

图图的数组

Description

图图给了你一个长度为 n 的整数数组 a_1, a_2, \dots, a_n ，你每次可以从中任选两个数 a_i, a_j 删除，然后将他们的差 $|a_i - a_j|$ 加入数组，如此进行 $n-1$ 轮后数组中只剩下一个数字。

问是否存在一种方案，使得最终剩下的数字是一个偶数。如果存在则输出 Yes，否则输出 No。

Input

输入数据的第一行为数据组数 T ，表示接下来有 T 组数据。

每组数据的第一行为一个正整数 n ，表示初始数组长度。

第二行包含 n 个整数，代表初始数组中的数字。

数据保证 $2 \leq n \leq 10^5$ ， $-10^9 \leq a_i \leq 10^9$ 。

Output

对于每组数据第一行输出一行字符串，"Yes" 或 "No"，无引号。

如果输出 Yes，你需要给出该方案，接下来输出 $n-1$ 行，每行两个数字，表示每轮删除的数。

Sample Input and Sample Output

Sample Input	Sample Output
2	No
5	Yes
1 2 3 4 5	2 4
5	6 8
2 4 6 8 10	10 2
	2 8

La La String

Description

涛涛最近对一种名为“La La String”的字符串特别感兴趣。“La La String”是一个字符串，它的定义如下：

- 1)该字符串里只包含小写字母。
- 2)已知一个优雅的小写字母集合 T（集合内有 m 个互不相同的小写字母），”La La String”只出现 T 集合里的任意小写字母，并且每个字母最多出现两次。
- 3)该字符串是一个回文串。“回文串”是一个正读和反读都一样的字符串，比如“level”或者“noon”等就是回文串。

例如 $T=\{ 'a', 'b', 'c' \}$ ，那么“abcba”，“cabbac”，“aba”是符合要求的“La La String”，“aabbcc”，“abcccb”，“ABA”都不是符合要求的“La La String”。

涛涛向你求助，给定集合 T，一个字符串 S，你能帮他算出 S 有多少个子串（字符串中任意个连续的字符组成的非空子序列称为该串的子串）是“La La String”吗？

Input

第一行为数据组数 $n(n \leq 10)$ ，表示接下来有 n 组数据。

每组数据的第一行为一个数字 $m(1 \leq m \leq 26)$ ，表示集合 T 的大小。

第二行为 m 个小写字母，表示该集合 T。两个字母间用一个空格隔开。

第三行为一个包含小写和大写字母的字符串 $S(1 \leq |S| \leq 50000)$ 。

Output

对于每组数据输出一行，输出格式为"Case #X: Y"，其中 X 为第几组数据，Y 表示字符串 S 有多少子串是“La La String”。

Sample Input and Sample Output

Sample Input	Sample Output
2	Case #1: 0
1	Case #2: 22
b	
AAA	
3	
a b c	
abcabbccabbac	

可持久化动态树

Description

给出一个 $n*n$ 的网格， (i,j) 表示第 i 行的第 j 个格子。有一些格子中有障碍物(用'#'表示)。定义一个半径为 d ，中心为 (x,y) 的格点正方形为左上角为 $(x-d,y-d)$ 右下角为 $(x+d,y+d)$ 的正方形（该正方形边长为 $2d+1$ ）。格点正方形能在网格中上下左右移动，不过在移动过程中需要保证没有任何一个障碍物在任何一个时刻被格点正方形覆盖，格点正方形的边界也不能超出网格的边界。现在有 q 组询问，每次询问能将中心从 (a,b) 移动到 (c,d) 的半径最长的格点正方形的边长。

Input

输入包含多组数据，以文件末为结束标志
每组数据的第一行包含一个正整数 n ($n \leq 1000$)，表示网格大小。
接下来 n 行，每行包含一个长度为 n 的字符串，若第 i 行的第 j 个字符为'#'则表示 (i,j) 有障碍物，若字符为'.'则表示 (i,j) 中无障碍物。
接下去一行一个正整数 q ，表示询问次数。
接下去 q 行 ($q \leq 300000$)，每行四个正整数， a,b,c,d ，表示一次从 (a,b) 走到 (c,d) 的询问。如果无法到达则输出 0。
数据保证询问的起始点和目标点不会重合，且起始点和目标点中没有障碍物。
共有 13 组极限数据。

