Problem A. 教科书般的亵渎

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

"现在给大家展示一波教科书般的亵渎""扭了扭了"

炉石传说是一款比较火的卡牌游戏，不同的卡牌间可以打出许多不可思议的配合。

炉石传说的战斗大部分由随从来完成。每个随从都有血量。当随从血量为 0 时，随从就会死亡。

"恐怖的奴隶主" 是一个有 3 血的随从。他的特效是：**每当该随从收到伤害且没有死亡时，召唤一个恐怖的奴隶主**。也就是说，一个 3 血或 2 血的奴隶主，受到 1 点伤害后，都会召唤一个新的 3 血的奴隶主。而一个 1 血的奴隶主如果受到了 1 点伤害，就会直接死亡而不会召唤奴隶主。

"亵渎" 是一张法术牌。其效果是：**对全场随从造成 1 点伤害，如果有随从被亵渎的伤害打死，就再次施放亵渎**。如果场上有一个 1 血一个 2 血的随从，亵渎会对全场打 1，杀死那个 1 血随从，再全场打 1，杀死那个 2 血的随从，再对全场打 1（虽然此时场上已经没有随从了）

可以看出，奴隶主和亵渎的效果都是递归定义的。两张牌只要配合起来，就可能造成无限的递归。炉石的程序员肯定不希望游戏发生这种情况，他们偷偷给亵渎加了一个使用次数限制，亵渎在自我递归一定次数后就会停止了。

现在，假设你是一个炉石主播。你的场上有 a_1 个 1 血的奴隶主，a_2 个 2 血的奴隶主，a_3 个 3 血的奴隶主，亵渎在释放 n 次后就会停止递归。游戏中随从的数量不做限制，可以同时有无限个随从出现在游戏里。现在你释放了一次亵渎，游戏陷入了漫长的结算过程。这个时候，不断地有弹幕在问你："主播主播，一共死了多少个奴隶主啊？"你决定用你高超的编程技术来回答这个问题。

Input

输入共一行。这一行包括四个数 $a_1, a_2, a_3, n (0 \le a_1, a_2, a_3 \le 10^9, 2 \le n \le 10^{18})$ 分别表示 1 血奴隶主的个数、2 血奴隶主的个数、3 血奴隶主的个数、亵渎的最大释放次数。

Output

输出共一行，包含一个整数（对 $10^9 + 7$ 取模），表示死掉奴隶主的个数。

Examples

standard input	standard output
1 1 1 4	5
1 0 1 5	1

Problem B. 北理工的恶龙

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

北理工的恶龙其实并没有被完全杀死，但是恶龙的身体却发生了很大的变化。恶龙仍然长有 n 个头，但经过了勇士的调教，恶龙变得善良了一些。它的 n 个头每个头要么仍是邪恶的头，用 “1” 表示，要么已经变得善良，用 “0” 表示，因而恶龙的 n 个头就可以用 n 位 01 串来表示。你作为被选召的勇士，目标是把所有的龙头都变成 $0000\cdots00$ 完全善良的龙头。每一次，你可以砍掉龙最右侧的一个头，同时龙会在最左侧长出新的一个头，以保证龙头数量不变。如果你砍掉的是一个 1，即邪恶的头，你可以决定龙在最左侧会长出什么样的头；但如果你砍掉了一个善良的头，那么玻璃心的龙龙将会在左侧不受控制的长出一个随机的头，既可能是善良的头，也可能是邪恶的头，而且对你错杀无辜的它总会与你作对阻挠你，尽力的破坏你的计划。

注意，这个问题可能没有你想的那么简单。显然，你必须把一些 1 变成 0，这样才能让 1 的数量减少并消失。但是如果只是简单的每次把 1 变成 0，最终不见得能取胜。比如，如果龙头的状态是 101，那么去掉最右边的 1 并选择在左边长出一个 0，则龙头会变成 010；再把 010 右边的 0 去掉后，如果左边仍长出一个 1，则龙头又变回了 101 的状态，如此反复，将永远不能得到 000。

