该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

提高级模拟赛

题目名称	量子随机排序（quantum）	星 穹 列 车（astral）	礼物分配（gift）	抓兔子（rabbit）
题目类型	传统型	传统型	传统型	传统型
目录	quantum	astral	gift	rabbit
可执行文件名	quantum	astral	gift	rabbit
输入/输出文件	与文件名相同			
时间限制	1 秒	2 秒	1 秒	4 秒
内存限制	512MB	512MB	512MB	512MB
测试点数目	20	40	20	20
测试点是否等分	是	是	是	是

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

量子随机排序（quantum）

【题目描述】

大样例：quantum.zip

checker：checker.zip （已包含 testlib.h）

V 试图侵入荒坂公司的主服务器，获取关键的机密数据。然而，这台服务器采用了先进的安全防护机制，其中最核心的一环是一个基于**量子快速排序算法**的防御系统。

这个防御系统会对一个数据序列不断进行量子快速排序，以防止未经授权的访问。

数据序列由 n 个数组成，初始序列为 $a_1, a_2, \cdots, a_n$。

每当有人试图访问时，防御系统会自动触发量子随机排序算法，对数据序列进行重新排列。而只有当数据序列被排列成一个特定的序列 $b_1, b_2, \cdots, b_n$ 时，V 才能绕过防御系统，成功获取机密数据。

服务器使用的量子快速排序算法如下：

```
void quick_sort(vector<int> &a, int l, int r) {
    if(l >= r) return;
    int x = quantum_rand() % (r - l + 1) + l; // 量子随机数生成器
    swap(a[x], a[r]);
    int j = l;
    for (int i = l; i < r; ++i) {
        if(quantum_rand()) { // 量子随机数生成器
```

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

```
                swap(a[i], a[j++]);
            }
        }
        swap(a[j], a[r]);
        quick_sort(a, l, j - 1);
        quick_sort(a, j + 1, r);
}
void shuffle(vector<int> &a) {
        quick_sort(a, 0, a.size() - 1);
}
```

其中，quantum_rand() 是量子随机数生成器，可以用于产生随机数。

不过，V 已经成功破解了这个量子随机数生成器，可以通过干预指定每次 quantum_rand() 的返回值。

quantum_rand() 的调用需要大量时间，由于 V 的时间有限，最多只能等待 quantum_rand() 执行 k 次。一旦 quantum_rand() 的调用超过了 k 次，即视为任务失败。

V 的目标是，让初始序列 $a_1, a_2, \cdots, a_n$ 在调用 shuffle(a) 后，通过指定前 k 次 quantum_rand() 的返回值，使得最终的序列变成

该文档是极速PDF编辑器生成
如果想去掉该提示,请并下载
http://www.jisupdfeditor.com/

$b_1, b_2, \cdots, b_n$。

若任务无法完成，请输出 -1；否则，输出 k 次 quantum_rand 的返回结果。答案可能不止一种，输出任意一种满足条件的答案即可。

【输入格式】

第一行包含两个用空格分隔的整数 n, k。

第二行包含 n 个用空格分隔的整数 $a_1, a_2, \cdots, a_n$。

第三行包含 n 个用空格分隔的整数 $b_1, b_2, \cdots, b_n$，保证 b 是 a 的一个排列。

【输出格式】

如果让序列 a 变成 b 所需的quantum_rand 的调用次数必然超过 k，则输出 -1。

否则，输出一行 k 个用空格分隔的整数 $r_1, r_2, \cdots, r_k$ $(0 \leq r_i < n)$ 代表第 i 次调用 quantum_rand 时，返回的结果是 r_i。

