7 ~ 9 的约束条件。

### 【样例 5】

见选手目录下的 *night/night5.in* 与 *night/night5.ans*。  
该样例满足测试点 12 ~ 14 的约束条件。

### 【数据范围】

对于所有测试数据, 均有:

- $1 \leq t \leq 10^3$ ;
- $1 \leq m \leq n \leq 250$ ;
- 对于所有  $1 \leq i \leq n$ , 均有  $0 \leq a_i < 2^{30}$ ;
- 对于所有  $1 \leq i \leq m$ , 均有  $0 \leq b_i < 2^{30}$ 。

测试点编号	$n \leq$	$m \leq$	$T \leq$	特殊性质
1, 2	16	16	$10^3$	无
3	250	1	30	
4		2		
5, 6				A
7 ~ 9	50	50	50	B
10, 11	250	250	30	
12 ~ 14	50	50	50	C
15, 16	250	250	30	
17 ~ 19	50	50	50	D
20, 21	250	250	30	
22, 23	50	50	50	无
24, 25	250	250	30	

定义序列  $S = [s_1, \dots, s_k]$  是**奇异的**，当且仅当  $k \geq 3$ ，且  $s_1 = s_2 = s_3 = 1$ ，且对于所有  $4 \leq i \leq k$ ，均有  $2 \mid s_i$ 。

定义序列  $S = [s_1, \dots, s_k]$  的**循环移位**如下：对于正整数  $p$  ( $1 \leq p \leq k$ )，序列  $[s_p, s_{p+1}, \dots, s_k, s_1, \dots, s_{p-1}]$  是序列  $S$  的一个循环移位。

特殊性质 A:  $3 \mid n$ 。

特殊性质 B: 序列  $A, B$  都是**奇异的**。

特殊性质 C: 序列  $A$  与序列  $B$  各自存在一个**循环移位**是**奇异的**。

特殊性质 D:  $2m \geq n$ 。

### 【评分方式】

本题包含两个小问。对于每个测试点：

- 小问 1: 对该测试点中每组测试数据，正确判断可行性，即可获得该测试点 50% 的分数；
- 小问 2: 在此基础上，若对每组答案为 Yes 的测试数据还能正确给出一组合法的施法方案，即可获得该测试点另外 50% 的分数。

注意：对于答案为 Yes 的测试数据，无论选手是否尝试给出正确的施法方案，都需要在第二行输出  $n - m$  个  $\{1, 2\}$  中的正整数，以满足输出格式。

### 【提示】

本题目录下提供了一份 `checker.cpp` 用于检验施法方案的可行性。注意：提供的 `checker.cpp` 只会检验答案为 Yes 的测试数据中施法方案的正确性，而不会检查可行性判断的正确性。

选手可以在本题目录下使用如下命令编译得到可执行程序：

```
1 g++ checker.cpp -o checker -std=gnu++14 -O2 -static
```

编译得到可执行程序后，选手可以在本题目录下使用如下命令进行测试：

```
1 ./checker <input_file> <output_file>
```

其中 `<input_file>` 与 `<output_file>` 分别表示输入文件与输出文件的路径。

**注意：**选手提供的输入文件需满足题目给定的输入格式与数据范围，输出文件需满足给定的输出格式，否则不保证检验结果的正确性，并可能发生无法预料的错误。