现在给你一个恶龙头的初始状态，即一个 01 串，请判断能否在有限步之内让全部的龙头都变成善良的龙头。

Input

输入第一行 $T, T \le 1000$，表示用例组数。

之后 T 行，每行一个 01 串 S 表示龙头的初始状态，“0” 表示善良的头，“1” 表示邪恶的头。$|S| < 100$

Output

能否将全部的龙头变成善良的头，可以的话输出 “kind longlong”，不可以则输出 “evil longlong”（不含引号）。

Example

standard input	standard output
1 1111	kind longlong

Problem C. 特殊生成树

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

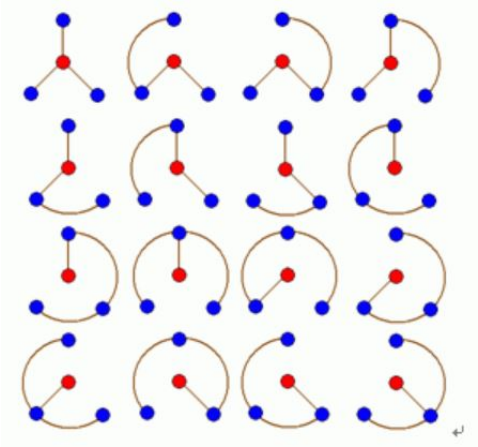
将 N 个点排列成一个圆形，中间放置一个点固定为根节点，问特殊生成树的种类数。

特殊生成树：除根节点以外，其他节点只能与自己左右节点相连，或与根节点相连。

p.s. 若节点的左右节点为同一个节点，向左或向右连接视为不同的生成树。

由于种类数可能过大，对 $1,000,000,007$ 取模。

当 $N = 3$ 时，所有生成树表示如下：



Input

多组数据，请输入要文件结束。

每组数据包含一个整数 $N(0 < N < 10^9)$。

Output

对于每组数据，输出一个整数，表示特殊生成树的种类数，答案对 $1,000,000,007$ 取模。

Example

standard input	standard output
2 3	5 16

Problem D. 某亮的零幺魔法

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

某天，某亮在地上捡到一张纸。

这张纸除了一串互异的正整数啥都没有。某亮准备锻炼一下他刚刚学会的零幺魔法。

某亮可以选择一个数 k，将小于 k 的数变成 1，将不小于 k 的数变为 0，这个魔法耗费的魔力值为 k。

当然某亮施展完魔法后也能解除魔法。(意思就是你可以慢慢试)

某亮一直信奉既患寡又患不均，所以他施展完零幺魔法后，希望 1 的块数最多（连续的 1 视为一块），同时希望每一块当中 1 的数目相同。(例如，在样例中，如果选择 $k = 2$，变为 10000000，连续的 1 只有一块，选择 $k = 7$, 变为 11011011，连续的 1 有三块，且每块都是 2 个 1)。当然某亮也希望在达到目的的情况下尽量减少魔力值 k 的消耗。

ball ball 您帮帮智商很低的某亮找到这个需要的魔力值 k。即寻找使得在每块 1 的个数相同时块数最多的最小的 k 值。

Input

第一行一个数 n，代表数字个数，$n \le 2 \times 10^5$。

下面 n 行，每行一个数 a_i，每个数都不同,$1 \le a_i \le 10^9$，表示这个正整数序列。

Output

满足情况的最小魔力值 k。

Example

standard input	standard output
8 1 2 7 3 4 8 5 6	7

Problem E. 懋懋的血流成河

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

可爱的懋懋是一名狂热的麻将爱好者，由于他有一个四川女朋友，所以他尽心尽力学习四川麻将中血流成河的打法。

麻将是我国国粹，有很多种变种。在四川成都，人们通常打一种叫做 “成麻” 的麻将。成麻包含 “万”、“筒”、“条”3 种花色，每种花色从一到九 9 张牌，每种牌 4 张，所以一套成麻共有 108 张牌，称水浒 108 将（我胡说的）。每种花色的叫法就用数字加上花色，例如筒的第 5 张就叫做 “五筒”，万的第九张就叫做 “九万”。我们用分别用字母 m,p,t 表示万筒条，在字母前面加上数字表示大小，例如 “3t” 表示 “三条”，“八万” 用 “8m” 表示。

通常，一局麻将需要四个玩家才能开始游戏，每人最初有 13 张牌，游戏进行时，每人从剩余牌墙中摸一张牌，再打出一张牌。麻将胜利的条件是和牌，和牌是指玩家获得一张牌时，TA 的牌形成某种番型即胜利。下面介绍成麻的和牌规则：

