

A.减肥计划

YahAHa 是一个嘴馋的孩子，他每天都要吃大量的零食，这导致他的体重严重超标，因此他的打算减肥，但他希望有人能陪他一起减肥，于是 YahAHa 组织了一场游戏，获胜者将被 YahAHa 选中。

游戏的规则如下： n 个参与者排成一队，游戏从队列中的第 1 个人（从左到右）开始，每一轮由队列的前 2 个人进行游戏，如果第 1 个人的体重 $a_1 \geq$ 队列中的第 2 个人的体重 a_2 ，那么第 1 个人获胜，第 2 个人离开队列并重新加入到队尾（成为队列中的最后一个人），反之第 2 个人获胜，第 1 个人离开队列并重新加入到队尾，之后继续进行游戏，如果某一个人连续获胜 k 轮，那么游戏结束，这个人成为最终的获胜者。

已知一开始队列中每个人的体重，YahAHa 想知道最终获胜的是谁。

输入格式：

第 1 行给定 2 个正整数 n, k ，表示队列中有 n 个人，需要连续获胜 k 轮才能成为最终的获胜者， $2 \leq n \leq 10^6$ ， $1 \leq k \leq 10^9$ 。

第 2 行给定 n 个正整数，按顺序给出初始队列中第 i 个人的体重 a_i ， $1 \leq a_i \leq 10^9$ 。

输出格式：

一个正整数，表示获胜者在初始队列中的位置。

输入样例：

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

输出样例：

```
4
```

Note:

样例解释(括号内表示每个人在初始队列的位置)

第 1 轮， $a_1 \geq a_2$ ，第 1 个人获胜，第 2 个人放入队尾，队列变为 1(1), 4(3), 5(4), 1(5), 4(6), 1(2)。

第 2 轮， $a_1 \leq a_3$ ，第 3 个人获胜，第 1 个人放入队尾，队列变为 4(3), 5(4), 1(5), 4(6), 1(2), 1(1)。

第 3 轮， $a_3 \geq a_4$ ，第 4 个人获胜，第 3 个人放入队尾，队列变为 5(4), 1(5), 4(6), 1(2), 1(1), 4(3)。

第 4 轮， $a_4 \geq a_5$ ，第 4 个人获胜，第 5 个人放入队尾，队列变为 5(4), 4(6), 1(2), 1(1), 4(3), 1(5)。

第 5 轮， $a_4 \geq a_6$ ，第 4 个人获胜，第 6 个人放入队尾，队列变为 5(4), 1(2), 1(1), 4(3), 1(5), 4(6)。

此时第 4 个人已经赢了 3 轮，因此初始顺序中的第 4 个人是最终赢家，因此答案为 4。

B.修建道路 II

在维斯特洛大陆上，有 n 座城堡，共有 $n - 1$ 条道路将这些城堡连接在一起，保证任意两个城堡之间都有且只有一条简单路径。这些道路因为年久失修，现在需要翻新。国王之手现在有一个计划清单，计划清单里包含若干个城堡的编号，他希望将计划清单中每对城堡之间的路径上的道路都翻新一下（同一条道路只会翻新一次）。如果一个城堡相邻的道路得到翻新或者这个城堡在清单中，城堡的领主将赞助 a_i 元。现在，我们假定我们选定的城堡编号分别为 $b_1, b_2, \dots, b_k$ (对于任意的 $i < j$, $1 \leq b_i < b_j \leq n$)，那么收到的总赞助显然为 $\sum a_{b_i}$ 。但是国王之手是一个很麻烦的人，他可能随时会改变计划：增加或删除清单里的城堡。

