2025 CSP-S 模拟赛

题目名称	豪宴	奇巧	仙缘	离魂
题目类型	传统型	传统型	传统型	传统型
可执行文件名	`dinner`	`clever`	`fate`	`soul`
输入文件名	`dinner.in`	`clever.in`	`fate.in`	`soul.in`
输出文件名	`dinner.out`	`clever.out`	`fate.out`	`soul.out`
时间限制	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	256 MiB	256 MiB	256 MiB	256 MiB
测试点数目	10	10	10	10
测试点等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	`dinner.cpp`	`clever.cpp`	`fate.cpp`	`soul.cpp`

编译选项

对于 C++ 语言	`-O2 -std=c++14 -static`

1. C++ 中函数 main() 的返回值类型必须是 int，值必须为 0。
2. 若无特殊说明，输入文件中同一行内的多个整数、浮点数、字符串等均使用一个空格进行分隔。
3. 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。
4. 程序可使用的栈空间大小与该题内存空间限制一致。
5. 在终端下可使用命令 `ulimit -s unlimited` 将栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
6. 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。
7. 本场比赛未使用捆绑测试。
8. 考试过程中若对题目有疑问，请联系出题人。

豪宴（dinner）

【题目描述】

zak 的豪宴结束后，打乒乓球。

每局乒乓球比赛由两名选手进行，分别称为 A 和 B。每局比赛一定会产生一名胜者——因此乒乓球比赛不会出现平局。每局比赛的胜者将获得 1 分，另一名选手不得分。当同时满足以下两个条件时，两名选手之间的锦标赛立即结束：

- 至少有一名选手已经得到 m 分；
- 有一名选手领先对手至少 2 分。

zak 不理解《随机过程》，因为他能预测随机数———包括乒乓球比赛的结果，他给出了一条长度为 n 的字符串 $s = s_1 s_2 ... s_n$，其中每个字符为 ‘A‘ 或 ‘B‘。这里 ‘A‘ 代表选手 A，‘B‘ 代表选手 B，表示比赛中第 i 分将由 $s_{((i-1) \bmod n)+1}$ 所代表的选手获得。

你需要计算比赛结束前共会进行多少分，或者指出比赛永远不会结束。

【输入格式】

存在多组测试用例。第一行包含一个整数 T $(T \geq 1)$，表示测试用例的数量。

对于每组测试用例：

- 第一行包含两个整数 n 和 m。
- 第二行包含一个仅由 ‘A‘ 和 ‘B‘ 组成的字符串 $s = s_1 s_2 \ldots s_n$。

保证所有测试用例的 n 之和不超过 2×10^5。

【输出格式】

对于每组测试用例：

- 若锦标赛永远不会结束，输出单独一行，仅包含单词 `No`。
- 否则，第一行输出单词 `Yes`；第二行输出一个整数，表示锦标赛结束前将会进行的总分数。

【样例 1 输入】

```
3
1 11
A
2 11
AB
3 11
ABB
```

【样例 1 输出】

```
Yes
11
No
Yes
17
```

【样例 2】

见下发文件，该样例满足测试点 $1 \sim 5$ 的限制。

【样例 3】

见下发文件，该样例满足测试点 $1 \sim 10$ 的限制。

【解释与说明】

样例一解释：
对于第一组数据，容易验证选手 A 连胜 11 局后比赛结束。
对于第二组数据，选手 AB 交替得分，比分差距始终 ≤ 1，比赛拖入无尽加时。
对于第三组数据，容易验证在进行 17 局后为 $6:11$，B 获胜，比赛结束。

【数据范围】

本题共 10 个测试数据，每个测试数据 10 分。
对于所有数据，满足 $1 \leq n, T \leq 2 \times 10^5, 1 \leq m \leq 10^{18}$，$\sum n \leq 2 \times 10^5$。
对于 20% 的数据满足 $1 \leq n, m \leq 100$，$\sum n, \sum m \leq 100$。
对于另外 30% 的数据满足 $1 \leq n, m \leq 2 \times 10^5$，$\sum n, \sum m \leq 2 \times 10^5$。
对于另外 50% 的数据，无特殊限制。