- 缺一门：成麻的特殊规则，即玩家想要和牌就不能同时有三种花色的牌，此规则适用于所有和牌情况。例如玩家手上同时有万筒条，即使此时玩家手上的牌看似符合某种和牌规则，玩家也不能和牌。如果没有满足缺一门的条件，就叫做花猪。
- 平胡：我们称三个一样的牌，或者同花色连续的三张牌为一个 “搭子”。例如 7m7m7m 就是一个搭子，3t4t5t 也是一个搭子，但是 3m4m5p 就不是一个搭子。我们称一对相同的牌为 “将牌”，例如 2t2t 就是一对将牌。平胡的达成条件是 4 个搭子和 1 对将牌。例如，1m2m3m4m4m4m7m7m2p3p4p7p8p9p 就是平胡牌，注意 2m3m4m3p4p5p4p5p6p1t2t3t4t4t 不是平胡，这是因为它不满足缺一门的条件。
- 一条龙：在平胡基础上，如果某一花色从 1 到 9 都有的话，就叫一条龙。例如 1m2m3m4m5m5m6m6m7m7m8m9m2p2p 就是一条龙。
- 对对胡：在平胡的基础上，如果所有的搭子都不是顺子，那么就是对对胡。例如 1m1m1m2m2m2m4p4p4p8p8p8p5p5p 就是对对胡。
- 暗七对：7 对将牌达成暗七对胡法，例如 4m4m5m5m6m6m2p2p3p3p6p6p9p9p 就构成暗七对。龙七对：7 对将牌中至少有 2 对将牌一样，例如：1m1m1m1m3m3m7m7m5p5p8p8p9p9p 就是龙七对。
- 清一色：在和牌基础上，如果所有牌都是一个花色的，那么称之为清一色。清一色是一个叠加番型，一般不和其他番型冲突。在清一色的作用下，各番型有新的称呼：平胡 + 清一色就叫 “清一色”，一条龙 + 清一色 = 清一色一条龙，对对胡 + 清一色 = 清一色对对胡，暗七对 + 清一色 = 清七对，龙七对 + 清一色 = 清龙七对。

每一种番型有其倍数，下面给出倍数表：

```
| 番型                        | 倍数 |
| 平胡PingHu                  | 1    |
| 对对胡DuiDuiHu              | 2    |
| 一条龙YiTiaoLong            | 2    |
| 暗七对AnQiDui               | 4    |
```

```
| 清一色QingYiSe                  | 4    |
| 清一色对对胡QingYiSeDuiDuiHu    | 8    |
| 龙七对LongQiDui                 | 8    |
| 清一色一条龙QingYiSeYiTiaoLong  | 8    |
| 清七对QingQiDui                 | 16   |
| 清龙七对QingLongQiDui           | 32   |
```

除此之外，如果手里有 4 张一样的牌，就算做一根，在倍数上乘上 2，例如 3m4m5m5m6m6m6m6m7m8m8m8m9m9m，除了本身的清一色 4 倍之外，还要乘以 2 即 8 倍。需要注意的是龙七对或者清龙七对在算根的时候，需要减去 1 根，例如 1m1m1m1m3m3m5m5m7m7m9m9m9m9m 本身是清龙七对 32 倍，虽然有 1m 和 9m 两根但是减去一根，倍数只乘上 2 为 64 倍。

成麻中还有碰杠等其他很复杂的情况，不过懋懋还是新手，先学完这些再说吧。

某天，懋懋在和他女朋友打麻将，这个时候他女朋友突然摊牌说自己胡了。懋懋很慌，他不知道女朋友是否真的胡了（如果没有胡却说自己胡了，就叫做诈胡）。于是他来找你帮忙了，如果你成功解决了这个问题，你将获得加入 ACM 雀魂群的邀请。（ACM 雀魂群：qls 一手创建的巨佬云集的群，进去就能和 qls，xzm 等大神在赛 pai 场 zhuo 上一决高下）

Input

多组用例，请处理到文件结束。

对于每组用例，输入一个字符串，表明懋懋女朋友的牌，保证输入合法（即不会出现 5 张一样牌或者斗地主的牌）。数据组数不超过 10^5 组。

Output

如果懋懋女朋友诈胡了，输出"ZhaHu"，如果还是花猪，输出"HuaZhu"。

如果懋懋女朋友真胡了，请按照番数尽可能高的情况来考虑。如果是七对胡法，先按字典升序输出七对，如果是平胡胡法，先考虑将牌字典序最小的胡法，然后按字典升序输出所有搭子，再输出将牌，此后再输出番型和倍数。