现在身为财政大臣的 YahAHa 被搞得焦头烂额，他想请你帮他算一算，每次国王之手改变计划后，将预计收到的领主们的赞助的总和。

输入格式：

第一行包含一个整数 T ($1 \leq T \leq 10^5$)，表示测试用例的个数。

每个测试用例的第一行包含三个整数 $n, m(m \leq n), q$ ，分别表示城堡的个数，起始时清单城堡的个数以及操作次数。

第二行，包含 n 个整数，表示第 i 个城堡领主将赞助 a_i ($1 \leq a_i \leq 10^9$) 元。

第 3 到第 $n + 1$ 行，每行有两个整数 u, v ，代表城堡 u 和城堡 v 存在一条道路。

第 $n + 2$ 行，包含 m 个整数，表示起始时清单里的城堡（保证 m 个整数互不相同）。

第 $n + 3$ 到第 $n + q + 2$ 行，每行描述一个操作，其格式为： $op \ u$

当 $op = 1$ 时，表示将城堡 u 加入清单，保证 u 之前一定不在清单中。

当 $op = 2$ 时，表示将城堡 u 从清单中删除，保证 u 之前一定在清单中。

保证对于所有用例 $\sum n \leq 10^5, \sum q \leq 10^5$ 。

输出格式

对于每次操作，输出一行一个整数表示操作之后将收到多少赞助费。

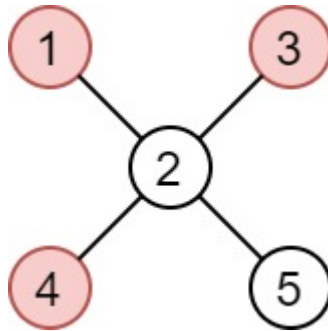
输入样例：

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

输出样例：

```
10
15
```

Note:



对于第一次操作之后，清单有 1、3、4 号城堡，1、2、3、4 号城堡将提供赞助费。

C.测量学

YahAHa 是虎纠大学的学生。作为 HJU-ACM 的队员，他每天要从生活区走到实验室参加训练。

HJU 的教学楼群大体上是图书馆中心为圆心，呈若干个同心圆的分布。

经过测量，YahAHa 发现，生活区和实验室都位于半径为 R 的大道上；从图书馆到大道上一共有 n 条同心圆小道，他们的半径分别为 r_1 到 r_n 。YahAHa 可以在任意同心圆小道上，顺时针或逆时针地绕着小道行走。

此外，HJU 的教学楼群中，有两条大道分别从图书馆直通生活区和实验室。直通生活区的那条大道绕图书馆逆时针旋转 θ 的弧度即和直通实验室的大道重合。

现在，每天准时上班的 YahAHa 想知道，他从生活区出发前往实验室，最短的路程是多长呢？

输入格式：

第一行输入一个整数 $n(1 \leq n \leq 10^5)$ 和两个实数 $R, \theta(0 < R \leq 10^9, 0 \leq \theta < 2\pi)$ 。

