

NOIP 模拟赛

题目名称	镜子	匹配	修行	数数
英文名称	mirror	match	asceticism	count
输入文件名	mirror.in	match.in	asceticism.in	count.in
输出文件名	mirror.out	match.out	asceticism.out	count.out
程序名称	mirror	match	asceticism	count
时间限制	1s	1s	0.6s	2s
空间限制	256MB	512MB	256MB	256MB
子任务数量	10	3	4	4
子任务是否等分	是	否	否	否

编译选项：

C++	-o %s %s.* -Wl,--stack=0x40000000 -O2
-----	---------------------------------------

注意事项：

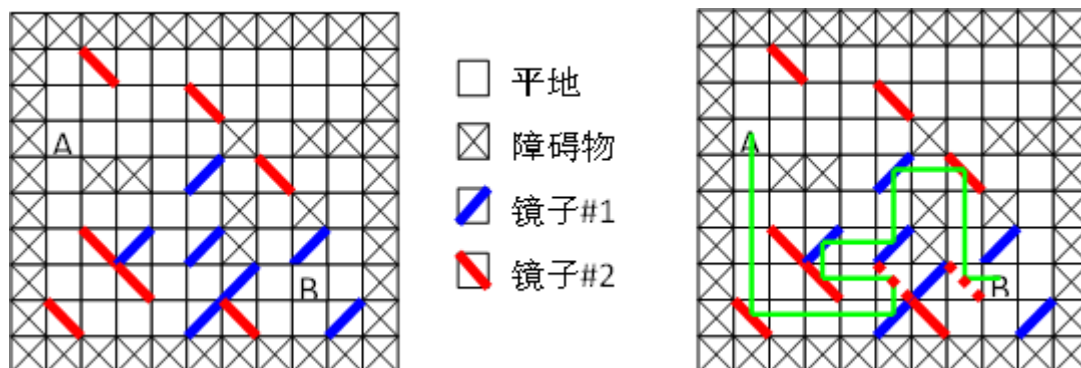
1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C++ 中函数 main() 的返回值类型必须是 int，程序正常结束时的返回值必须是 0。
3. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
4. 评测时采用的机器配置为：Intel(R) Core(TM) i5-5300U CPU @ 2.30GHz，内存 8GB。上述时限以此配置为准。
5. 选手应将各题的源程序放在选手文件夹内，**不要建立子文件夹**。
6. 评测使用 Windows 系统，系统为 64 位。

镜子 (mirror)

题目描述

两年前，King 制作了一个名叫 Mirror 的游戏，受到了广大玩家喜爱。游戏如下：有 $N \times M$ 的格栅，每个格子要么是平地，要么是障碍物（最外围的一圈肯定为障碍物）。光线能够水平（正东正西方向）或竖直（正北正南方向）的在平地上行进，但是如果射在障碍物上就会引起障碍物剧烈反应，从而导致爆炸。某些平地上已经事先安放上了若干面镜子，每面镜子有两种方向，若光线射在镜子#1上，则方向发生如下变化：向南 \leftrightarrow 向西；向北 \leftrightarrow 向东；若光线射在镜子#2上，则方向发生如下变化：向南 \leftrightarrow 向东；向北 \leftrightarrow 向西。现在有一个战士拿着激光枪站在 A 格上，有一个魔鬼站在 B 格上（ A 、 B 格都是平地，并且 $A \neq B$ ），请你帮助战士消灭魔鬼。你可以先在某些平地上放一些镜子，然后告诉战士往哪个方向开枪，注意：任意两面镜子（包括事先放好的和你新添加的）都不能放在同一格上，而且镜子是要钱的，所以你应该使镜子越少越好。这个游戏保证有解，为了确保自身安全，战士不能让任何一个障碍物发生爆炸。

看下面的范例：



只需要新添加两面镜子就足够了。

输入格式

第一行： N, M （用一个空格隔开， $4 \leq N, M \leq 1000$ ）

以下 N 行，每行 M 格字符（无多于空格）。空格表示平地；大写字母 X 表示障碍物；斜杠 $/$ 表示镜子#1；斜杠 \backslash 表示镜子#2； A 、 B 分别表示战士和魔鬼的位置。