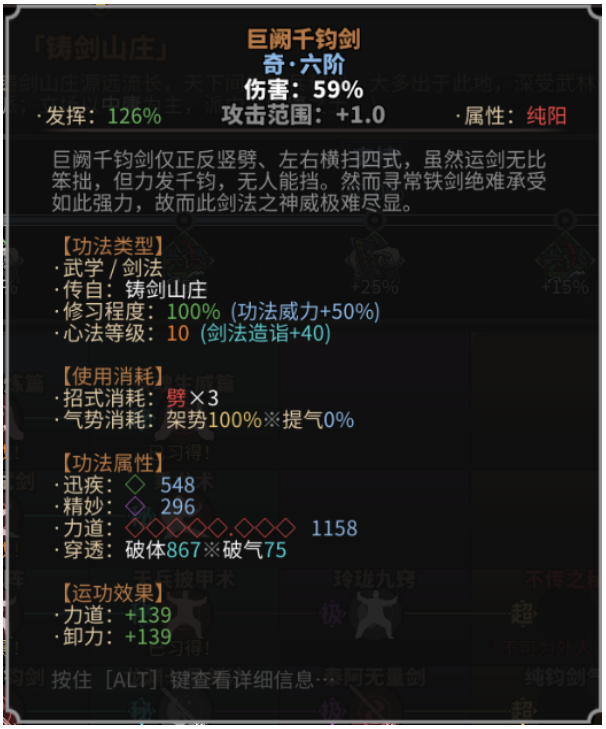
具体格式见样例。

Example

standard input	standard output
1m1m1m1m2m2m2m2m3m3m3m3m4m4m 3p4p5p5m6m6m6m6m7m8m8m8m9m9m 1m1m1m2m2m3m4m9p9p4t5t6t7t8t	1m1m 1m1m 2m2m 2m2m 3m3m 3m3m 4m4m QingLongQiDui *128 5m6m7m 6m6m6m 8m8m8m 3p4p5p 9m9m PingHu *2 ZhaHu HuaZhu

Problem F. 巨阙千钧剑

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes



巨阙千钧剑仅正反竖劈，左右横扫四式，虽然运剑无比笨拙，但力发千钧，无人能挡。然而寻常铁剑绝难承受如此强力，故此剑法之神威极难尽显。

——《太吾绘卷》

在太吾世界中，侠客们每挥动一次武器攻击敌人，就可以攒下一个卡片，称为“招式”，我们我们将招式简记为 1，2，3。

每种武器都会以六个一循环攒下招式，比如某种武器的招式顺序是 1，2，3，1，2，3.

而施放功法需要消耗“式”，比如铸剑山庄的六阶功法“巨阙千钧剑”需要消耗的“招式”是三个'1'，而另外一门九阶功法“十二路鱼肠刺剑”只需要消耗一个"2"就可以施放。

让我们简化一下太吾世界的战斗，假想两个侠客，分别有一定的生命值，分别拿着武器，它们也都分别会不同的一门或几门功法，侠客 A 先攒下一个“招式”，然后他可以选择消耗“招式”施放功法或者攒下“招式”，然后轮到 B 获得一个“招式”，B 再选择施放功法或者等待。

举个例子，A 的生命值是 20，武器的招式顺序是 1，2，3，1，2，3,，他会两门功法，一门功法消耗一个 2，伤害 15，一门功法消耗三个 1，伤害 100.

B 的生命值是 30，武器的招式顺序是 1，1，2，3，1，3，只会一门功法，消耗一个 3，伤害 15.

如果 A 足够聪明，胜负是显然的，第 2 个回合，A 消耗一个 2 打掉 B 的 15 血，第 4 个回合，B 消耗一个 3 打掉 A 的 15 血，第 5 个回合，A 再消耗一个 2 打掉 B 的 15 血，就把 B 打死了。

	A的的生命	A的招式	B的招式	B的生命
第一回合	20	1		30
	20	1	1	30
第二回合	20	1，2	1	30
A使用了功法	20	1	1	30-15=15
	20	1	1，1	15
第三回合	20	1，3	1，1	15
	20	1，3	1，1，2	15
第四回合	20	1，3，1	1，1，2	15
	20	1，3，1	1，1，2，3	15
B使用了功法	5	1，3，1	1，1，2	15
第五回合	5	1，3，1，2	1，1，2	15
A使用了功法	20-15=5	1，3，1	1，1，2	15-15=0
A胜利				

然而，如果 A 只会那个三个 “3” 的功法（就是巨阙千钧剑），要第七回合才能使出来，那么他还会在式还没攒完的第六回合被 B 拍死。

	A的的生命	A的招式	B的招式	B的生命
第一回合	20	1		30
	20	1	1	30
第二回合	20	1，2	1	30
	20	1，2	1，1	30
第三回合	20	1，2，3	1，1	30
	20	1，2，3	1，1，2	30
第四回合	20	1，2，3，1	1，1，2	30
	20	1，2，3，1	1，1，2，3	30
B使用了功法	20-15=5	1，2，3，1	1，1，2	30
第五回合	5	1，2，3，1，2	1，1，2	30
	5	1，2，3，1，2	1，1，2，3	30
B使用了功法	5-15=-10	1，2，3，1，2	1，1，2	30
B胜利				

