



HPC³ 2024

问题 E, 中文

撕碎的秘密

最高分：40

你最近因在休息室吃零食和挪用公款而被国际海盗宝藏安全委员会 (IPTS C) 解雇。这意味着他们会抹去你的记忆，把你的所有文件都撕碎，然后扔进垃圾桶。这是一个可怕的消息，因为你恰好知道委员会监视下所有宝藏的秘密位置。

然而，你在另一个部门有一个未被发现的同伙，他可以从垃圾桶里找到碎纸。你计划将宝藏的地点告知你的同伙，然后他就会去寻找宝藏，这样你们俩都会变得非常富有。

宝藏可以表示为 P 大小为 n 的网格上 $N \times N$ ($1 \leq N \leq 10^5$) 长度为 l 的点数组。您将编写一个任意长度的非负整数数组 K ，该数组将由碎纸机以某种方式修改。然后，您的同伙将收到修改后的整数，并且必须确定原始的 P 。

笔记

- 此问题要求您编写两个程序：一个程序将点数组转换为整数数组，另一个程序将修改后的整数数组转换回点数组。HPC³ 为您提供了两种方法来实现此目的：提交两个不同的文件或提交一个包含两个不同函数的文件。
如果选择第二种方法，则必须隔离所有变量。
- 此问题有特殊评分。提交的内容将根据测试用例准确率、运行时间、内存使用情况以及除以的值进行评分 l 。 n 得分最高的是 $\frac{l}{n} \leq 3z$ ，且为 $0 < \frac{l}{n} > 10z$ 。
- 这个问题具有随机性。因此，它可能会出现不确定的结果。但是，正确的解决方案始终是最优的，因此始终能获得最高分。

子问题 1

碎纸机是标准的 IPTSC 碎纸机。它所做的就是将数组切分成各个元素并对其进行打乱。正式来说，对于数组 K ，碎纸机将为您提供 \hat{K} ，元素 K 被随机重新排列。

的价值 z 对于这个子问题，评分标准中的分数是1。

可以写入的最大整数大小为 K 。 10^6 给定 P ，确定 a K ，然后给定 \hat{K} ，确定的原始值 P 。

输入 A 格式

每个输入的第一行包含 1 个整数 n 。

每个输入的第二行包含 n 整数对：数组的内容 P 。

```
n
P[0][0] P[0][1] P[1][0] P[1][1] ... P[n-1][0] P[n-1][1]
```

输出 A 格式

每个输入的第一行包含 1 个整数 l 。

每个输入的第二行包含 l 整数：数组的内容 K 。

```
l
K[0] K[1] K[2] ... K[l-1]
```

输入 B 格式

每个输入的第一行包含 1 个整数 l 。

每个输入的第二行包含 l 整数：数组的内容 \hat{K} 。

```
l
 $\hat{K}[0]$   $\hat{K}[1]$   $\hat{K}[2]$  ...  $\hat{K}[l-1]$ 
```

输出 B 格式

每个输入的第一行包含 1 个整数 n 。

每个输入的第二行包含 n 整数对：数组的内容 P 。

```
n
P[0][0] P[0][1] P[1][0] P[1][1] ... P[n-1][0] P[n-1][1]
```

示例测试用例

输入 1A

```
2
1 2 4 2
```

输出 1A

```
6
1 2 4 2 1 1
```

输入 1B

```
6
1 4 2 1 2 1
```

输出 1B

```
2
4 2 1 2
```

[1, 2, 4, 2, 1, 1] 在随机化过程中变为 [1, 4, 2, 1, 2, 1]。请注意，程序可能会以与输入不同的顺序响应点。

子问题 2

粉碎机不是粉碎机，而是数据混淆设备！它的工作原理如下：它有一个非负整数 d ($1 \leq d \leq 100$) 和一个 A 长度为 a 的二进制数组 a ($1 \leq a \leq l$)。对于给定的 K 中的每个元素 K_i ($0 \leq i < l$) 如果 $i \bmod a$ 的第个元素 A 为 1，则 \hat{K}_i 将 K_i 在 K_i 之间添加一个随机整数值 $(-d, d)$ 。否则， \hat{K}_i 将是 K_i 。

的价值 z 在评分标准中²针对的是这个子问题。

可以写入的最大整数大小为 K 。10⁵ 给定 P 、 A 和 d ，确定 a K ，然后给定 \hat{K} ，确定的原始值 P 。

输入A 格式

每个输入的第一行包含 3 个整数 n 、 d 和 a 。

每个输入的第二行包含 n 整数对：数组的内容 P 。每个输入的第三行包含 a 二进制值：数组的内容 A 。

```
n d a
P[0][0] P[0][1] P[1][0] P[1][1] ... P[n-1][0] P[n-1][1]
A[0] A[1] A[2] ... A[a-1]
```