输出格式

仅一个数，表示最少需要添加多少面镜子。

样例 1

输入

```

10 11
X X X X X X X X X X
X  \                X
X      \            X
X A      X  X X X
X  X X  /  \      X
X      X  X  X
X  \ /  / X  /  X
X      \  /  B  X
X \      / \      / X
X X X X X X X X X X

```

输出

2

数据范围与提示

- 对于 10% 的数据， $1 \leq N, M \leq 30$
- 对于 20% 的数据， $1 \leq N, M \leq 200$
- 对于另外 20% 的数据， $N = 1000, M \leq 25$
- 对于 100% 的数据， $1 \leq N, M \leq 1000$ ，保证数据有解。

匹配 (match)

题目描述

Farmer John 有 N ($1 \leq N \leq 3000$) 头各种大小的奶牛。他原本为每头奶牛量身定制了牛棚，但现在某些奶牛长大了，使得原先的牛棚大小不够用。具体地说，FJ 原来建造了 N 个牛棚的大小为 t_1, t_2, \dots, t_N ，现在奶牛的大小为 s_1, s_2, \dots, s_N ($1 \leq s_i, t_i \leq 10^9$)。

每天晚上，奶牛们都会按照某种方式寻找睡觉的牛棚。奶牛 i 可以睡在牛棚 j 中当且仅当她的大小可以进入牛棚 ($s_i \leq t_j$)。每个牛棚中至多可以睡一头奶牛。

我们称奶牛与牛棚的一个匹配是极大的，当且仅当每头奶牛可以进入分配给她的牛棚，且对于每头未被分配牛棚的奶牛无法进入任何未分配的空牛棚。

计算极大的匹配的数量模 $10^9 + 7$ 的结果。

输入格式

输入的第一行包含 N 。

第二行包含 N 个空格分隔的整数 s_1, s_2, \dots, s_N 。

第三行包含 N 个空格分隔的整数 t_1, t_2, \dots, t_N 。

输出格式

输出极大的匹配的数量模 $10^9 + 7$ 的结果。

样例

输入

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

输出

```
9
```

以下是全部九种极大的匹配。有序对 (i, j) 表示奶牛 i 被分配到了牛棚 j 。

(1, 1), (2, 2), (3, 4)
(1, 1), (2, 3), (3, 4)
(1, 1), (2, 4)
(1, 2), (2, 3), (3, 4)
(1, 2), (2, 4)
(1, 3), (2, 2), (3, 4)
(1, 3), (2, 4)

(1, 4), (2, 2)
(1, 4), (2, 3)

数据范围与提示

子任务编号	分数	特殊限制
1	15	$N \leq 8$
2	25	$N \leq 50$
3	60	无特殊限制

修行 (asceticism)

题目描述

一天，JOI 君得到了一台时间机器。他决定回到九世纪的日本。他遇见了当时日本最伟大的僧人之一——空海法师。这位法师想要创造一种新的修行方式。

他的修行方式如下：

- 空海法师要读一本有 N 句话的佛经，这些句子是有顺序的，他必须要按顺序读；
- 每句话都标有一个从 1 到 N 的正整数，没有两个不同的句子标有相同数字；
- 一天被平均分为 N 个时段，他只能在某一天的第 i 个时段读编号为 i 的句子。保证他能在第 i 个时段读完编号为 i 的句子。

空海法师想要尽快读完整部佛经。然而，读完佛经花费的天数取决于佛经有多少句话。空海法师让 JOI 君计算一下，如果他采用最佳方案，用恰好 K 天读完佛经的方案数是多少。