第二行输入 n 个实数 $r_i(0 < r_i \leq R)$ 。

数据保证上述的所有实数都是 6 位小数。

输出格式

输出一个实数，表示答案。

若你的答案为 a 而正确答案为 b ，则当且仅当相对误差或绝对误差不超过 10^{-3} 时，你的答案会被认为是正确的。

即当且仅当下述公式满足时，你的答案被认为是正确的：

$$\frac{|a - b|}{\max(1, b)} \leq 10^{-3}$$

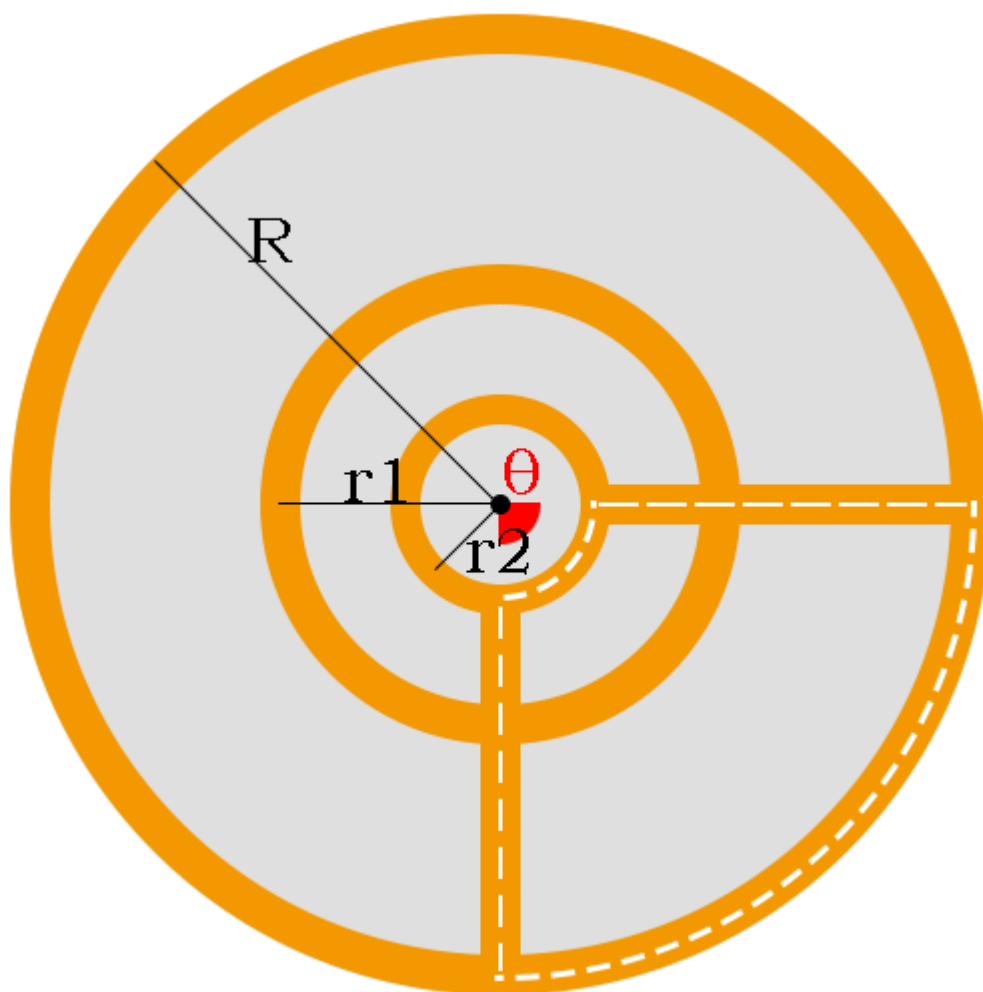
输入样例：

```
2 2.000000 1.570796
1.000000 0.500000
```

输出样例：

```
3.141592
```

Note:



图中灰色部分表示不可通行的教学楼群，橙色部分为可通行的大道、同心圆小道。

YahAHa 需要从图片正下方的生活区走向图片正右方的实验室；图中白色虚线部分表示两条可能的路程。

D.Devil May Cry

在 YahAHa 很喜欢的游戏《Devil May Cry》系列中，有一个角色 Dante，拥有非常多的武器和招式。举例来说，在《Devil May Cry 5》中，Dante 有六把近战武器，六把远程武器，四种战斗风格，三种战斗形态。



每种武器，风格和战斗形态，都将赋予 Dante 不同的技能，因此，Dante 拥有极为丰富的技能（超过200！），利用这些丰富的技能，你可以打出各种各样的花式连招。

现在，YahAHa 想要设计一些 Dante 的连招，连招由若干个依次释放的技能组成，假如在某个连招中一共释放了 n 个技能，则这个连招可以描述成一个下标从 1 开始的长度为 n 的数列 a ，数列中的每个项都是一个正整数，代表一个技能。

一个合法的连招要满足如下条件：

- YahAHa 只想用 m 个技能组成连招，因此对于任意的 $1 \leq i \leq n$ 要满足 $1 \leq a_i \leq m$ （不必每个技能都用上）， m 由输入给出
- 同一个技能不能释放太多次，因为那样会导致评分降低，若用 cnt_i 表示连招中技能 i 出现的次数，则对于任意的 $1 \leq i \leq m$ ，要满足 $cnt_i \leq lim_i$ ，其中 lim_i 由输入给出。
- 有些技能有释放限制，例如：只能在空中释放，因此在连招的某些特定位置不能释放某个技能，具体而言，有若干形如 $a_{p_i} \neq s_i$ 的限制， p_i 和 s_i 由输入给出。

