

2021 “科大讯飞杯” 上海市大学生程序设计竞赛 暨 CCPC 2021 上海市赛



正式赛

- A 小 A 的点面论
- B 小 A 的卡牌游戏
- C 小 A 的期末考试
- D Zztrans 的班级合照
- E Zztrans 的庄园
- F 鸡哥的限币令
- G 鸡哥的雕像
- H 鸡哥的 AI 驾驶
- I 对线
- J Alice and Bob I
- K Alice and Bob II
- L 高低奥马哈扑克

上海大学

2021 年 7 月 11 日

Problem A. 小 A 的点面论

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 1024 megabytes

众所周知，点面论是小 A 最擅长的科目。一个只看到点的人，会被小 A 说格局小了，一个只看到面的人，小 A 又会认为没有成绩。

于是，小 A 觉得我们既要看到点，也要看到面，点面结合才能成大事！

现在，小 A 在三维空间上选取了四个互异的点。但我们不光要看到点，也要看到面，小 A 又将四个点分组，组成了两个非零不共线向量。这样一来，这两个向量就能组成一个平面。

点面论出色的小 A 想考考你，他要你输出一个与这两个向量同时垂直的向量。

Input

第一行有三个正整数 x_1, y_1, z_1 ($0 \leq x_1, y_1, z_1 \leq 10$)，表示小 A 给出的第一个向量 $\vec{v}_1 = (x_1, y_1, z_1)$ 。

第二行有三个正整数 x_2, y_2, z_2 ($0 \leq x_2, y_2, z_2 \leq 10$)，表示小 A 给出的第二个向量 $\vec{v}_2 = (x_2, y_2, z_2)$ 。

输入保证 $\vec{v}_1 \neq \vec{v}_2 \neq \vec{0}$ 。

Output

在一行输出三个整数 x, y, z ，中间以空格分隔，表示所求向量 $\vec{v} = (x, y, z)$ 。

如果有多个向量满足条件，输出任意一个即可。

不过由于小 A 的数学并不是特别理想，请确保输出的每一维都是 $[-200, 200]$ 之间的整数，且这个向量不是零向量，否则他无法判断你的答案是否正确。

Example

standard input	standard output
1 0 0 0 1 0	0 0 1

Problem B. 小 A 的卡牌游戏

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 1024 megabytes

小 A 最近沉迷于一款名为 Hearthverse 的卡牌游戏。在这款游戏中，卡被分为了三个种类（随从、法术和魔法阵），在组卡时，这款游戏严格规定了卡组中每种卡牌的数量，具体来说，一副 n 张卡的卡组需要包含恰好 a 张随从卡， b 张法术卡和 c 张魔法阵卡，并且 $a + b + c = n$ 。

在游戏中，有一个叫做“3pick”的组卡竞技方式，游戏会给出 n 次三选一的机会，三张卡分别来自三个种类。每次玩家需要从三张卡中选出一张卡加入自己的卡组。为了使卡组强度尽量高，小 A 给每张卡设定了一个强度值，而卡组的强度就是所有卡强度值的和，他希望他最后组出的卡组的强度可以尽量高。

Input

第一行有四个整数 n, a, b, c ($1 \leq a, b, c \leq n \leq 5000, a + b + c = n$)，分别表示卡组的总卡数与每种种类的卡的张数要求。

接下来 n 行，每行代表一次“3pick”的机会，第 i 行有三个整数 a_i, b_i, c_i ($1 \leq a_i, b_i, c_i \leq 10^9$) 分别表示这一次选择中随从卡、法术卡和魔法阵卡的强度值。

Output

输出一个整数，表示小 A 能组出的卡组的最大强度值。

Examples

standard input	standard output
3 1 1 1 4 1 6 1 1 10 5 7 4	21
6 3 2 1 1000000000 1 1 1000000000 1 1 1000000000 1 1 1 1000000000 1 1 1000000000 1 1 1 1000000000	6000000000

Problem C. 小 A 的期末考试

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 1024 megabytes

春季学期的期末考试不久前刚结束，小 A 也考完了他大学的所有考试。他很感伤，仿佛已经毕业了一样。但我们知道 SHU 是没有补考机制的，害怕挂科的小 A 决定进入盗梦空间里去修改他的考试分数。修改规则如下：

1. 假如他的成绩没有及格（即低于 60 分）他会把自己改成 60 分。
2. 同时由于学校里没有人喜欢小 A，他为了平衡他敏感弱小的心理，他也会修改其他人的分数，若某个除了他以外的人的成绩大于等于平均分（按所有人初始成绩计算），小 A 会把他的成绩改低 2 分，但不会低于 0 分。

现在小 A 想知道他的梦里，每个人都会是多少分，请你帮帮他。

Input

第一行有两个整数 n, m ($1 \leq n \leq 100, 1 \leq m \leq n$) 分别表示总人数和小 A 的学号。

接下来 n 行，第 i 行有两个整数 s_i, a_i ($1 \leq s_i \leq n, 0 \leq a_i \leq 100$)，表示学号为 s_i 的同学的分数为 a_i 。

数据保证任意两个同学的学号不同。

Output

在一行按学号的递增顺序输出 n 个整数表示每个人最终的分数，分数之间以空格分隔。

Examples

standard input	standard output
3 2 1 1 2 4 3 100	1 60 98
4 2 4 49 2 98 3 1 1 22	22 98 1 47