奇巧（clever）

【题目描述】

有一栋共 10^9 层的大楼，却只有 1 部电梯。最初电梯停在第 f 层。有 n 个人正在等电梯，第 i 个人当前在第 l_i 层，想乘电梯到第 r_i 层（$l_i < r_i$）。由于电梯极小，一次最多只能载 1 人。

电梯每向上移动 1 层需消耗 1 单位电能；向下移动不耗能。换言之，从第 x 层到第 y 层耗能 $\max(y-x, 0)$ 单位。

请找出运送所有人的最优顺序，使总电能消耗最小。

形式化地，令 $a_1, a_2, \ldots, a_n$ 为 $1, 2, \ldots, n$ 的一个排列，其中 a_i 表示第 i 个被运送者的编号。总耗能为

$$\sum_{i=1}^{n}\Big(\max(l_{a_i} - r_{a_{i-1}}, 0) + r_{a_i} - l_{a_i}\Big),$$

这里为方便记 $a_0 = 0$，$r_{a_0} = f$。

回忆：长为 n 的序列 $a_1, a_2, \ldots, a_n$ 是 1 到 n 的排列，当且仅当 $1 \sim n$ 每个整数恰出现一次。

【输入格式】

多组测试用例。第一行给出整数 T（$1 \le T \le 10^4$），表示测试用例数。对于每组用例：第一行给出两个整数 n 和 f，分别表示人数与电梯初始楼层。接下来 n 行，第 i 行给出两个整数 l_i 和 r_i（$1 \le l_i < r_i \le 10^9$），表示第 i 个人欲从第 l_i 层到第 r_i 层。

【输出格式】

对于每组用例：第一行输出一个整数，表示最小总电能消耗；第二行输出 n 个整数 $a_1, a_2, \ldots, a_n$，用空格分隔，表示最优运送顺序。这些整数须构成 $1 \sim n$ 的排列。若有多种最优顺序，输出任意一种即可。

【样例 1 输入】

```
2
4 2
3 6
1 3
2 7
5 6
2 5
```

```
2 4
6 8
```

【样例 1 输出】

```
11
2 1 4 3
5
2 1
```

【样例 2】

见下发文件，该样例符合测试点 $3 \sim 4$ 的数据范围。

【样例 3】

见下发文件，该样例符合测试点 $5 \sim 6$ 的数据范围。

【样例 4】

见下发文件，该样例符合测试点 $7 \sim 8$ 的数据范围。

【样例 5】

见下发文件，该样例符合测试点 $9 \sim 10$ 的数据范围。

【解释与说明】

样例一解释：

对于第一组数据，一开始电梯位于 2

- 先运送 $(1,3)$ 这个人，花费 2，结束后位于 3。
- 然后运送 $(3,6)$，花费 3 结束后位于 6。
- 然后运送 $(5,6)$，花费 1，结束后位于 6。
- 最后运送 $(2,7)$，花费 5，最后停留在 7。
- 总开销 $2+3+1+5=11$。

注意，`2 1 4 3` 不是唯一的构造方法，任何满足最小花费的构造，例如 `2 3 4 1` 都会被认为是正确的。

对于第二组数据，一开始电梯位于 5

- 先运送 $(6,8)$ 这个人，花费 3，结束后位于 8。
- 然后运送 $(2,4)$，花费 2 结束后位于 4。
- 总开销 $3+2=5$。

【数据范围】

对于所有数据，满足 $1 \le n \le 10^5, 1 \le T \le 10^4, 1 \le f \le 10^9, l_i < r_i, \sum n \le 2 \times 10^5$。

对于 20% 的数据，满足 $n \le 10, T \le 3$。

对于另外 20% 的数据，满足 $\sum n \le 2000$。

对于另外 20% 的数据，满足不同人乘坐的电梯不存在相交关系，即 $\forall i \ne j, r_i < l_j \vee r_j < l_i$。

对于另外 20% 的数据，满足不同人乘坐的电梯不存在包含关系，即 $\forall i \ne j$，不满足 $l_i < l_j \wedge r_j < r_i$

对于另外 20% 的数据，无特殊限制。

仙缘 (fate)

【题目描述】

所有整体以相同速度向“地面”（矩阵最后一行）下落，且不做任何旋转。每秒，所有整体都尝试向下移动一行。若移动后某整体会越过矩阵下边界，则该整体立即停驻；若移动后某整体会与另一整体重叠（注意，后者此时必已停驻），则前者亦立即停驻。换言之，整体在下触地面或上触其他停驻整体时便停止下落。