所以说，祸福难料，生死难知，横练偏门功法的市井泼皮，乱拳打翻了神完气足的宗匠名师，这种事情也不是不曾发生过。

另外，为了使得评测机不会被卡爆，太吾世界的侠客们约定，三百回合后仍未分出胜负就算战平。

现在告诉你两个人的武器，功法，生命值，请你判断，如果两个人都用最优策略战斗的话，谁会获得胜利，或者双方战平，以及，何时战斗会结束。

Input

第一行输入先手的生命值

第二行输入六个数字，都为 1，2，3 中之一，代表先手的武器招式顺序，注意，这些招式是循环施放的。

第三行输入一个数字 n，代表先手会的功法数量。

之后的 n 行，每行输入 4 个数字，前三个分别代表此功法消耗招式 1，2，3 的数量，最后一个代表此功法的伤害。

（再强调一遍，招式没有伤害，功法才有伤害）

然后用同样的格式，输入后手的生命，武器，功法状态。

Output

在一行中输出两个数字，代表在两人都使用最优策略战斗的前提下，谁获胜（先手获胜输出 1，后手获胜输出 2，战平输出 0），以及在第几个回合战斗结束（战平则一定是 300）。

Examples

standard input	standard output
20 1 2 3 1 2 3 2 0 1 0 15 3 0 0 100 30 1 1 2 3 1 3 1 0 0 1 15	1 5
20 1 2 3 1 2 3 1 3 0 0 100 30 1 1 2 3 1 3 1 0 0 1 15	2 6

Note

两人的生命值均在 [1,300] 之间

每人会的功法的总数在 [1,5] 之间

功法消耗每种式的总数在 [0,5] 之间，但保证对三种式的消耗不会都为 0

功法的伤害均在 [10,300] 之间

Problem G. 监管运输

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

物流公司与某公司达成协议，要把一大批货物分 n 天从 A 地（编号 1）运输向 B 工厂（编号 m），由于物流公司的信誉不太好，甲方公司要求沿路需要设置人员监管，每条道路由相应的监管成本。

为了监管的方便，一般固定一条路线运输，但是有些时候由于地方施工，某些城市无法通行。

修改路线是一件十分麻烦的事情，需要重新安排沿路的监管工作，所以会消耗 k 的成本更换监管方案。

政府会发布公告，告知这些天的地方施工情况，方便进行运输道路规划。

希望你能够帮忙订一个 n 天的运输计划，使得总监管成本尽可能地小。

Input

第一行输入 $n, m, k, e (n \le 100, m \le 20, e \le 190)$ 分别代表运输计划 n 天，一共 m 个城市，更换监管安排的成本 k，以及 e 条道路连接各城市。

接下来 e 行，每行输入 u, v, w，表示一条连接 u, v 两城市的道路，该道路监管成本 w。

接下来一行输入一个数 $p (p \le 1000)$，表示政府公告数量。

接下来的 p 行，每行输入 x, a, b，表示城市 x 在 a 到 b 天会施工无法通行。(包括 a, b)

Output

一个整数，表示总的运输计划所需的最小监管成本。

若由于施工原因无法完成运输输出 “-1”（不含引号）。

Example

standard input	standard output
5 5 10 8 1 2 1 1 3 3 1 4 2 2 3 2 2 4 4 3 4 1 3 5 2 4 5 2 4 2 2 3 3 1 1 3 3 3 4 4 5	32

Explanation

样例解释:

前三天走 1-4-5

后两天走 1-3-5

总成本为 (2+2)*3+(3+2)*2+10=32

Problem H. 种瓜得瓜，种豆得豆

Input file:	`standard input`
Output file:	`standard output`
Time limit:	10 seconds
Memory limit:	256 megabytes