答案可能不止一种，输出任意一种满足条件的答案即可。

注意：

如果你的输出导致 quantum_rand 调用的次数超过了 k 次，那么在第 $k+1$ 次调用时会返回 Wrong Answer。

如果 quantum_rand 实际调用的次数小于 k 次，那么多余的返回值不会有任何影响。

但是，你仍然要保证你输出的每个 r_i 在规定的范围内 $(0 \leq r_i < n)$

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

【样例 1 输入】

3 3

1 2 3

2 3 1

【样例 1 输出】

2 0 1

【样例 2 输入】

3 2

1 2 3

2 3 1

【样例 2 输出】

−1

【样例 3 输入】

3 2

1 1 1

1 1 1

【样例 3 输出】

−1

【样例 3 说明】

虽然 a 最终一定会排列成 b，但是 quantum_rand() 的调用次数必然超过 2。

因此由于时间原因，任务无法完成。

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

【备注】

对于所有测试数据，保证：

$1 \leq n \leq 5 \times 10^4, 1 \leq k \leq 10^6$，$1 \leq a_i \leq n$

保证 b 是 a 的一个排列

对于 40% 的数据，满足 $1 \leq n \leq 1000, n^2 \leq k \leq 10^6$。

对于另外 20% 的数据，满足 $1 \leq n \leq 1000$。

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

星穹列车（astral）

【题目描述】

大样例：astral.zip

星穹列车是穿梭于星系之间的神秘交通工具。

整个星系可以视作一个有 n 个星球、m 条航线的**有向图**。每条航线连接两个星球，表示星穹列车可以从一个星球跃迁到另一个星球，跃迁所需的时间已知且固定。

星穹列车的跃迁依赖于星球上的**星门**，只有在星门开启时，列车才能从该星球出发，沿着特定的航线跃迁至下一个目的地。然而，即使星门处于关闭状态，星穹列车也可以正常抵达该星球。

每个星球上的星门会周期性地开启和关闭。第 i 个星球的星门开启时间由 l_i, r_i, t_i（$0 \le l_i \le r_i < t_i$）决定，表示星门以周期 t_i 进行循环，且仅在每个周期的 $[l_i, r_i]$ 时间段内开启。

当星穹列车从停止状态进入跃迁状态时，需要耗费 k 时间，并且只有在当前星球的星门开启时才能出发。（列车只需要在退出停止状态的那一时刻，星门处于开启状态）

例如，列车在时刻 T 选择了航线 (u, v, w) 并从停止状态出发。那么需要在时刻 T 时，星球 u 的星门处于开启状态，且列车会在时刻 $T + k$ 进入跃迁状态，沿着航线跃迁至星球 v，并在时刻 $T + k + w$ 到达。

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

当星穹列车到达某个星球后，有以下两种选择：

停止：退出跃迁状态，在星门开启时可以从停止状态再次出发。

继续跃迁：不退出跃迁状态，沿航线继续前进。但此时这个星球的星门必须处于开启状态。

星穹列车从起点星球 1 出发，初始状态为停止，初始时间为 0（对于所有星门，都是循环周期的 0 时刻）。请计算从起点星球到达每个星球所需的最短时间。

【输入格式】

第一行包含三个整数 n, m, k，分别表示星球的数量、航线的数量，以及进入跃迁状态需要耗费的时间。

接下来 m 行，每行包含三个整数 u_i, v_i, w_i，代表从 u_i 到 v_i需要花费 w_i 的时间。

接下来 n 行，每行包含三个整数 l_i, r_i, t_i，表示第 i 个星球的星门开启时间段和循环周期。

【输出格式】

输出一行，包含 n 个用空格分隔的整数，依次表示从起点星球 1 出发，到达每个星球所需的最短时间。

保证从起点出发可达所有点。

【样例 1 输入】

```
3 2 10
1 2 1
```

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

2 3 1

1 1 2

3 4 10

1 4 5

【样例 1 输出】

0 12 15

【样例 1 说明】

在第一个样例中，1 的可出发时间是 1、3、5、…，

2 的可出发时间是 3、4、13、14、…

从点 1 到点 2 的最短时间是在时刻 1 从点 1 出发到点 2，花费 1 + 10 + 1 = 12。即：列车一开始位于点 1，列车在时刻 1 时退出停止状态，在时刻 11 时进入跃迁状态，在时刻 12 时到达点 2。

