

NOIP2024 模拟赛

时间：4.5 小时

题目	灯	矩阵	折纸	钻石
程序	light.cpp	matrix.cpp	paper.cpp	diamond.cpp
输入文件	light.in	matrix.in	paper.in	diamond.in
输出文件	light.out	matrix.out	paper.out	diamond.out
时间限制	2s	2s	2s	2s
内存限制	512M	512M	512M	512M
测试点数量	10	20	10	10
是否等分	是	是	是	是
评测方式	spj	传统	传统	传统

编译选项

对于 C++ 语言：-O2 -std=c++14 -static

灯(light.cpp /2s/ 512M)

题目描述

有 n 个灯排成一排，有的灯是开着的，有的灯是关着的。第 i 盏灯的开关状态是 a_i (1 表示开, 0 表示关)。现在希望让第 i 盏灯的开关状态变成 b_i 。

有 m 个开关，第 i 个可以翻转 $[l_i, r_i]$ 中的所有灯的开关状态 (即 1 变成 0, 0 变成 1)。每种开关最多只能按一次。

你需要求出有多少种按开关的方案，使得所有灯都能够到达想要的状态。由于方案数可能很大，只需要输出答案对 $10^9 + 7$ 取模的结果。特别的，如果存在至少一组方案，你还需要给出一组构造。

输入格式

第一行共两个正整数 n, m 。

第二行一个长度为 n 的 01 串，表示 a_i 。

第三行一个长度为 n 的 01 串，表示 b_i 。

接下来 m 行每行两个正整数 l_i, r_i ，表示一个开关。

输出格式

第一行一个非负整数，表示方案数模 $10^9 + 7$ 。

第二行一个字符串 YES 或 NO，表示是否存在至少一种按开关的方案满足要求。

如果是 YES，那么第三行输出一个长度为 m 的零一串表示每个开关是否被按下，表示一种合法的方案。否则第三行不输出任何东西。

评分方式

如果方案数输出正确，获得该测试点 50% 的分数。

如果方案构造正确，获得该测试点另外 50% 的分数。

样例输入 1

```
3 3
010
101
1 2
2 2
3 3
```

样例输出 1

```
1
YES
101
```

样例输入 2

```
3 3
010
101
1 2
1 1
2 2
```

样例输出 2

```
0
NO
```

数据范围

测试点编号	$n, m \leq$	特殊性质
1	20	$\forall i, a_i = 0$
2	20	无
3	10^6	$\forall j, l_j = 1$
4	10^6	$\forall j, l_j = 1$
5	10^6	无
6	10^6	无
7	10^6	无
8	10^6	无
9	10^6	无
10	10^6	无

矩阵(matrix.cpp /2s/ 512M)

题目描述

给一个仅包含整数的 n 行 m 列的矩阵 A ，你需要构造一个 n 行 m 列的**整数**矩阵 B ，使得：

$$A_{i,j} = \sum_{k=1}^m B_{i,k} + \sum_{k=1}^n B_{k,j} - B_{i,j} \pmod{10^9 + 7}$$