任务

给出文章中句子数 N 和天数 K ，计算空海法师用恰好 K 天读完佛经的方案数，对 $10^9 + 7$ 取模。

输入格式

从标准输入读入下列数据：

- 第一行包含两个正整数 N 和 K ，用一个空格隔开。

输出格式

输出空海法师用恰好 K 天读完佛经的方案数，对 $10^9 + 7$ 取模。

样例 1

输入

```
3 2
```

输出

```
4
```

有 4 种可能的编号方式：

- 第一个句子编号为 1，下一个句子编号为 3，最后一个句子编号为 2。他在第一天读前两句话（编号分别为 1, 3），在第二天读最后一句话（编号为 2）。
- 三个句子分别编号为 2, 1, 3；
- 三个句子分别编号为 2, 3, 1；
- 三个句子分别编号为 3, 1, 2。

样例 2

输入

10 5

输出

1310354

数据范围与提示

所有数据满足 $1 \leq N \leq 10^5, 1 \leq K \leq N$ 。

详细子任务的附加限制及分数如下：

Subtask #	附加限制	分数
1	$N \leq 10$	4
2	$N \leq 300$	20
3	$N \leq 3 \times 10^3$	25
4	无附加限制	51

数数 (count)

题目描述

Bessie 有一个连通无向图 G 。 G 有 N 个编号为 $1 \dots N$ 的结点，以及 M 条边 ($1 \leq N \leq 10^2, N-1 \leq M \leq \frac{N^2+N}{2}$)。 G 有可能包含自环 (一个结点连到自身的边)，但不包含重边 (连接同一对结点的多条边)。

令 $f_G(a, b)$ 为一个布尔函数，对于每一个 $1 \leq a \leq N$ 和 $0 \leq b$ ，如果存在一条从结点 1 到结点 a 的路径恰好经过了 b 条边，则函数值为真，否则为假。如果一条边被经过了多次，则这条边会被计算相应的次数。

Elsie 想要复制 Bessie。具体地说，她想要构造一个无向图 G' ，使得对于所有的 a 和 b ，均有 $f_{G'}(a, b) = f_G(a, b)$ 。

你的工作是计算 Elsie 可以构造的图 G' 的数量，对 $10^9 + 7$ 取模。与 G 一样， G' 可以包含自环而不能包含重边 (这意味着对于 N 个有标号结点共有 $2^{\frac{N^2+N}{2}}$ 个不同的图)。

每个输入包含 T ($1 \leq T \leq \frac{10^5}{4}$) 组独立的测试用例。保证所有测试用例中的 N^2 之和不超过 10^5 。

输入格式

输入的第一行包含 T ，为测试用例的数量。

每个测试用例的第一行包含整数 N 和 M 。

每个测试用例的以下 M 行每行包含两个整数 x 和 y ($1 \leq x \leq y \leq N$)，表示 G 中存在一条连接 x 与 y 的边。

为提高可读性，相邻的测试用例之间用一个空行隔开。

输出格式

对每个测试用例，输出一行，为不同的 G' 的数量，对 $10^9 + 7$ 取模。

样例 1

输入

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

输出

```
3
```

在第一个测试用例中， G' 可以等于 G ，或以下两个图之一：


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

样例 2

输入

```
7

4 6
1 2
2 3
3 4
1 3
2 4
1 4

5 5
1 2
2 3
3 4
4 5
1 5

5 7
1 2
1 3
1 5
2 4
3 3
3 4
4 5

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

6 7
```

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

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

22 28
1 2
2 3
3 4
4 5
5 6
6 7
1 7
1 8
3 9
8 10
10 11
10 12
10 13
10 14
11 15
12 16
13 17
14 18
9 15
9 16
9 17
9 18
15 19
19 20
15 20
16 21
21 22
16 22

输出

45
35
11
1
15
371842544
256838540

有一些较大的测试用例。确保你的答案对 $10^9 + 7$ 取模。注意倒数第二个测试用例的答案为 $2^{45} \pmod{10^9 + 7}$ 。

数据范围与提示

子任务编号	分数	特殊限制
1	15	$N \leq 5$
2	10	$M = N - 1$
3	30	\mathcal{E}
4	45	无特殊限制

\mathcal{E} : 如果并非对于所有的 b 均有 $f_G(x, b) = f_G(y, b)$, 则存在 b 使得 $f_G(x, b)$ 为真且 $f_G(y, b)$ 为假。