



HPC<sup>3</sup> 2024

## 问题 F, 中文

银河和平

最高分数 : 50

---

你是一名星际贸易商, 从一个太阳系旅行到另一个太阳系。太阳系是三维笛卡尔坐标系中的一个长方体, 其中一个角位于  $(0, 0, 0)$ , 宽度 $W$ 、高度 $H$ 和长度分别为 $L$  ( $1 \leq W \leq 10^3, 1 \leq H \leq 10^3, 1 \leq L \leq 10^3$ )。您从矩形内的坐标进入太阳系 $(x_A, y_A, z_A)$ , 然后必须移动到矩形内另一个坐标处的出口点 $(x_E, y_E, z_E)$ 。您的飞船配备了曲速引擎, 可以穿越广阔的太空。

该引擎允许您从当前点传送到矩形内的任何给定点。

然而, 有许多行星由长度 $B$ 为 $3l$  ( $0 \leq l \leq 50$ )其中每个 $(B_{i_x}, B_{i_y}, B_{i_z})$ 是行星的位置。每个行星的引力场都会干扰驱动器。每个场都是一个完美的球体, 中心位于每个行星的位置, 每个场的半径由一个整数数组给出 $R$  ( $1 \leq R_i \leq 100$ )长度为 $l$ 。如果你使用驱动器通过行星的引力场进行传送, 你将永远被困在虚空中。正式地说, 如果你在扭曲的点之间画一条最短的线, 并且该线上存在一个点, 使得该点与任何行星之间的距离小于该行星引力场的半径, 那么你就失败了。

您想要快速通过系统, 因此请在矩形内找到一个 $P$ 任意大小的最小坐标点数组 $v$ , 使得从入口点开始, 依次扭曲到每个点,  $P$ ,然后扭曲到出口点, 不会导致您穿越任何行星的引力场。

## 子问题 1

问题如上所述，找到一组点， $P$  使得在每个点之间顺序绘制的线不与  $B$  半径的球体相交  $R$ 。

给予  $W, H, L, l, x_A, y_A, z_A, x_E, y_E, z_E, B$ , and  $R$ , 回報  $P$ 。

### 输入格式

每个输入的第一行包含 10 个整数  $l, W, H, L, x_A, y_A, z_A, x_E, y_E$ , and  $z_E$ 。

每个输入的第二行包含  $3l$  整数：数组的内容  $B$ 。每个输入的第三行包含  $l$  整数：数组的内容  $R$ 。

```
l W H L x_A y_A z_A x_E y_E z_E
B[0][0] B[0][1] B[0][2] ... B[l-1][0] B[l-1][1] B[l-1][2]
R[0] R[1] R[2] ... R[l-1]
```

### 输出格式

每个输入的第一行包含 1 个整数  $v$ 。每个输出的第二行包含  $3v$  整数：数组的内容  $P$ 。

```
v
P[0][0] P[0][1] P[0][2] ... P[v-1][0] P[v-1][1] P[v-1][2]
```

## 示例测试用例

### 输入 1

```
3 14 26 50 4 14 7 48 14 7
15 13 7 36 16 7 46 18 7
7 6 3
```

## 输出 1

```
1
7 2 7
```

入口和 (7, 2, 7) 之间的线不跨越任何田地, (7, 2, 7) 和出口之间的线不跨越任何田地, 并且 的最小长度为 1。因此, 程序可以输出此结果。请注意, 有许多有效的可能  $P_s$ 。

## 子问题 2

您刚刚购买了飞船的重大升级, 它可以让您更精确地穿越太阳系, 即精密点驱动器。它的运行方式与您的旧驱动器完全相同, 但可以处理实数点。问题仍然与所述相同, 找到一组点,  $P$  使得在每个点之间顺序绘制的线不与  $B$  半径为  $R$  的球体相交。但是, 所有值都可以是实数, 而不是整数。

给予  $W, H, L, l, x_A, y_A, z_A, x_E, y_E, z_E, B,$  and  $R$ , 回報  $P$ 。

### 笔记

- 由于答案必须合理评分, 答案将四舍五入到小数点后 5 位。因此, 此问题可以使用小数点后 5 位的数字来解决。

### 输入格式

每个输入的第一行包含 1 个整数  $l$  和 9 个实值  $W, H, L, x_A, y_A, z_A, x_E, y_E,$  and  $z_E$ 。

每个输入的第二行包含  $3l$  实数值: 数组的内容  $B$ 。每个输入的第三行包含  $l$  实数值: 数组的内容  $R$ 。

```
l W H L x_A y_A z_A x_E y_E z_E
B[0][0] B[0][1] B[0][2] ... B[1-1][0] B[1-1][1] B[1-1][2]
R[0] R[1] R[2] ... R[1-1]
```

## 输出格式

每个输入的第一行包含 1 个整数 $v$ 。每个输出的第二行包含  $3v$  实数：数组 的内容 $P$ 。

```
v  
P[0][0] P[0][1] P[0][2] ... P[v-1][0] P[v-1][1] P[v-1][2]
```

## 示例测试用例

### 输入 1

```
4 20.5 30.5 40.5 3.85 9.75 12.25 18.35 25.15 30.65  
5.3 15.3 35.4 15.6 15.8 30.3 14.98 16.7 20.8 5.25 15.8 30.88  
4.75 6.25 5.09 5.555
```

### 输出 1

```
2  
10.25 5.25 20.25 10.25 25.25 20.25
```

入口和 (10.25, 5.25, 20.25) 之间的线不穿过任何田地, (10.25, 5.25, 20.25) 和 (10.25, 25.25, 20.25) 之间的线不穿过任何田地, (10.25, 25.25, 20.25) 和出口之间的线不穿过任何田地, 并且 的最小长度为 2。因此, 程序可以输出此结果。