或者判断无解。注意，如果有解，你需要输出**字典序最小**的 B 。字典序最小指：先让第一行第一个元素尽量小，再让第一行第二个元素尽量小.....以此类推。

输入格式

第一行两个正整数 n, m 。

接下来 n 行每行 m 个整数，表示矩阵 A 。

输出格式

若有解，第一行输出 1，否则输出 -1。

共 n 行，每行 m 个整数，表示矩阵 B 。你需要保证 B 中的元素在 $[0, 10^9 + 6]$ 之中。

样例输入 1

```
3 3
4 5 3
3 3 2
3 4 3
```

样例输出 1

```
1
1 1 1
0 1 0
1 1 0
```

样例输入 2

```
1 2
1 2
```

样例输出 2

```
-1
```

数据范围

对于所有数据, $1 \leq n, m \leq 1000, 0 \leq A_{i,j} < 10^9 + 7$ 。

测试点编号	n	m
1	≤ 2	≤ 10
2	≤ 2	≤ 10
3	≤ 2	≤ 10
4	≤ 2	≤ 1000
5	≤ 2	≤ 1000
6	≤ 2	≤ 1000
7	≤ 10	≤ 10
8	≤ 10	≤ 10
9	≤ 10	≤ 10
10	≤ 10	≤ 10
11	≤ 50	≤ 50
12	≤ 50	≤ 50
13	≤ 50	≤ 50
14	≤ 50	≤ 50
15	≤ 1000	≤ 1000
16	≤ 1000	≤ 1000
17	≤ 1000	≤ 1000
18	≤ 1000	≤ 1000
19	≤ 1000	≤ 1000
20	≤ 1000	≤ 1000

折纸(paper.cpp /2s /512M)

题目描述

一张折纸是一个 n 行 m 列的字符矩阵，每个位置上是一个大写英文字母。你每次可以选择相邻两行或者相邻两列之间的空隙，把小的半边沿着这条空隙折到另一边（如果两边一样大你可以选择任意一边折过去），前提是折过去的对应位置上的字母必须相同。

例如：

```
CAAA
ADAA
ADAA
CAAA
```

它可以以第三列和第四列之间的空隙作为对称轴，把第四列对折到左边；或者可以以第二行和第三行间的空隙，把下面对折到上面。但是其他位置都不能进行对折。

这个矩阵沿着第三列第四列对折可以得到：

```
CAA
ADA
ADA
CAA
```

继续沿着第二行第三行对折可以得到：

```
CAA
ADA
```

注意这里既可以往上对折也可以往下对折。

你要求出总共能够得到多少个不同的子矩形。注意**最终位置不同**就算不同，并不一定要字母不同才算不同。例如上面 4×4 的矩阵可以得到总共 6 种不同的子矩形（1 个 4×4 ，1 个 4×3 ，2 个 2×4 ，2 个 2×3 ）。

输入格式

第一行输入两个正整数 n, m 。

接下来 n 行每行一个长度为 m 的仅包含大写英文字母的字符串，表示矩阵。

输出格式

共一行一个数字，表示方案数。

样例输入 1

```
4 4
CAAA
ADAA
ADAA
CAAA
```

样例输出 1

```
6
```

数据范围

对于所有数据， $1 \leq n \times m \leq 10^6$ 。

测试点编号	n	m
1	$= 1$	≤ 100
2	$= 1$	≤ 1000
3	$= 1$	$\leq 10^6$
4	$= 1$	$\leq 10^6$
5	≤ 20	≤ 20
6	≤ 20	≤ 20
7	≤ 1000	≤ 1000
8	≤ 1000	≤ 1000
9	无特殊限制	无特殊限制
10	无特殊限制	无特殊限制

钻石(diamond.cpp / 2s /512M)

题目描述

有一张 n 个点 m 条边的简单无向图（即无重边无自环的无向图），你需要判断图中是否存在“钻石”。

一个钻石包含四个点 A, B, C, D ，其中恰好仅有 A, D 之间不存在边。

本题需要多测。

输入格式

第一行一个正整数 T ，表示数据组数。

对于每组数据，第一行两个正整数 n, m 表示点数和边数。

接下来 m 行每行两个正整数 u, v 表示一条边。

输出格式

每组数据输出一行 YES 或 NO，分别表示存在和不存在。

样例输入 1

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

样例输出 1

```
YES
NO
```

数据范围

对于所有数据， $\sum n \leq 10000, \sum m \leq 5 \times 10^5$ 。

测试点编号	\sum	\sum
1	≤ 50	无特殊限制
2	≤ 50	无特殊限制

测试点编号	$\sum n$	$\sum m$
3	≤ 400	无特殊限制
4	≤ 400	无特殊限制
5	≤ 3000	≤ 10000
6	≤ 3000	≤ 10000
7	≤ 10000	无特殊限制
8	≤ 10000	无特殊限制
9	≤ 10000	无特殊限制
10	≤ 10000	无特殊限制