

NOI 模拟赛

Day 1

时间：2022 年 3 月 1 日 00:00 ~ 23:59:59

题目名称	人格面具	9 号迷宫	飞行城堡
题目类型	传统型	传统型	传统型
目录	persona	maze	flyburg
可执行文件名	persona	maze	flyburg
输入文件名	persona.in	maze.in	flyburg.in
输出文件名	persona.out	maze.out	flyburg.out
每个测试点时限	4.0 秒	3.0 秒	8.0 秒
内存限制	512 MiB	512 MiB	1024 MiB
测试点数目	7	10	9
测试点是否等分	否	否	否

提交源程序文件名

对于 C++ 语言	persona.cpp	maze.cpp	flyburg.cpp
-----------	-------------	----------	-------------

编译选项

对于 C++ 语言	-O2 -std=c++14
-----------	----------------

注意事项（请选手务必仔细阅读）：

- 评测时采用的机器配置为：AMD Ryzen 7 8845H @ 3.80 GHz，内存 32GB。上述时限以此配置为准。
- 采用捆绑测试，只有你通过一个子任务中所有数据，才能获得该子任务的分数。子任务之间必要的依赖关系。
- 选手提交的源程序请存放在子文件夹中。
- 文件名（包括程序名和输入输出文件名）必须使用英文小写。
- C++ 中函数 `main()` 的返回值类型必须是 `int`，值必须为 `0`。
- 对于因未遵守以上规则对成绩造成的影响，相关申诉不予受理。
- 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
- 程序可使用的栈空间大小与该题内存空间限制一致。
- 在终端中执行命令 `ulimit -s unlimited` 可将当前终端下的栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
- 若无特殊说明，每道题的代码大小限制为 100KB。
- 若无特殊说明，输入与输出中同一行的相邻整数、字符串等均使用一个空格分隔。

12. 输入文件中可能存在行末空格，请选手使用更完善的读入方式（例如 `scanf` 函数）避免出错。
13. 直接复制 PDF 题面中的跨页样例，数据将带有页眉页脚，建议选手直接使用对应目录下的样例文件进行测试。
14. 使用 `std::deque` 等 STL 容器时，请注意其内存空间消耗。
15. 请务必使用题面中规定的的编译参数，保证你的程序在本机能够通过编译。此外不允许在程序中手动开启其他编译选项，一经发现，本题成绩以 0 分处理。

人格面具 (persona)

【题目背景】

Xia o Optokia tou Squa Crait Fimoa Oltemia Baly
隐藏自己的容颜毫无用处，因为无论何时
Olizes Ranya Crait Selaf
都有人能看到真正的你
Tvulei Poet nai Hierti Farsha Bai
就不要再继续无谓的苏生了
Xia o Optokia Note Oll Olzy Imiyl folme Bis Fel ou Emernes Drone Imm Finz Ween
最后，我们每个人都会独自离开这充满谎言之地

【题目描述】

Tairitsu 是于绝望的世界中诞生的。
空中有 n 枚玻璃碎片。它们倒映着回忆。刻画着痛苦、背叛、嫉妒的回忆。
我们将第 i 枚碎片的回忆量化为一个四元组 $g_i = (a_i, b_i, c_i, d_i)$ 。
碎片的回忆之间有着相互作用。具体而言，我们定义回忆 (a, b, c, d) 与 (a', b', c', d')
相互作用产生的痛苦值 $f((a, b, c, d), (a', b', c', d')) = \min(a - a', b - b', c - c', d - d') + \max(a - a', b - b', c - c', d - d')$ 。
Tairitsu 想知道总的痛苦值之和，也即 $\sum_{i=1}^n \sum_{j=i+1}^n f(g_i, g_j)$ 。

【输入格式】

从文件 *persona.in* 中读入数据。
第 1 行包含一个正整数 n 。
接下来 n 行，每行包含四个正整数 a_i, b_i, c_i, d_i ，含义见上。

【输出格式】

输出到文件 *persona.out* 中。
输出 1 行，仅 1 个整数，表示答案。

【样例 1 输入】

```
1 4
2 5 3 10 3
3 9 3 9 5
4 7 5 2 4
5 3 8 8 2
```

【样例 1 输出】

```
1 134
```

【样例 2】

见选手目录下的 *persona/persona2.in* 与 *persona/persona2.ans*。

【子任务】

对于所有数据， $1 \leq n \leq 5 \times 10^5, 1 \leq a_i, b_i, c_i, d_i \leq 10^6$ 。

子任务	n	特殊性质	分值
1	$\leq 10^4$	无	10
2	$\leq 5 \times 10^4$		10
3	$\leq 10^5$		15
4	$\leq 2.5 \times 10^5$		15
5	$\leq 5 \times 10^5$	A	10
6		B	15
7		无	25