现在 YahAHa 想知道，有多少长度为 n 的合法连招？由于他还想去打游戏，所以他让你帮他解决这个问题。由于答案可能会很大，YahAHa 只想知道答案对 998244353 取模的结果。

输入格式：

第一行包括两个整数 n, m ($1 \leq n \leq 20, 1 \leq m \leq 100$)

接下来一行，包括 m 个整数，其中第 i 个数表示 lim_i ，表示第二条限制 ($0 \leq lim_i \leq n$)。

接下来一行包括一个整数 r ($0 \leq r \leq 1000$)

接下来 r 行, 每行两个正整数 p_i, s_i , 表示第三条限制, $1 \leq p_i \leq n, 1 \leq s_i \leq m$ 。

输出格式:

仅一个整数, 表示答案对 998244353 取模的值

输入样例1:

```
1 1
1
0
```

输出样例1:

```
1
```

输入样例2:

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

输出样例2:

```
2
```

输入样例3:

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

输出样例3:

```
21
```

E.睡觉

YahAHa 最近失眠了，他这几天睡觉经常听着他最喜欢的歌，希望可以帮助入睡。但是这样过了几天，他发现他还是睡不着，他开始怀疑这首歌有问题。

我们用一个整数来衡量 YahAHa 的清醒度，如果连续经过 t 秒清醒度都小于等于 k ，他就可以顺利入睡，清醒度在最开始时为 x ，且 $x \geq k$ 。

当他准备睡觉时，他会从头开始循环播放他最喜欢的歌，这首歌一共有 n 秒，每秒一个数字 a_i 表示这首歌第 i 秒的分贝值。如果第 i 秒时，分贝值 a_i 小于等于 d ，在第 i 秒的开始，YahAHa 的清醒度会减少 1，否则 YahAHa 的清醒度会增加 1。

介于最近的持续失眠，他想问问你，从他每晚放这首歌开始（即“清醒度连续 t 秒小于等于 k ”只从歌曲播放后开始计算），在无限长的时间内，他能否成功的入睡。如果可以请你输出"YES"，不能请输出"NO"。

输入格式：

第一行一个整数 $T(1 \leq T \leq 10)$ 表示数据组数。

接下来每组数据两行。

第一行 5 个整数 $x, t, k, n, d(1 \leq x, t, k, d \leq 10^9, 1 \leq n \leq 10^5)$ ，含义如题面描述。

输入数据保证 $k \leq x$ 。

接下来一行 n 个整数，第 i 个数字 $a_i(1 \leq a_i \leq 10^9)$ 表示 YahAHa 最喜欢的歌第 i 秒的分贝值。

输出格式：

每组数据输出一行，"YES" 或 "NO" 表示 YahAHa 是否能成功入睡。

输入样例：

```
2
5 3 4 5 5
3 4 6 7 8
5 4 4 5 5
3 4 6 7 8
```

输出样例：

```
YES
NO
```

Note：

第一组数据中最初 YahAHa 的清醒度为 5，从播放歌曲开始：

第 1 秒， $a_1 = 3$ 小于等于 d ，清醒度减少 1，变为 4。

第 2 秒, $a_2 = 4$ 小于等于 d , 清醒度减少 1, 变为 3。

第 3 秒, $a_3 = 6$ 大于 d , 清醒度增加 1, 变为 4。

至此为止, 从第 1 秒的开始到第 3 秒的结束, 经过 3 秒, 清醒度都小于等于 k , 符合条件, 此时 YahAHa 可以顺利入睡。

第 4 秒, $a_4 = 7$ 大于 d , 清醒度增加 1, 变为 5。

第 5 秒, $a_5 = 8$ 大于 d , 清醒度增加 1, 变为 6。

第二组数据, 清醒度变化的过程同上, 容易发现不存在长为 4 秒的时间, 清醒度始终小于等于 4。

Note:

第一组数据中最初 YahAHa 的清醒度为 5, 从播放歌曲开始:

第 1 秒, $a_1 = 3$ 小于等于 d , 清醒度减少 1, 变为 4。

第 2 秒, $a_2 = 4$ 小于等于 d , 清醒度减少 1, 变为 3。

第 3 秒, $a_3 = 6$ 大于 d , 清醒度增加 1, 变为 4。

至此为止, 从第 1 秒的开始到第 3 秒的结束, 经过 3 秒, 清醒度都小于等于 k , 符合条件, 此时 YahAHa 可以顺利入睡。

第 4 秒, $a_4 = 7$ 大于 d , 清醒度增加 1, 变为 5。

第 5 秒, $a_5 = 8$ 大于 d , 清醒度增加 1, 变为 6。

第二组数据, 清醒度变化的过程同上, 容易发现不存在长为 4 秒的时间, 清醒度始终小于等于 4。

F.位运算谜题

YahAHa 有三个非负整数 a, b, c ，他想和你玩个游戏，他不告诉你 a, b, c 分别是多少，但他会告诉你这三个数两两之间的按位异或，按位或，以及按位与的结果，总共九个数，注意：你只知道这九个数的值，但不知道这九个数哪个数具体是哪个运算的结果。

现在他想请你还原 a, b, c 的值。保证存在解，如果有多组解，任意输出一组即可。

输入格式：

第一行输入一个正整数 t ，表示共有 t 组数据，保证 $1 \leq t \leq 10000$ 。

接下来 t 组数据，每组占一行，每行九个非负整数，含义如题所述，保证这一行所有数均不超过 $2^{60} - 1$ 。

输出格式：

对于每组数据输出一行，行内三个空格分开的非负整数表示 a, b, c ，需满足 $0 \leq a, b, c \leq 2^{60} - 1$

输入样例：

```
3
0 3 3 0 3 0 3 3 3
1 0 7 7 6 0 0 1 6
0 2 2 7 7 2 7 0 5
```

输出样例：

```
3 3 3
0 1 6
2 7 0
```

G.排队打卡

最近 YahAHa 所在的学校最近非常神秘，学校要求所有学生每天早起参加一个全员打卡的活动。学生可以在活动的入口排队，在每一秒结束前会放一次排队的人进入入口，一次最多进入不超过 k 个人。

YahAHa 非常的懒，他想找到一个时间参与排队以达到最快地完成这个活动。经过几天的观察，YahAHa 整理出了一份打卡日志，他在日志上记载了若干这样的事件“在第 t_i 秒的开始会有 x_i 个人到达打卡现场”。有天他要验证这份日志的正确性。这天他在第 t 秒的开始睡醒，他看到打卡入口的队列已经有 n 个人了。如果按照日志模拟排队的情况，第 t 秒确实是 n 个人(排队的队伍一开始是没有人的)，YahAHa 就会认为他的日志是完美无缺的。

由于 YahAHa 比较社恐，不敢一个人去排队，他只会跟在某一个时刻去排队的人后面去排队。他会模拟他的日志，直到找到排队时间最短的时候，跟在那个时间排队的人后面一起去排队。这里排队的时间指的是，从 YahAHa 加入队列的时刻直到 YahAHa 被放行进入入口的这段时间。

由于他需要在很快的时间内找出方案，于是他来请教聪明的你。

输入格式：

第一行两个数字 t 和 n 表示 YahAHa 在第 $t(0 \leq t \leq 10^9)$ 秒醒来，此时已经有 $n(0 \leq n \leq 10^{18})$ 个人在排队了。

第二行两个数字 $m, k(1 \leq m \leq 5 \times 10^5, 1 \leq k \leq 10^9)$ 表示打卡日志中事件的个数以及每秒放进入口的人数。

接下来 m 行，每行两个数字 $t_i(0 \leq t_i \leq 10^9)$ 和 $x_i(1 \leq x_i \leq 10^9)$ 表示第 t_i 秒有 x_i 人加入队列的后面。

保证对于任意的 i, j ， $t_i \neq t_j$ ，保证任意的 $t_i \neq t$ ，保证存在 t_i 使得 $t_i > t$ 。

输出格式：

如果 YahAHa 的日志没问题，请输出两个整数，用空格隔开，表示参与排队的时刻和排队等待的时间。如果两个方案排队等待的时间一致，请输出参与排队时刻更靠后的那个。