输出 A 格式

每个输入的第一行包含 1 个整数 l 。

每个输入的第二行包含 l 整数：数组的内容 K 。

```
l
K[0] K[1] K[2] ... K[l-1]
```

输入 B 格式

每个输入的第一行包含 1 个整数 l 。

每个输入的第二行包含 l 整数：数组的内容 \hat{K} 。

```
1
 $\hat{K}[0]$   $\hat{K}[1]$   $\hat{K}[2]$  ...  $\hat{K}[l-1]$ 
```

输出 B 格式

每个输入的第一行包含 1 个整数 n 。

每个输入的第二行包含 n 整数对：数组的内容 P 。

```
n
P[0][0] P[0][1] P[1][0] P[1][1] ... P[n-1][0] P[n-1][1]
```

示例测试用例

输入 1A

```
4 5 2
1 3 1 4 3 3 4 3
0 1
```

输出1A

```
8
7 14 18 19 19 23 24 24
```

输入 1B

```
8
7 18 18 24 19 23 24 29
```

输出 1B

```
4
1 3 1 4 3 3 4 3
```

d 为 5, 且 a 为 $[0, 1]$, 因此每 2 个输入都可修改 $\{-5, 5\}$ 。这样, $[7, 14, 18, 19, 19, 23, 24, 24]$ 变为 $[7, 18, 18, 24, 19, 23, 24, 29]$ 。

变化为 $[0, 4, 0, 5, 0, 0, 0, 5]$ 。

子问题 3

IPTSC 的安全性非常高! 粉碎机是前两个子问题中的机器的组合。它首先会将阵列随机化, 然后对其应用数据混淆设备过程。

的价值 z 在评分标准中 $\frac{5}{3}$ 针对的是这个子问题。

可以写入的最大整数大小为 K 。给定 P 、 A 和 d , 确定 a K , 然后给定 \hat{K} , 确定的原始值 P 。

输入 A 格式

每个输入的第一行包含 2 个整数 n 、 d 和 a 。

每个输入的第二行包含 n 整数对: 数组的内容 P 。每个输入的第三行包含 a 二进制值: 数组的内容 A 。

```
n d a
P[0][0] P[0][1] P[1][0] P[1][1] ... P[n-1][0] P[n-1][1]
A[0] A[1] A[2] ... A[a-1]
```

输出 A格式

每个输入的第一行包含 1 个整数 l 。

每个输入的第二行包含 l 整数: 数组的内容 K 。

```
l
K[0] K[1] K[2] ... K[l-1]
```

输入 B 格式

每个输入的第一行包含 1 个整数 l 。

每个输入的第二行包含 l 整数：数组的内容 \hat{K} 。

```
l
 $\hat{K}[0]$   $\hat{K}[1]$   $\hat{K}[2]$  ...  $\hat{K}[l-1]$ 
```

输出 B 格式

每个输入的第一行包含 1 个整数 n 。

每个输入的第二行包含 n 整数对：数组的内容 P 。

```
n
P[0][0] P[0][1] P[1][0] P[1][1] ... P[n-1][0] P[n-1][1]
```

示例测试用例

输入 1A

```
5 20 3
1 1 2 2 3 3 4 4 5 5
0 1 1
```

输出 1A

```
25
4 7 9 2 6 22 25 27 29 38 41 44 36 43 53 56 59 51 55 67 70 7
3 68 72 82 85 89 83 87 97 99 96 98 94
```

输入 1B

```
25
41 93 6 98 68 25 23 4 50 93 29 50 29 54 92 36 96 14 73 93 5
1 6 56 68 71 43 51 87 6 94 83 25
```

输出1B

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

[4, 7, 9, 2, 6 22, 25, 27, 29, 38, 41, 44, 36, 43, 53, 56, 59, 51, 55, 67, 70, 73, 68, 72, 82, 85, 89, 83, 87, 97, 99, 96, 98, 94] 变成

[41, 93, 6, 98, 68, 25, 23, 4, 50, 93, 29, 50, 29, 54, 92, 36, 96, 14, 73, 93, 51, 6, 56, 68, 71, 43, 51, 87, 6, 94, 83, 25]。

阵列随机化之前的变化包括：

[0, 4, -2, 0, -6, 9, 0, 7, -10, 0, 5, 3, 0, -1, 1, 0, 3, -7, 0, 8, -3, 0, 2, -5, 0]。