从点 1 到点 3 的最短时间是在时刻 3 从点 1 出发到点 2，到达点 2 传送门仍然开着，可以继续前往点 3，花费 3 + 10 + 1 + 1 = 15

【样例 2 输入】

6 6 10

1 2 1

2 3 1

3 2 1

2 4 1

4 5 1

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

5 6 1

0 0 10

0 1 2

0 1 2

4 4 6

2 2 7

0 1 2

【样例 2 输出】

0 11 12 12 17 24

【样例 2 说明】

在第二个样例中，

点 1 到点 2 的最短时间是在时刻 0 从点 1 出发到点 2，花费 10 + 1 = 11

点 1 到点 3 的最短时间是在时刻 0 从点 1 出发到点 2 再到点 3，花费 10 + 1 + 1 = 12

点 1 到点 4 的最短时间是在时刻 0 从点 1 出发到点 2 再到点 4，花费 10 + 1 + 1 = 12

点 1 到点 5 的最短时间是在时刻 0 从点 1 出发到点 2，然后开始在点 2 和点 3 绕圈，在时刻 15 从点 2 出发到点 4 再到点 5，花费 15 + 1 + 1 = 17。

点 1 到点 6 的最短时间是在时刻 0 从点 1 出发到点 2，然后开始在点 2 和点 3 绕圈，在时刻 21 从点 2 出发到点 4 再到点 5

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

再到点 6，花费 21 + 1 + 1 + 1 = 24

【样例 3 输入】

6 5 2

1 2 3

2 3 1

3 4 1

4 5 1

5 6 1

0 0 10

0 1 2

0 0 3

0 0 5

1 1 10

0 1 2

【样例 3 输出】

0 5 6 7 11 12

【样例 3 说明】

第三个样例中，

到点 2 的最短时间是在时刻 0 从点 1 出发到点 2，花费 2 + 3 = 5

到点 3 的最短时间是在时刻 0 从点 1 出发到点 2 到点 3，花费 2 + 3 + 1 = 6

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

到点 4 的最短时间是在时刻 0 从点 1 出发到点 2 到点 3 到点 4，花费 $2+3+1+1=7$

到点 5 的最短时间是在时刻 0 从点 1 出发到点 2，然后停止，等到时刻 6 再出发到点 3 到点 4 到点 5，花费 $6+2+1+1+1=11$

到点 6 的最短时间是在时刻 0 从点 1 出发到点 2，然后停止，等到时刻 6 再出发到点 3 到点 4 到点 5 到点 6，花费 $6+2+1+1+1+1=12$

【备注】

对于所有测试数据，保证：

$2 \leq n, m \leq 5000$

$0 \leq k \leq 10^3$

$1 \leq u_i, v_i \leq n$，保证没有重边和自环。

$1 \leq w_i \leq 10^3$

$0 \leq l_i \leq r_i < ti$

$2 \leq t_i \leq 10$

保证从起点出发可达所有点。

对于 30% 的测试数据，保证有 $2 \leq n, m \leq 100$

对于另外 30% 的测试数据，保证 $2 \leq n, m \leq 1000$

对于另外 20% 的测试数据，保证 $k = 0$

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

礼物分配（gift）

【题目描述】

大样例：gift.zip

作为鹅鹕镇的新成员，为了增进与居民的关系，你决定给每位居民赠送一份精心挑选的礼物。

鹅鹕镇上共有 n 位居民，他们的编号分别是 $1, 2, 3, \cdots, n$。

你准备了 n 份礼物，编号分别为 $m+1, m+2, \cdots, m+n$，你打算将礼物分配给居民，使得每个居民都恰好获得一个礼物。

但是，鹅鹕镇的居民对礼物有着特别的偏好：**每位居民只喜欢那些编号是自己编号倍数的礼物。**

例如，编号为 i 的居民最爱的礼物编号是 $i, 2i, 3i, \cdots i$。

你希望计算出有多少种不同的分配方案，可以让每位居民都能收到一份最爱的礼物。

结果可能较大，请将结果对 10^9+7 取模。

【输入格式】

一行包含两个用空格分隔的整数 n, m。

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

【输出格式】

一行一个答案，表示方案数，对 10^9+7取模。

【样例 1 输入】

5 1

【样例 1 输出】

3

【样例 1 说明】

有下面三种方案:

$[6,2,3,4,5]$

$[2,6,3,4,5]$

$[3,2,6,4,5]$

【样例 2 输入】

3 19

【样例 2 输出】

2

【样例 2 说明】

有下面两种方案:

$[22,20,21]$

$[20,22,21]$

【样例 3 输入】

19 5

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

【样例 3 输出】

6

【样例 4 输入】

758807789 17

【样例 4 输出】

962746378

【备注】

对于所有测试数据，保证：

$1 \leq n \leq 10^9, 1 \leq m \leq 20$

对于 30% 的数据，保证 $1 \leq n \leq 20$

对于另外 30% 的数据，保证 $m = 1$

该文档 是极速 编辑器成 ，
如果要去掉该 请访问 ：
www.jisupdfeditor.com/

抓兔子（rabbit）

【题目描述】

大样例：rabbit.zip

兔子洞里有一只超可爱的兔子……帕格丽希望抓住它！

兔子洞可以被视为一个 $n \times m$ 的矩形方格，每个方格可能是空地或障碍。

帕格丽每次可以向上、下、左、右四个方向中的一个移动一格，但只能移动到没有障碍物的空地上。

当帕格丽靠近兔子时，兔子就会逃跑。兔子的转向和移动都在一瞬间完成。但只有帕格丽与兔子相邻时，兔子才会移动。

当帕格丽和兔子相邻时，兔子会立刻转向帕格丽的反方向，并尝试向前移动一格，如果无法移动（遇到障碍或边界），则按顺时针方向转动，继续尝试向前移动一格（但不会靠近帕格丽），如果任何方向都无法移动，则兔子会停在原地不动。

当帕格丽和兔子在同一格子中，即视为抓到兔子。

现在，帕格丽想知道，她是否有可能通过一系列移动最终抓到兔子。

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

【输入格式】

多组输入输出。第一行输入一个整数 T，代表数据组数。

对于每组测试数据：

第一行两个用空格分开的整数 n, m，表示方格的行数和列数。

接下来 n 行，每行一个长度为 m 的仅由 '.','#','P','R' 组成的字符串，代表方格状态。

'.'：表示空地，可以通行

'#'：表示障碍，不可通行

'P'：帕格丽的初始位置

'R'：兔子的初始位置

保证一定存在且仅存在一个 'P' 和 'R'。

【输出格式】

对于每组测试数据，输出一行。如果帕格丽可以抓住兔子，输出 Yes；否则，输出 No。

【样例 1 输入】

5

5 5

#####

#P.R#

#.#.#

#....

#####

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

4 4

####

#P.#

#R.#

####

10 4

#P..

#.#.

#...

#R##

...#

#..#

#.##

#...

#.#.

#...

10 4

#...

#.#.

#...

#R##

...#

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

#...

#.#.

#...

#.#.

#.P.

5 5

....#

.##..

.#.R.

.##..

...P#

【样例 1 输出】

Yes

No

No

Yes

Yes

【样例 1 说明】

（按 1 – index）

在第一组数据中，P 先把 R 赶到 (4,4)，然后 P 从 (4,3) 就可以把 R 赶进 (4,5)。

在第二组数据中，R 会先反应过来跑到 (3,3)，然后 P 和 R 只能绕

该文档是极速PDF编辑器生成，
如果想去掉该提示,请访问并下载：
http://www.jisupdfeditor.com/

圈圈。

在第三组数据中，看似 P 可以把 R 赶到 $(5,1)$，其实不行。P 和 R 还是只能绕圈圈。

【备注】

对于所有测试数据，保证：

$1 \leq T \leq 1000$

$2 \leq n,m \leq 500$

对于每个测试点，满足$\sum nm \leq 500000$

对于 20% 的测试数据，满足 $2 \leq n,m \leq 5$

对于另外 20% 的测试数据，满足 $2 \leq n,m \leq 20$

对于另外 40% 的测试数据，满足$2 \leq n,m \leq 100$