如果 YahAHa 的日志有问题，请输出 "Wrong Record"（没有引号）。

输入样例1：

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

输出样例1：

```
5 4
```

输入样例2：

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

输出样例2:

```
Wrong Record
```

Note:

第一组数据中，YahAHa 在第 3 秒醒来，此时队列中有 3 个人，每过 1 秒，会有 1 个人被放入口。

模拟日志的过程如下：

第 1 秒的开始，队列中加入了 2 个人，此时 YahAHa 还没有醒，无法去排队。

第 1 秒结束时，有 1 人被放行，此时队列中还剩 1 个人。

第 2 秒的开始，队列中加入了 3 个人，此时队列中有 4 个人，此时 YahAHa 还没有醒，无法去排队。

第 2 秒结束时，有 1 人被放行，此时队列中还剩 3 个人。

第 3 秒的开始，YahAHa 醒来，此时队列中有 3 个人，符合 YahAHa 醒来时看到的人数，但因这一秒没有人去排队，YahAHa 无法跟着人去排队。

第 3 秒结束时，有 1 人被放行，此时队列中还剩 2 个人。

第 4 秒的开始，队列中加入了 1 个人，此时队列中有 3 个人；

此时 YahAHa 如果跟着去排队，从加入队列到被放行，需要等待 $(3 + 1)/1 = 4$ 秒，从第 4 秒的开始等到第 7 秒的结束

第 4 秒结束时，有 1 人被放行，此时队列中还剩 2 个人。

第 5 秒的开始，队列中加入了 1 个人，此时队列中有 3 个人；

此时 YahAHa 如果跟着去排队，需要等待 $(3 + 1)/1 = 4$ 秒，从第 5 秒的开始等到第 8 秒的结束。

因此 YahAHa 会选择跟着第 5 秒的队伍去排队，需要等待 4 秒。

第二组数据，日志模拟的过程同上，由于记录与看到的人数冲突，因此是 "Wrong Record"。

H.提瓦特之旅

熠熠星辰闪耀天际

旅程中的每一次相逢

都将被细细镌刻在

旅者传颂的歌谣中

YahAHa 准备和 ColdHeat 一起遨游提瓦特大陆。提瓦特大陆可以视为 n 个点 m 条边的无向连通图，如果 u 和 v 之前存在一条无向边，则从 u 到 v 或者从 v 到 u 需要的时间为 c_{uv} 。

刚开始 YahAHa 位于 1 号节点，ColdHeat 位于 t 号节点。YahAHa 想要尽快赶到 ColdHeat 身边，不过 YahAHa 经过的每一个节点都有可能刷新委托。为了帮 ColdHeat 抽到喜欢的角色 YahAHa 会顺手解决他们并获取原石，经过路上的第 i 个点时 YahAHa 需要 w_i 的时间完成委托，最多一共刷新 $n - 1$ 个委托。现在 YahAHa 想知道他最短需要多少时间找到 ColdHeat。

换言之，一共有 q 次询问，每次询问给出 $t, w_1, w_2, \dots, w_{n-1}$ ($t \neq 1$)，询问从 1 到 t 所需的最短时间。如果当前的路径经过了 $k + 1$ 个节点，分别为 $u_0 = 1, u_1, u_2, \dots, u_k = t$ ，则 YahAHa 所需额外的时间为 $w_1 + w_2 + \dots + w_k$ 。

输入格式：

第一行给定两个整数 n, m ($1 \leq n \leq 500, 1 \leq m \leq 10^5$)。

第二行开始 m 行，每行三个整数 u, v, c_{uv} ($1 \leq c_{uv} \leq 10^9$)。

第 $m + 2$ 行给定一个整数 q ($1 \leq q \leq 5000$)。

然后开始 q 行，每次询问给出 n 个整数 $t, w_1, w_2, \dots, w_{n-1}$ ($2 \leq t \leq n, 0 \leq w_i \leq 10^9$)。

