

NOI2018 模拟赛

竞赛时长：5 小时

| | | | |
|---------|----------|-----------|------------|
| 题目名称 | 细胞 | 种草 | 买票 |
| 题目类型 | 传统型 | 传统型 | 传统型 |
| 目录 | cell | green | ticket |
| 可执行文件名 | cell | green | ticket |
| 输入文件名 | cell.in | green.in | ticket.in |
| 输出文件名 | cell.out | green.out | ticket.out |
| 每个测试点时限 | 2 秒 | 3 秒 | 6 秒 |
| 内存限制 | 512 MB | 512 MB | 512 MB |
| 测试点数目 | 2 | 4 | 5 |
| 每个测试点分值 | 23~77 | 5~30 | 10~35 |

提交源程序文件名

| | | | |
|-----------|----------|-----------|------------|
| 对于 C++ 语言 | cell.cpp | green.cpp | ticket.cpp |
|-----------|----------|-----------|------------|

编译选项

| | | | |
|-----------|---------|---------|---------|
| 对于 C++ 语言 | -O2 -lm | -O2 -lm | -O2 -lm |
|-----------|---------|---------|---------|

注意事项：最终评测时，程序可以使用的栈空间上限与内存限制相同。

细胞 (cell)

【问题描述】

你成长为了一名生物学家。你的实验室里有一种新型生命体，这种生命体可以被看成是由 n 种细胞按一定顺序构成的序列。这 n 种细胞都以 1 分钟作为它们的生命周期：第 i 种细胞每经过 1 分钟会变成细胞序列 $H(i) = \{h_{i,1}, \dots, h_{i,l_i}\}$ ，其中 l_i 表示序列 $H(i)$ 的长度。

研究表明，细胞序列 S 具有重要的应用前景，因此你希望大量生产细胞序列 S 。但你现在只有由细胞 1 构成的单细胞生命体。幸运的是你已经掌握了细胞切割的技术，因此只要生命体的细胞序列中出现子序列 S ，你就能够将这一生命体切开，得到细胞序列 S 。注意，你没有掌握细胞拼接的技术。

你现在要求出只含细胞 1 的单细胞生命体最早能够在什么时候产生细胞序列 S 。

【输入格式】

输入的第一行为两个整数 n 和 m ，分别表示细胞的种类数和目标片段的长度。

接下来 n 行，第 i 行的第一个整数为 l_i ，表示第 i 种细胞分裂出来的细胞个数。接下来 l_i 个数 $h_{i,1}, h_{i,2}, \dots, h_{i,l_i}$ ，表示第 i 种细胞分裂出来的细胞序列。

最后一行 m 个整数，表示目标细胞序列 S 。

【输出格式】

输出一行一个整数，表示目标序列 S 最早出现的时间。如果目标序列不可能出现，那么输出 -1 。

【样例输入 1】

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

【样例输出 1】

```
3
```

【样例解释】

第 1 分钟细胞序列为 $\{1\}$;

第 2 分钟细胞序列为 $\{H(1)\} = \{2, 3\}$;

第 3 分钟细胞序列为 $\{H(2), H(3)\} = \{1, 3, 3, 1, 2\}$ ，此时目标片段 $\{3, 1\}$ 出现。

【数据规模与约定】

本题采用捆绑测试。

你需要通过一个子任务内的所有测试点才能得到该子任务的分数。

| 子任务 | 分值 | n, m | 其它约定 |
|-----|----|----------|---------|
| 1 | 23 | ≤ 5 | 答案不超过 5 |
| 2 | 77 | 无特殊约定 | 无 |

对于 100% 的数据，保证 $1 \leq n \leq 500, 1 \leq m \leq 1000, 1 \leq l_1 + l_2 + \cdots + l_n \leq 1000, l_i \geq 2$ 。

种草 (green)

【问题描述】

你有一片草原。这片草原可以看成大小为 $r \times c$ 的矩形网格, 西北角为 $(1, 1)$, 东南角为 (r, c) 。这片草原上有的网格种了草, 有的还只是一片荒芜。在一开始总共有 n 个网格 $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ 种了草。