亮皇轻轻松松在大学四年间拿到了北理工 rk0，也有很多家公司给他了 offer，但是他看了看觉得薪水都太低了，并不想参加，于是决定回家种地。

他在家里开始种西瓜，家里一共有 n 块瓜田，每年可以产出 a_i 个西瓜。每年亮皇回家的时候都会看看他种的这块地，他会做以下下两种操作之一：

- 询问一段土地从第一年开始，到当年结束一共生产了多少个西瓜。
- 在一段土地上种上黄豆，这样可以让这块西瓜地从下一年开始产量 +1。

亮皇一共会回家 m 年，请你帮他回答所有的查询。

Input

输入数据包括多组数据（不超过 8 组），请处理到文件结束。

对于每一组数据，第一行包括一个整数 $n(1 \le n \le 10^5)$，表示西瓜地的数量。

第二行包括 n 个整数，表示初始年产量 $a_i(a_i \le 10^5)$。

第三行包括 m 个整数，表示亮皇要回家 $m(1 \le m \le 10^5)$ 年。

接下来 m 行，每行一定是以下两种形式之一：

- "Q s t"：查询第 $[s, t]$ 块土地上从开始种地到当年结束的总产量。
- "I s t"：当年在第 $[s, t]$ 块土地上种上黄豆。

数据保证至少有一次查询操作。

Output

对于每一组数据，在一行内回答所有的查询，用空格隔开。

Example

standard input	standard output
4 1 1 1 1 3 Q 1 3 Q 1 4 Q 2 4 4 1 2 3 4 3 Q 1 4 I 1 3 Q 2 4	3 8 9 10 29

Problem I. 懋懋的插线板

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1.5 seconds
Memory limit:	256 megabytes

在冬训的时间里，懋神除了在 CodeForces 上 AK 题以外，还安排了足够多的插线板，保证每个人都有充足的电量来 A 题，然而天有不测风云，机房被考试占用，懋神只好把插线板搬到新的教室去接线。

教室的形状是一个二维平面，一共有 N 个同学坐在教室中坐标为 (nx_i, ny_i) 的位置上，一共有 P 个墙插/地插在教室中坐标为 (px_i, py_i) 的位置上。

懋神一共带来了 X 个插线板，插线板电线的长度分别为 x_i。为了避免某品牌充电器线短的问题，懋神决定直接把插线板拉到每个同学身边。

插线板可以通过与墙插/地插直接连接，或是与其他插线板间接连接到墙插/地插，为所在位置的同学提供能量。插线板只能连接距离小于 x_i 的两个同学或同学与墙插/地插，不可以在不经过同学的情况下连接插线板延长电线长度。插线板的插头上有另一个插座，因此一个插线板上可以接任意个其他插线板，一个墙插/地插上也可以接任意个插线板。

懋神扫视了一下教室和带来的插线板，在 $O(-1)$ 的时间内找到了插线板的接法，并叫你去帮忙。然而，你并没有听清懋神需要你帮忙接线，还是帮忙再拿一箱插线板。懋神已经开始与缠在一起的电线奋斗，需要你自己判断一下懋神搬来的插线板够不够用了。

Input

第一行有三个数字，分别为 n, p, x，分别代表同学个数，墙插/地插个数与插线板个数。$(1 \le n \le 1000, 1 \le p \le 1000, 1 \le x \le 2000)$

接下来 n 行，每行有两个用空格隔开的数字 (nx_i, ny_i) 表示第 i 个同学坐的位置。$(0 \le nx_i, ny_i \le 10^6)$

接下来 p 行，每行有两个用空格隔开的数字 (px_i, py_i) 表示第 i 个墙插/地插所在的位置。$(0 \le px_i, py_i \le 10^6)$

接下来一行，有 x 个用空格隔开的数字 x_i ，表示第 i 个插线板电线的长度。$(1 \le x_i \le 10^6)$

Output

输出一行表示上述规则下插线板是否够用，"YES" 表示插线板够用，"NO" 表示插线板不够用（不含引号，不区分大小写）

Examples

standard input	standard output
3 1 3 1 1 1 9 7 1 4 5 5 5 6	YES
2 1 2 3 1 5 1 1 1 2 1	NO
2 2 2 3 1 5 1 1 1 6 1 2 1	YES

Note

第一个样例有不止一种有效接法

Problem J. 这题真的是模板题

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

在看完其他出题人出的毒瘤题之后，良心出题人终于看不下去了，决定出一道模板题来送给大家一个 AC，那么，你们能不能接住这个送来的 AC 呢?