Output

对于每组数据输出 q 行，每行一个数字表示答案。

Sample Input and Sample Output

Sample Input	Sample Output
6	1
.....#	1
...#.#	1
....#.	0
....##	1
....#.	
#.....	
4	
2 2 6 3	
2 5 5 2	
2 2 6 6	
2 5 3 6	
2	

.. .. 1 1 1 2 2	
--------------------------	--

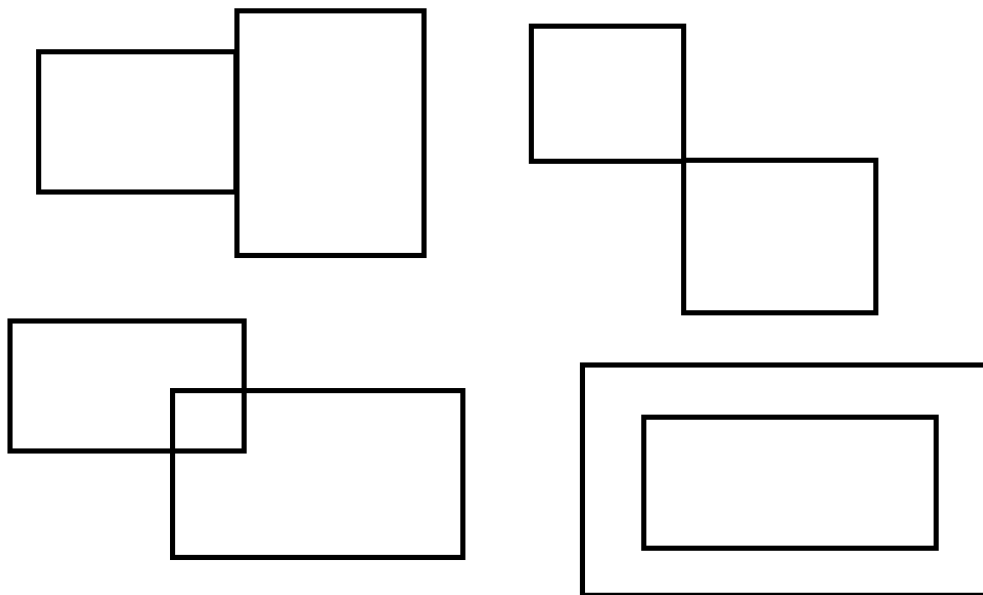
Hint

由于本题读入的数据量太大，为了避免读入数据的时间太长导致程序超时。请使用读入优化代替 scanf。

拆迁队

Description

S 城有 N 个小区，把 S 城的地图看成一个二维直角坐标系，每个小区都是一个平行坐标轴的矩形。每个小区由 $w \times h$ 个 1×1 的房子构成 (w, h 为矩形的长和宽)。第 i 个城市的左下角坐标为 (a_i, b_i) , 右上角坐标为 (c_i, d_i) 。显然的，小区与小区不会有相交和包含关系。但是两个小区可能有重叠的边或顶点。例如下图，上面两种情况是合法的，下面两种情况是非法的。



涛涛被开发商聘任为拆迁顾问。开发商有 Q 个询问，每次询问给出一个平行坐标轴的矩形，左下角坐标 (x_1, y_1) ，右上角坐标 (x_2, y_2) ，开发商想知道，如果征地范围为该矩形，开发商需要拆迁多少房子？除了这个 N 个小区覆盖的范围，你可以把其他区域都看成空地。现在，请你帮涛涛回答这 Q 个询问。

Input

第一行为数据组数 $T(T \leq 5)$ ，表示接下来有 T 组数据。

每组数据的第一行为两个数字 $N, Q(N, Q \leq 100000)$ ，表示小区数量和询问数量。

接下来 N 行，每行四个整数 a_i, b_i, c_i, d_i ，表示第 i 个小区的左下角 (a_i, b_i) ，右上角 (c_i, d_i) 。

接下来 Q 行，每行四个整数 x_1, y_1, x_2, y_2 ，表示询问拆迁范围为一个矩形，左下角为 (x_1, y_1) ，右上角为 (x_2, y_2) 。

输入中的坐标都是非负整数，范围为 $[0, 1000000]$ 。数据保证 $a_i < c_i, b_i < d_i, x_1 < x_2, y_1 < y_2$ 。