输出格式：

q 行，每行一个整数表示答案。

输入样例：

```
4 5
1 2 1
2 3 1
2 4 1
3 4 1
1 4 3
2
4 0 0 0
4 0 2 0
```

输出样例：

I.宠物对战

最近 YahAHa 沉迷于一款宠物对战的游戏，这个游戏有两只宠物对战。YahAHa 经常用电脑模拟对战的过程，但最近程序出了问题，于是他碰到了一个问题，他想来问问聪明的你。

现在有两类字符串， A 类和 B 类， A 类字符串有 n 个， B 类字符串有 m 个，每个字符串仅由小写字母组成。用这两类字符串交替拼接可以形成一个新的字符串，形如 $abababab.....$ 和 $bababababa.....$ ，其中 a 表示 A 类中的某个字符串， b 表示 B 类中的某个字符串。注意，同一个字符串可以被重复利用。

现在他有一个字符串 S ，他想问问你， S 是否是由 A 、 B 两类字符串交替拼接而成（即是否是上述的两种形式之一）。如果是，请输出 S 最少由多少个字符串拼接而成；如果不是，请输出 -1。

输入格式：

第一行一个整数 $n(1 \leq n \leq 10^5)$ 表示 A 类字符串的个数。

接下来 n 行，每行一个字符串 a_i 表示 A 类里的字符串。 $\sum |a_i| \leq 5 \times 10^5$

接下来一行一个整数 $m(1 \leq m \leq 10^5)$ 表示 B 类字符串的个数。

接下来 m 行，每行一个字符串 b_i 表示 B 类里的字符串。 $\sum |b_i| \leq 5 \times 10^5$

接下来一行一个字符串 S 表示 Stump 问你的字符串($1 \leq |s| \leq 5000$)。

输出格式：

一行一个整数表示答案。

输入样例1：

```
2
cc
pc
1
female
femalecc
```

输出样例1：

```
2
```

输入样例2：

```
2
cc
pc
1
female
ccpcfemaled
```

输出样例2:

```
-1
```

输入样例3:

```
2
cc
pc
1
female
ccfemalepc
```

输出样例3:

```
3
```

输入样例4:

```
2
cc
pc
1
female
femalecc
```

输出样例4:

```
2
```

J.瑞士轮

在一些竞技性比赛中，常在队伍数量较多时采用瑞士轮赛制，既保证了一定的公平稳定性，又让赛程不至于太长。

YahAHa 最近在观看一个瑞士轮赛制的竞技比赛，这个比赛共有 32 支队伍，每个队伍按 1 到 32 编号。所有队伍的积分都可以表示成 $x - y$ 的形式，表示累计 x 个胜场和 y 个负场，一开始所有队伍的积分都是 $0 - 0$ 。比赛有若干轮，若一支队伍累计积满 2 个胜场，将直接晋级，不再参与后续比赛；同样若累计积满 2 个负场，将直接淘汰，也不再参与后续比赛。

在比赛的每一轮中，同一种积分的队伍按编号排序，例如在第 R 轮中， $1 - 1$ 的队伍有 $2N$ 支，将他们按编号排序后，第 1 和第 2、第 3 和第 4、.....、第 $2N - 1$ 和第 $2N$ ，各进行一场比赛。保证 32 支队伍按这样的规则进行比赛，每一轮中每一种积分的队伍数都是偶数，且最终会有 16 支队伍晋级。

单纯这样的比赛当然不够刺激，于是比赛方出了一个竞猜活动。如果要参与竞猜，要提交一份竞猜表，竞猜表包含三个部分，第一部分和第二部分要各填入 1 个队伍，第三部分要填入 7 个队伍。三个部分总共要填入 9 支队伍，并且这 9 支队伍互不相同。

当第一部分中的队伍 $2-0$ 晋级时，算作正确预测。

当第二部分中的队伍 $0-2$ 淘汰时，算作正确预测。

第三部分的队伍，每有一个晋级（最终得分为 $2-0$ 或 $2-1$ ），就算作一个正确预测。一个竞猜表的正确预测队伍数量，是其三个部分的正确预测数量之和。

每个队伍有一个实力值 a_i ，如果第 i 支队伍和第 j 支队伍对阵比赛，第 i 支队伍有 $a_i / (a_i + a_j)$ 的概率获胜，第 j 支队伍有 $a_j / (a_i + a_j)$ 的概率获胜。YahAHa 对竞猜非常谨慎，他现在有好几份竞猜表，他想问问聪明的你，每一个竞猜表正确预测队伍数量的期望。为防止精度误差，请输出期望对 998,244,353 取模的结果。