给出一个 n 个结点 m 条边的带权有向图，若图中存在负权回路直接输出 “-1”，否则输出点 s 到每个点的最短路的长度，如果 s 点不连通，输出 “NoPath”。

Input

第一行输入三个值 $n, m, s(2 \le n \le 1000, m \le 10^5, 1 \le s \le N)$

接下来 m 行，每行三个整数 u, v, w 表示 u 和 v 之间有一条权值为 w 的有向边 $(1 <= u, v, s = N, -10^6 <= w <= 10^6)$。

Output

如果存在负权环，输出 “-1”

否则输出 n 行，分别表示 s 点到 i 点的最短路，如果 $s = i$ 输出 0。

Example

standard input	standard output
6 7 1 1 2 6 1 3 4 2 4 5 3 5 4 3 6 3 4 5 1 6 4 2	0 6 4 9 8 7

Problem K. 浣熊市下水道

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

在浣熊市狭窄的下水道中，伴随着枪响，那颗本应射中艾达的子弹被浣熊市新晋警察里昂挡了下来。艾达无法理解里昂为什么会为她挡下这发子弹，但她还是将里昂拖到了安全的地方并用绷带治疗了里昂。由于任务在身，艾达不得已丢下了昏迷的里昂，继续去追击 G 病毒的创造者。当里昂逐渐恢复意识，他发现自己已经是弹尽粮绝，尽管下水道中拥有大量的感染者，但他还是想追上艾达。已知浣熊市下水道的平面图可以简化为一个 n 行 m 列的矩阵 $(3 \le n, m \le 500)$，其中有 P 个僵尸 $(0 \le P \le 50)$，所有的这些僵尸会在一个 $1 * k(2 \le k \le 10)$ 的矩形中来回游走，撕咬所有它们碰到的猎物。好在这些僵尸并不会主动去出击，而只攻击自己闯上门来的倒霉蛋。里昂每次移动都会花费一个单位的时间，而且里昂也不能穿过墙壁、边界或者遇上僵尸，里昂甚至不能够停下，因为一旦停下休息他就没法再站起来。里昂太想赶快追上艾达，请你帮帮他最短需要多长时间才能追上艾达?

Input

第一行输入 4 个整数 $n, m, p, k(3 \le n, m \le 500, 0 \le P \le 50, 2 \le k \le 10)$, 分别表示矩阵的行、列，僵尸的数量，僵尸来回走动的长度。

第 2 行到第 $n+1$ 行输入一个矩阵，每行输入一个字符串，第 i 个字符串的第 j 个字符表示矩阵中第 i 行 j 列的状态，如果字符是 “#” 表示是可以走的路，如果是 “&” 表示是下水道的墙壁。

第 n+2 行到第 $n+p+1$ 行每行输入两个整数 x, y 和一个字符串，第 i 行的数据表示第 i 个僵尸当前时间会从第 x 行 y 列出发，沿着固有的方向前进 K 个单位时间后折返，再走回它之前的位置，再折返，依照这种方法循环下去。第三个字符串表示僵尸行进的方向，“UP” 表示向上走，“LEFT” 表示向左走，“DOWN” 表示向下走，“RIGHT” 表示向右走。数据保证在 k 个长度内僵尸不会碰到边界或者墙壁或者两个僵尸路径交错。

Output

如果里昂能够追上艾达，输出一行整数，表示里昂最短追上艾达的时间。

如果里昂不能追上艾达，输出一行 “You are Dead!”(不包含引号)

Examples

standard input	standard output
3 3 1 3 &&A ### &&L 2 1 RIGHT	2
4 4 1 2 L#&A ##&# #&## #### 3 3 DOWN	You are Dead!

Explanation

样例一说明:

如果用代表僵尸的位置，则起始时状态为

```
&&A
*##
&&L
```

一个单位时间后，里昂向上走，僵尸向左走，两者并没有碰上

```
&&A
#*L
&&#
```

再一个单位时间后，里昂向上走，追上艾达，达成目的。

```
&&L
##*
&&#
```

样例二说明:

初始状态

```
L#&A
##&#
#&*#
####
```

一单位时间后

```
##&A
L#&#
#&##
##*#
```

二单位时间后

```
##&A
##&#
L&*#
####
```

三单位时间后

```
##&A
##&#
#&##
L#*#
```

四单位时间后

```
##&A
##&#
#&*#
#L##
```

里昂如果再向左走的话就会跟僵尸碰个正着，而且不论里昂怎么往回走，在（4,3）总能遇见僵

尸，所以里昂挂了。

Problem L. 大学物理

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

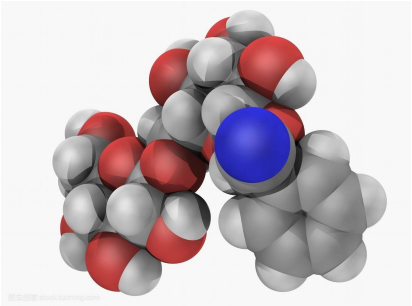
今年大学物理考试的题好像有点难，某学长觉得应该让大家也体验一下，于是就想了一道物理题来让大家感受一下。