请输出所有整体均停驻后的最终矩阵状态。

【输入格式】

第一行给出两个整数 N, M。

随后 N 行描述矩阵，每行连续 M 个字符，仅含”.” 或”#”，行间无多余空格。

【输出格式】

打印所有整体停落后的结果矩阵，格式与输入相同，但不再输出尺寸行。

【样例 1 输入】

```
10 10
..........
..######..
..#....#..
..#.#..#..
..#..#.#..
..#....#..
..######..
..........
..#....#..
.......#..
```

【样例 1 输出】

```
..........
..........
..######..
```

```
..#....#..
..#....#..
..#....#..
..#.##.#..
..######..
.......#..
..#....#..
```

【样例 2】

见下发文件，该样例符合测试点 $1 \sim 3$ 的数据范围。

【样例 3】

见下发文件，该样例符合测试点 $4 \sim 6$ 的数据范围。

【样例 4】

见下发文件，该样例符合测试点 $7 \sim 10$ 的数据范围。

【数据范围】

对于所有数据，满足 $1 \le N, M \le 2000$。
本题共有 10 个测试数据。
对于 30% 的数据满足 $1 \le N, M \le 400$。
对于另外 30% 的数据，空白区域连通，左上角方格 $(1, 1)$ 是空白单元格。
对于另外 40% 的数据，没有特殊性质。

离魂 (soul)

【题目描述】

你有一块大小为 $N \times M$ 的布料，但由于一场火灾，上面有若干洞。我们用一个 $N \times M$ 的 01 矩形来描述这块布料，为 1 的位置就是洞，为 0 的位置就是完好的。

Kraw 想要忘记那场大火，他希望能从布料上裁剪出一个矩形，把剩下的部分都扔掉。新的布料必须满足面积至少为 K，并且不能包含任何洞。

Kraw 只能沿着规则的网格线裁剪布料（只能横平竖直的剪）。Kraw 想知道，有多少种方法可以裁剪出一个面积至少为 K、且不包含任何洞的矩形。

【输入格式】

你的程序应从标准输入读取数据。输入包括：

- 一行，包含三个整数 N 和 M（$1 \le N, M \le 2000$），分别表示布料的高度和宽度，以及 K（$1 \le K \le MN$），即矩形的最小面积（以网格单元数计）；
- 接下来 N 行，每行包含 M 个整数 $s_{0y}, s_{1y}, \ldots, s_{(M-1)y}$。若坐标为 (x, y) 的网格单元有洞，则 $s_{xy} = 1$，否则 $s_{xy} = 0$。

【输出格式】

输出一行，包含一个整数：表示可以裁剪出多少种面积至少为 K、且不包含任何洞的矩形。

【样例 1 输入】

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

【样例 1 输出】

```
3
```

【样例 2】

见下发文件，该样例满足测试点 1 的约束。

【样例 3】

见下发文件，该样例满足测试点 $2 \sim 3$ 的约束。

【样例 4】

见下发文件，该样例满足测试点 $4 \sim 6$ 的约束。

【样例 5】

见下发文件，该样例满足测试点 $7 \sim 10$ 的约束。

【说明/提示】

样例一解释
可以从布料上裁剪出 3 个面积至少为 3 的矩形。以左上角为 $(0,0)$，它们分别是：

- 2 个面积为 3 的矩形——$\{(1,0),(2,0),(3,0)\}$，$\{(0,1),(1,1),(2,1)\}$
- 1 个面积为 4 的矩形——$\{(1,0),(2,0),(1,1),(2,1)\}$

【数据范围】

本题共 10 个测试点，每个测试点 10 分。
对于所有数据，满足 $1 \leq N, M \leq 2000$。

测试点编号	限制条件
1	满足 $0 < N, M \leq 2000$，$K = 1$ 且仅有一个 (x, y) 满足 $s_{xy} = 1$
$2 \sim 3$	满足 $0 < N, M \leq 500$
$4 \sim 6$	满足 $0 < N, M \leq 2000$ 且 $K = 1$
$7 \sim 10$	满足 $0 < N, M \leq 2000$