- 特殊性质 A: $b_i = c_i = d_i$ 。
- 特殊性质 B: $c_i = d_i$ 。

9 号迷宫 (maze)

【题目背景】

Esp ou Shelista Nai Dilem Arsia
世界的碎片没有理由
Elize imm Krey Phier Holo Miis imm Aim
所以至少在你迷失自我的时候
Taz Madie Ofa Ellif Slay
请微笑着原谅自己吧
Quim Herz Nai Yien Kelletta Lootaria Xia Mane Vell Dialoss
无法使用的心总有一天会枯萎

【题目描述】

Hikari 与 Tairitsu 在这世间游走。

她们来到了密室之中。

地面上摆放有 n 个箱子。第 i 个箱子初始有 a_i 枚残片。

她们意外地发现，空中还漂浮着 m 枚额外的残片。她们打算将额外的残片放入箱中，这也许是解开密室的线索。

每枚额外残片似乎映射着自己独特的画面——这意味着其只能放入特定的箱子中。具体而言，每枚额外残片都有参数 x_i, y_i ，表示其只能放入箱子 x_i 或 y_i 中。保证 $x_i \neq y_i$ 。

由于残片的相互作用，同一箱子中的残片数量不能太多。因此，Hikari 与 Tairitsu 希望放入所有额外残片后，残片数量最多的箱子的残片数量尽可能少。

然而，这是一个不稳定的世界——残片随时可能出现或消失。具体而言，有以下几种异象：

- **1 x y**: 出现了新的额外残片，参数为 x, y 。保证 $x \neq y$ 。
- **2 x y**: 有一枚额外残片消失了，其参数为 x, y 。保证这样的残片存在。
- **3 x**: 箱子 x 中新出现了一枚初始残片。相当于给 a_x 加 1。
- **4 x**: 箱子 x 中有一枚初始残片消失了。相当于给 a_x 减 1。保证操作前箱子 x 有初始残片。

自然的，她们希望你告诉她们在每次异象后，放入所有额外残片后，残片数量最多的箱子的残片数量的最小值。异象之间不是互相独立的。

【输入格式】

从文件 *maze.in* 中读入数据。

第 1 行输入 3 个非负整数 n, m, q ，分别表示箱子的数量、额外残片的数量、异象的数量。

接下来 1 行，输入 n 个非负整数 a_1, \dots, a_n ，分别表示箱子 $1, \dots, n$ 的初始残片数量。

接下来 m 行，每行输入两个非负整数 x_i, y_i ，表示额外残片 i 的参数。

接下来 q 行，每行输入 2 或 3 个整数，表示一次异象（格式见上）。

【输出格式】

输出到文件 *maze.out* 中。

输出共 $q + 1$ 行，第 1 行输出所有异象出现前的答案；接下来 q 行，第 i 行输出前 i 次异象发生后的答案。

【样例 1 输入】

```
1 4 5 3
2 0 2 0 1
3 3 4
4 2 3
5 1 2
6 1 3
7 1 4
8 1 2 4
9 2 1 3
10 4 2
```

【样例 1 输出】

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

【样例 1 解释】

- 所有异象出现前：额外残片 1 ~ 5 放于箱子 3,3,1,1,4 中。
- 第 1 次异象后：设新出现的残片为 6 号，额外残片 1 ~ 6 放于箱子 3,3,1,1,4,2 中。
- 第 2 次异象后：4 号残片消失，额外残片 1,2,3,5,6 放于箱子 3,3,1,1,4 中。
- 第 3 次异象后： a_2 减 1 变为 1，额外残片 1,2,3,5,6 放于箱子 3,3,1,1,2 中。

【样例 2】

见选手目录下的 `maze/maze2.in` 与 `maze/maze2.ans`。

【样例 3】

见选手目录下的 `maze/maze3.in` 与 `maze/maze3.ans`。

【样例 4】

见选手目录下的 `maze/maze4.in` 与 `maze/maze4.ans`。

【子任务】

对于所有数据， $1 \leq n \leq 500, 1 \leq m \leq 2 \times 10^5, 1 \leq q \leq 10^5, 1 \leq a_i \leq n, 1 \leq x_i, y_i, x, y \leq n, x_i \neq y_i, x \neq y$ 。