你的目标是使这片草原充满绿色。你可以通过选定风向的方式来种草: 每一年你可以选择今年的风向为东南西北中的一个。以北风为例, 如果你选择今年刮北风, 并且网格 (x, y) 在今年长了草, 那么一年以后网格 $(x - 1, y)$ 也会被种上草。

你需要求出, 假如风向合适的话, 最少需要多少年才能使这片草原种满草。

【输入格式】

输入的第一行为两个整数 r, c , 表示草原的大小。

接下来一行一个整数 n , 表示一开始就有植被的网格。

接下来 n 行, 每行两个整数 x_i, y_i , 表示 (x_i, y_i) 在第一年就有草。

【输出格式】

输出一行一个整数, 表示最少需要的年份数。

【样例输入 1】

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

【样例输出 1】

```
3
```

【样例输入 2】

```
4 4
4
1 1
1 4
4 1
4 4
```

【样例输出 2】

4

【样例解释】

样例一的风向分别为西、南、南。

【数据规模与约定】

本题采用捆绑测试。

你需要通过一个子任务内的所有测试点才能得到该子任务的分数。

| 子任务 | 分值 | r | c | n |
|-----|----|-----------|-----------|------------|
| 1 | 5 | ≤ 4 | ≤ 4 | 无特殊约定 |
| 2 | 10 | ≤ 40 | ≤ 40 | |
| 3 | 15 | | | |
| 4 | 30 | 无特殊约定 | 无特殊约定 | ≤ 25 |
| 5 | 20 | | | ≤ 100 |
| 6 | 20 | | | 无特殊约定 |

对于 100% 的数据，保证 $1 \leq n \leq 300, 1 \leq x_i \leq r \leq 10^9, 1 \leq y_i \leq c \leq 10^9$ ，初始时刻草原没有被种满草。

买票 (ticket)

【问题描述】

有 n 个车站, 编号依次为 1 到 n 顺时针排列在一段环形铁路上。总共有 n 种车票, 第 i ($1 \leq i < n$) 种车票可以让一位乘客从第 i 个车站到第 $(i+1)$ 个车站, 或者从第 $(i+1)$ 个车站到第 i 个车站。第 n 种车票可以让一位乘客从第 1 个车站到第 n 个车站, 或者从第 n 个车站到第 1 个车站。

现在有 m 个需求, 第 i 个需求说有 c_i 个人要从第 a_i 个车站到第 b_i 个车站 (他们经过的路线可以不同)。

你现在要为他们买票, 票只能一套一套买, 一套车票共包含 n 张车票, 每种各一张。你需要求出最少要买多少套车票才能满足所有需求。

【输入格式】

输入的第一行为两个整数 n, m , 分别表示车站和需求的数量。

接下来 m 行, 每行三个整数 a_i, b_i, c_i , 含义如题述。

【输出格式】

输出一行一个整数, 表示最少要买多少套车票。

【样例输入 1】

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

【样例输出 1】

```
1
```

【样例输入 2】

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

【样例输出 2】

```
1
```

【样例输入 3】

6 3
1 4 1
2 5 1
3 6 1

【样例输出 3】

2

【样例解释】

对于样例 1，你只需要让所有人沿顺时针方向移动即可。
对于样例 2，可以让一个人沿 $1 \rightarrow 2$ 移动，另一个人沿 $2 \rightarrow 3 \rightarrow 1$ 移动。
对于样例 3，可以让三个人分别沿 $1 \rightarrow 2 \rightarrow 3 \rightarrow 4, 2 \rightarrow 1 \rightarrow 6 \rightarrow 5, 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ 移动。

【数据规模与约定】

本题采用捆绑测试。

你需要通过一个子任务内的所有测试点才能得到该子任务的分数。

| 子任务 | 分值 | n, m | c_i |
|-----|----|-------------|-------|
| 1 | 10 | ≤ 20 | = 1 |
| 2 | 35 | ≤ 300 | |
| 3 | 20 | ≤ 3000 | |
| 4 | 20 | 无特殊约定 | |
| 5 | 15 | | 无特殊约定 |

对于 100% 的数据，保证 $2 \leq n \leq 200000, 1 \leq m \leq 100000, 1 \leq a_i, b_i \leq n, a_i \neq b_i, 1 \leq c_i \leq 10^9$ 。