Output

对于每组数据，第一行输出 "Case #X:"，其中 X 为第几组数据。

接下来输出 Q 行，每行输出一个整数，表示该询问需要拆迁的房子。

Sample Input and Sample Output

Sample Input	Sample Output
2	Case #1:
3 3	1
1 0 3 2	4
6 0 8 2	12
3 2 5 4	Case #2:
4 3 6 5	2
1 0 3 2	
0 0 9 6	
2 1	
1 1 3 3	
3 3 5 5	
2 2 4 4	

Hint

由于本题读入的数据量太大，为了避免读入数据的时间太长导致程序超时。请使用读入优化代替 `scanf`。读入优化的参考代码如下：

```
inline int read()
{
    char ch=getchar();
    while (!(ch>='0'&&ch<='9')) ch=getchar();
    int re=0;
    while (ch>='0'&&ch<='9')
    {
        re=re*10+ch-'0';
        ch=getchar();
    }
    return re;
}
```

你可以把这段代码放入你的程序中，并直接调用该函数，例如：

```
T=read();
n=read();
```

走棋子

Description

涛涛在一个 N 行 N ($N \geq 2$) 列的棋盘上走棋子. 棋盘的每个格子上标有一个整数, 涛涛选择第 1 行的一个格子作为棋子起点. 棋子的移动规则如下:

1. 当棋子在第 i 行第 1 列时, 棋子可以被移动到第 $i+1$ 行的第 1 列或第 2 列;
2. 当棋子在第 i 行第 N 列时, 棋子可以被移动到第 $i+1$ 行的第 $N-1$ 列或第 N 列;
3. 当棋子在第 i 行第 j ($1 < j < N$) 列时, 棋子可以被移动到第 $i+1$ 行的第 $j-1$, j 或 $j+1$ 列.

涛涛的幸运数字是 3, 所以他想知道有多少种方案可以将棋子从第 1 行移动到第 N 行, 使得棋子走过的这 N 个格子上标的数字之和是 3 的倍数.

Input

输入有多组数据. 每组数据的第一行一个整数 N ($2 \leq N \leq 100$). 接下来的 N 行, 每行有 N 个整数, 每个整数在 1 和 100 之间, 代表棋盘对应位置上的整数. $N=0$ 表示输入结束.

Output

对于每组数据, 输出方案数除以 1007 的余数.

Sample Input and Sample Output

Sample Input	Sample Output
2	1
1 2	17
3 4	
3	
3 3 3	
3 3 3	
3 3 3	
0	

没玩过炉石还请跳过此题

Description

某大师沉迷于炉石这个游戏无法自拔，某天他开始思考新的策略来提升自己的实力，在此之前，我先简要介绍一下炉石传说这个游戏...

炉石传说是由暴雪娱乐开发的一款集换式卡牌游戏，故事背景基于魔兽争霸系列的世界观。首先游戏开始会随机在玩家的卡牌组(30 张)抽取 4 张（后手）或 3 张（先手）卡牌，然后玩家可以换掉其中若干张来完成开局的牌组。每个回合会自动增加 1 点魔力值，魔力值是用来召唤卡牌的必要单位，回合结束后会自动回复(在界面右下方的蓝色宝石就是魔力值)。当玩家召唤了一张仆从类卡牌后，此回合卡牌是无法进攻的，但是可以用来阻挡对手的进攻。卡牌除了仆从卡牌(用来进攻和防御的无回合限制的卡牌)外，还有技能卡牌，是消耗品。当玩家魔力值很多的时候，可以多次使用魔力值允许内的卡牌。不同的英雄技能不同，除了可拥有公共卡牌外，还可获得英雄专属卡牌...

没玩过的同学看完是不是有点晕了，没错，要的就是这种效果，其实以上都是我随手从百度百科上复制的，请自动忽略，不开心？好，现在我们来简化一下这个问题...

假如现在大师有 n 个随从，对手有 m 个随从，每个随从有一定攻击力和血量。现在轮到大师的回合，若不考虑出牌，在此回合他可以让每一个随从去攻击对手的随从，但每个随从至多只能攻击一次，也可以选择不攻击，而对手的随从一旦受到攻击便会进入潜行状态（即每个随从最多只能被攻击一次），假如随从 A 攻击随从 B，则他们的血量都将扣除对方相应的攻击力数值，当一个随从的血量小于或等于 0 则将被视为死亡，大师想找到一种策略，使得最后己方存活的随从数 X 减去对方存活的随从数 Y 尽可能大，请问 $X-Y$ 最大为多少？