输入格式：

一开始四行，每行 8 个数，第 i 个数 a_i ($1 \leq a_i < 998,244,353$) 表示编号为 i 的队伍的实力值。保证对于任意不相等的 i, j , $a_i + a_j \neq 998,244,353$ 。

接下来一行一个数 m ($1 \leq m \leq 10^5$) 表示竞猜表的数量。

接下来 m 行，每行 9 个数，第一个数表示第一部分预测队伍的编号，第二个数表示第二部分预测队伍的编号，接下来 7 个数表示第三部分预测队伍的编号。保证对于任意的 b_i , $1 \leq b_i \leq 32$ 。

输出格式

一共 m 行，第 i 行一个整数 x_i 表示第 i 份竞猜表的正确预测数量的期望。

输入样例1：

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

输出样例1：

4

输入样例2：

```
1 2 3 4 5 6 7 8
9 10 11 12 13 14 15 16
17 18 19 20 21 22 23 24
25 26 27 28 29 30 31 32
4
1 2 3 4 5 6 7 8 9
10 11 12 13 14 15 16 17 18
19 20 21 22 23 24 25 26 27
28 29 30 31 32 1 2 3 4
```

输出样例2：

```
526205442
233631779
899336506
198951272
```

K.区间和

给定一个数列 $[a_1, a_2, \dots, a_n]$ ，这个数列中只包含 0 和正整数，保证所有 a_i 的和不超过 10^6 。

定义一个区间 $[l, r]$ 的权值为区间内所有元素的和。

依据这个权值对所有 $n(n + 1)/2$ 个区间按照权值从小到大的顺序排序。

接下来给你 m 个询问，每个询问给出一个 k ，请你输出排第 k 的区间的权值。

输入格式：

第一行一个整数 $n(1 \leq n \leq 10^6)$ 表示数列的数字个数。

第二行 n 个整数，其中第 i 个数 a_i 表示数列中的第 i 个数 ($a_i \geq 0, \sum a_i \leq 10^6$)。

第三行一个整数 $m(1 \leq m \leq 10^6)$ 表示询问的个数。

接下来 m 行，每行一个整数 $k(1 \leq k \leq n(n + 1)/2)$ ，表示询问排序后的第 k 个区间的权值。

输出格式：

一共 m 行，每行一个整数表示答案。

输入样例：

```
6
1 2 0 3 0 1
7
1
2
3
11
12
17
21
```

输出样例：

```
0
0
1
3
3
5
7
```

L.彩色的树

YahAHa 特别喜欢有颜色的树，他有一棵 n 个点、以 1 为根的有根树，树上每个点有一个颜色。YahAHa 的幸运数字是 k ，于是他想问问你，在这棵树中，以 x 为根的子树内，与 x 距离不超过 k 的点中，一共出现了多少种不同的颜色。定义 x 与 y 两个点的距离为， x 到 y 的简单路径上边的个数。YahAHa 有很多次这样的询问想问你，他一共会问你 m 次。

输入格式：

第一行两个数 $n(1 \leq n \leq 10^5)$ 和 $k(0 \leq k \leq n)$ ，分别表示有根树的节点个数和询问中的距离。

第二行 n 个整数，其中第 i 个整数 $c_i(1 \leq c_i \leq n)$ 表示 i 号节点的颜色。

接下来 $n - 1$ 行，每行两个整数 x 和 y 表示 x 和 y 之间连有一条边。

接下来一行一个整数 $m(1 \leq m \leq 10^5)$ 表示询问的次数。

接下来 m 行每行一个整数 $x_i(1 \leq x_i \leq n)$ ，表示询问以 x_i 为根的子树。

输出格式：

共 m 行，每行一个整数表示答案。

输入样例：

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

输出样例：

```
2
2
1
1
```