众所周知，物质由分子组成，而分子又由原子组成，在近代物理的研究中，德布罗意提出了物质的波粒二象性的概念，可以用波函数描述微观系统状态，而薛定谔给出的薛定谔方程能够正确地描述波函数的量子行为：

$-\frac{\hbar^2}{2\mu}(\frac{\delta^2\Psi}{\delta x^2}+\frac{\delta^2\Psi}{\delta y^2}+\frac{\delta^2\Psi}{\delta z^2})+U(x,y,z)\Psi=i\hbar\frac{\delta\Psi}{\delta t}$

它与一般的波动方程类似，是一个二阶偏微分方程（这里我们当然不需要求解它），用它可以形象的描绘出物质分子的样子。

科学家们需要计算分子的大小，只有知道了分子的大小，才能进行后续的研发。现在你的任务便是计算给定分子的大小。



我们在一个坐标系中描述一个分子，每个分子由 n 个原子组成，由于自己的大学物理题都不会，能及格全靠老师的努力，所以，学长也并不想难为大家，统一用所有边都平行于坐标轴的长方体代替原本近似为球体的原子，每个原子由 6 个整数 x_0, y_0, z_0, x, y, z 表示，其中 x_0, y_0, z_0 表示离坐标原点最近的那个顶点，x, y, z 分别表示该原子沿相应坐标轴正向的长度，我们需要计算由这些原子组成的分子的体积和表面积（注意，分子可能由多个联通块组成）。

为了进一步使问题变得友好，我们约定分子内部如果有密闭的空间，其体积计算在总体积之内，而表面积只计算外表面积，内部的面不算。

Input

输入数据包括多组数据，第一行为数据组数 $T(T \le 100)$。

对于每一组用例，第一行为一个整数 $n(n \le 50)$，表示组成该分子的原子个数。

接下来 n 行，每行包括 6 个整数 $x_0, y_0, z_0, x, y, z(1 \le x_0, y_0, z_0, x, y, z \le 500)$，具体含义如题面所述。

Output

对于每组用例输出一行两个数（用空格隔开），第一个数为该分子的表面积，第二个数为该分子的体积。

Examples

standard input	standard output
1 6 2 2 1 2 2 2 1 2 2 2 2 2 4 2 2 2 2 2 2 1 3 2 2 2 2 4 3 2 2 2 2 2 4 2 2 2	96 44
1 2 1 1 1 1 1 1 3 3 3 1 1 1	12 2

Explanation

对于样例一，可知组成分子的形状如图，其中有一个 1x1x1 的密闭空间，算其体积而不算其表面积，可得表面积为 96，体积为 44。

对于样例二，分子由两个不相连的原子组成，分别计算两个正方体的表面积及体积，相加可得表面积为 12，体积为 6。

Problem M. 是谁在呼叫舰队？

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

亮皇是科普路星区泰伦帝国的一名光荣的指挥官。他乘坐 Hyperion 号，带领了他的游骑兵拯救了星灵，异虫和人类。然而现在虚空中埃萌还在蠢蠢欲动，亮皇必须要去打败这邪恶的 Xel'Naga。亮皇的 Hyperion 号可以进行星际折跃，每次折跃有一个折跃距离 x，在普通空间时，折跃花费 $f(x) = x$，但是在埃萌的虚空里面，折跃花费为 $f(x) = x * ln(x + 1)$ 。因为亮皇的游骑兵们已经在 Hyperion 号和埃萌的连线上驻扎，提供补给，亮皇可以选择折跃到这些补给点上，其他地方太过危险不可以折跃。亮皇很聪明，稍稍一想就知道怎么折跃了花费最少，但是你作为他的副官，也必须要有这样的智慧。

Input

第一行一个数 n 表明补给站的数目，$0 \le n \le 100000$。

第二行 n 个整数表示 n 个补给站和 Hyperion 号的距离 a_i，$0 \le a_i \le 100000000$ 。

第三行一个整数表示埃萌距离 Hyperion 的距离 x ，$1 \le x \le 100000000$。

Output

输出一个数表示最小花费。如果你的答案是 a，正确答案是 b，当且仅当 $min(|a-b|, \frac{|a-b|}{|b|}) < 10^{-6}$ 的时候你的答案会被视为正确。

Example

standard input	standard output
10 1 2 3 4 5 6 7 8 9 10 20	30.9104245