子任务	n	m	q	特殊性质	分值
1	≤ 10	≤ 50	≤ 50	无	4
2	$\leq 10^2$	$\leq 10^3$	≤ 400		12
3	≤ 500	$\leq 10^4$	$\leq 2,000$		14
4			$\leq 10^4$	A	10
5				无	16
6		$\leq 2 \times 10^5$	$\leq 25,000$		12
7			$\leq 10^5$	AB	6
8				A	10
9				B	6
10				无	10

- 特殊性质 A：没有 2,4 操作。
- 特殊性质 B：没有 3,4 操作。

飞行城堡 (flyburg)

【题目背景】

Haol Sanz Miz Elza end, Jac Verry Miz Lusia
触碰的手变成羽翼，干枯的话语变成光芒
a Stea Endroll ou Laiziin Karna Awter Crai
航线到达终点，在旅途中再次相逢
a Brait Ani Miz Laila, Ritto Aka Goss Pray
绽放的气息化作鸟儿，飞沫送出祝福
ou Elza Challen Endroll end Sartain Harmiy Lar Crai Lar
每当羽翼之音回响于终末的庆典，太阳将升起和涌出回忆

【题目描述】

空中，玻璃碎片，翩翩起舞。

它们在空中排成了整齐的阵列。每枚碎片都映射出不同的世界——或是纯洁，或是黑暗。我们记前者种类的碎片为 1，后者为 0。这样，矩阵化为一个 $n \times m$ 的 01 矩阵。初始时，所有碎片均为 0 状态。

Hikari——这位少女，拥有神的力量。她可以通过如下操作将阵列净化：

1. 在第 1 行，不重复地放置恰好 c 个分身。
2. 分身所在位置的碎片将变为 1。
3. 如果不存在下一行，退出流程；否则，所有分身下移一行；
4. 所有分身可以左右移动，但总的移动步数不超过 k 步，且分身不能碰撞；
5. 回到步骤 2。

执行完所有操作后，Hikari 会游历整个矩阵，欣赏各异的纯洁世界。显然，这要求矩阵中的 1 构成一条蛇。具体而言，如果把四联通意义下相邻的 1 连边，则形成的图应该是一条链。

Hikari 希望你帮助她寻找答案。你只要输出净化后的阵列即可。

【输入格式】

从文件 *flyburg.in* 中读入数据。

本题单个测试点中有多组测试数据。

输入的第 1 行包含 3 个正整数 T, c, k 。其中 T 表示数据组数， c, k 含义见上。特别的， c, k 在各组测试数据中保持不变。

接下来 T 行，每行包含 2 个正整数，表示 n, m ，含义见上。
请注意本题关于 c, k 的限制。

【输出格式】

输出到文件 *flyburg.out* 中。

对于每组数据，第 1 行输出 YES 或 NO，表示是否有解。

若有解，接下来输出 n 行，每行包含 m 个字符 0 或 1，表示净化后的矩阵。

如有多种方案，输出任意一种即可。

【样例 1 输入】

```
1 5 5
2 2
3 4 8
4 6 6
```

【样例 1 输出】

```
1 YES
2 11000111
3 10011101
4 10110011
5 11100110
6 NO
```

【样例 1 解释】

对于第 1 组数据，Hikari 可以通过如下操作将阵列净化为目标样式：

- 第 1 行选择 1, 2, 6, 7, 8 位置；
- 净化，下移至第 2 行，（从左至右）第 2 个分身右移 2 格，第 3 个分身左移 1 格，第 4 个分身左移 1 格，总移动次数为 4。
- 以此类推。

最后元素 1 构成的图形显然是一条链。

【样例 2】

见选手目录下的 *flyburg/flyburg2.in* 与 *flyburg/flyburg2.ans*。

注意，这组样例的答案文件中只会显示是否存在解，而没有给出具体解的构造。

【子任务】

对于所有数据， $1 \leq c \leq 8, c < k \leq 10^2, 1 \leq T \leq 500, 2 \leq n \leq 300, 1 \leq m \leq 300$ 。

子任务	c	k	n	分值
1	≤ 1	≤ 10	≤ 5	8
2	≤ 3		≤ 8	10
3	≤ 5	≤ 8	≤ 10	12
4			≤ 300	8
5	≤ 6	≤ 10	≤ 10	12
6		$\leq 10^2$	≤ 300	10
7	≤ 7	≤ 10		12
8		$\leq 10^2$		10
9	≤ 8			18