Input

第一行包括两个数 n, m ($1 \leq n, m \leq 100000$)，如题所述。

接下来 n 行，第 i 行包括两个数 a_i, b_i ($1 \leq a_i, b_i \leq 10^9$)，表示己方第 i 个随从的攻击力和防御力。

接下来 m 行，第 i 行包括两个数 c_i, d_i ($1 \leq c_i, d_i \leq 10^9$)，表示对方第 i 个随从的攻击力和防御力。

Output

输出一行一个数 S ，表示 $X-Y$ 的最大值。

Sample Input and Sample Output

Sample Input	Sample Output
2 1 2 3 4 2 1 3	2
3 4	1

5 5	
4 4	
6 2	
2 6	
3 5	
1 4	
9 9	

Hint

样例一用我的 2 号随从打死对方的 1 号随从，最后己方两个随从都没死，所以为 $2-0=2$ 。

样例二用己方的 1 号随从打敌方的 2 号随从，己方的 2 号随从打对方的 3 号随从，己方的 3 号随从打对方的 1 号随从，最后我方剩余 2 个随从，对方剩 1 个随从，故 $2-1=1$ 。

洛神求期望

Description

众所周知，在三国杀里面甄姬是一个十分强大的英雄，尤其是她的那招“洛神”，爆发力十分惊人，同时再配合“倾国”，防御力也相当突出，可谓是一个全能英雄。在每回合开始，她可以进行摸牌判定阶段（洛神），每次摸一张牌，假如牌的颜色是黑色的话她又可以继续摸牌，直到摸到红色牌为止。那我们现在来考虑一个问题，假如已知牌库中有 N 张黑色牌和 M 张红色牌，那甄姬期望一回合通过洛神能摸到多少张牌呢？

虽然甄姬这个英雄很厉害（她只是出来搞笑的），然而一个非主流舍友偏偏要玩周泰这种冷门英雄，周泰只有一招技能叫“不屈”，每当生命值小于或等于 0 的时候，他可以从牌堆亮出一张牌并置于你面前。若此牌的点数与你面前已有的任何一张牌都不同，你不会死亡。若出现相同点数的牌，你将被视为死亡。那么问题来了，杀马特舍友想知道：若牌库的牌只考虑让“不屈”摸牌，已知牌库中有 n 种不同数字的牌以及每种数字的牌的数量，周泰的“不屈”技能期望摸到第几张牌失效呢？请问你能帮他解决这个问题吗？

Input

第一行一个数 n ($1 \leq n \leq 100000$)，如题所述。

接下来一行包括 n 个数，第 i 个数 a_i ($1 \leq a_i \leq 100$)，表示第 i 种牌有多少张。保证至少有一个 a_i 大于 1。

Output

输出一行一个数 S ，表示期望第几回合死亡，设最终答案为 A/B ，则输出 $A * (B^{-1})$ ，其中 B^{-1} 表示 B 在模 P 下的逆元，即满足 $B * B^{-1} = 1 \pmod{P}$ ，最终答案对 985661441 取模。

Sample Input and Sample Output

Sample Input	Sample Output
2 1 2	657107630
3 2 2 2	788529156

Hint

样例一算出的值应该为 $(3+2+3)/3=8/3$ 。

聚集度

Description

在一个社群中，每个人各自认识一些人。我们定义的“认识”关系是相互的，即如果甲认识乙，则乙也认识甲；反之亦然。一个人的 1-聚集度定义为他认识的人数，2-聚集度定义为他所认识人的 1-聚集度之和。一个社群的聚集度定义为社群中所有人的 2-聚集度之和。对于一个由 N 个人（编号从 1 到 N ）组成的社群，你的任务是求出社群的聚集度。

Input

输入有多组数据。每组数据的第一行包含两个整数 N ($1 \leq N \leq 20000$) 和 M ($0 \leq M \leq 100000$)，分别表示社群中的人数和认识关系的数量。接下来的 M 行，每行有两个不同的整数 u 和 v ($1 \leq u, v \leq N$)，表示编号为 u 和 v 的两个人相互认识。一对认识关系只在一组数据的输入中给出一次。 $N=M=0$ 表示输入结束。

Output

对于每组数据，输出社群的聚集度。

Sample Input and Sample Output

Sample Input	Sample Output
3 2 1 2 2 3 3 3 1 2 2 3 1 3 0 0	6 12