Problem D. Zztrans 的班级合照

Input file: standard input
Output file: standard output
Time limit: 3 seconds
Memory limit: 1024 megabytes

Zztrans 决定和他的同学们去拍一张班级合照，为了让每个人都能在照片里出境，他们约定按如下方式排队进行拍照：排成两排，每排从左向右身高都不递减，且第二排同学的身高不低于第一排对应位置同学的身高。

Zztrans 擅长排序，他得到了每个同学身高从低到高排序后的排名（身高相同的同学排名相同），这样他就不用一个个统计每个同学的身高了。但是他并不擅长数数，希望你教教他一共有多少种排队的方案。

由于答案可能很大，你只需要输出答案对 998244353 取模后的结果。

Input

第一行有一个整数 n ($2 \leq n \leq 5000$)，表示 Zztrans 和他同学们的总人数，保证 n 为偶数。

第二行有 n 个整数 a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$)，中间以空格分隔，分别表示每个同学的身高排名。

Output

在一行输出一个整数，表示 Zztrans 和他的同学们按约定可能的排队方案数对 998244353 取模后的结果。

Examples

standard input	standard output
4 1 2 3 4	2
4 1 1 1 1	24
10 1 1 3 3 5 5 5 5 9 10	960

Problem E. Zztrans 的庄园

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 1024 megabytes

Zztrans 最近沉迷庄园，他又可以种地了。

在庄园中，要成为一个好的农夫升级农场，每天还必须钓鱼，作为一个精打细算的农夫，他想知道他一天钓鱼的期望收益是多少。

庄园里，鱼的稀有度分为以下五个等级，“普通”，“稀有”，“罕见”，“珍稀”，“传说”，分别用 D, C, B, A, S 来表示。

一般来说，钓来的鱼都会卖给一个叫花婶的精明女商人，对于以上稀有度的鱼，她分别收 16, 24, 54, 80, 100 豆子。你可以认为前面四类鱼对于 Zztrans 来说就值这么些钱。

但“传说”鱼就不一样了，除了概率比较低外，钓某些“传说”鱼还要特定天气特定时间。每次钓到传说鱼，Zztrans 都会高兴得跳起来，所以 Zztrans 认为传说鱼一条值 10000 豆子。(即传说鱼不再被认为是原收购价 100 豆子，而是 10000 豆子)

但钓鱼不能空手套白“鱼”，钓一条鱼需要 23 豆子的鱼饵。

现在假设你知道鱼塘里有 n 种鱼以及他们各自的稀有度和钓到的概率，以及 Zztrans 今天准备钓多少杆，Zztrans 想请你教教他今天的期望收益是多少豆子？

Input

第一行有二个整数 n, k ($1 \leq n \leq 100, 1 \leq k \leq 100$)，分别表示鱼塘里鱼的种类数和 Zztrans 准备钓的杆数。

接下来 n 行，第 i 行有一个字符 t , $t \in \{D, C, B, A, S\}$ 和一个实数 P_i ($0.00 \leq P_i \leq 1.00$)，分别表示第 i 条鱼的等级和钓到的概率。

保证所有的 P_i 均给到小数点后两位， $\sum P_i = 1.00$ 。

Output

在一行输出一个实数，表示 Zztrans 今天的期望收益。你的答案的相对或绝对误差不超过 10^{-4} 会被认为是正确的。

形式化地说：令你的答案为 a ，标准答案为 b ，你的答案会被判为正确当且仅当 $\frac{|a-b|}{\max(1, |b|)} \leq 10^{-4}$ 。

Example

standard input	standard output
7 2 D 0.33 D 0.11 D 0.06 C 0.30 B 0.15 A 0.04 S 0.01	207.0000

Problem F. 鸡哥的限币令

Input file: standard input
Output file: standard output
Time limit: 10 seconds
Memory limit: 1024 megabytes

众所周知，鸡哥炒币亏了很多钱，他的钱大部分都被无良庄家骗走了。那么，打不过就加入，鸡哥也决定下海成为一个庄家。于是机智的鸡哥决定开一个炒币平台，取名“鸡币”。

在鸡币平台中，有 n 个商家，编号为 1 到 n 。商家之间存在 m 个单向的供货关系，编号为 1 到 m 。一条 $u \rightarrow v$ 权值为 w 的边代表商家 u 向商家 v 卖 w 的虚拟货币。

但好景不长，鸡哥才小赚没几天，就出现了问题。原来是鸡哥的服务器是土豆做的，无法承受这么多交易。为了减轻服务器的压力，鸡哥决定出台限币令限制一些供货关系，使虚拟货币的交易总额尽可能小，但不能没有交易让鸡哥白白运维服务器亏钱。

鸡哥立志建立良好币圈生态，同时还能每天小赚到一点。而有些商家只卖不买，有些商家只买不卖，就会造成了市场的不平衡。于是，鸡哥又在限币令中加了一条规定：每个商家对于与其他商家的买和卖的供货关系至少各存在一组。

请你帮帮鸡哥选出一组满足上述要求的最好的商家供货关系，使得它们的交易总额最小。鸡哥会报答你的。

Input

第一行有两个整数 n, m ($2 \leq n \leq 300, 1 \leq m \leq n \cdot (n - 1)$)，分别表示鸡币平台商家数量和商家之间供货关系数量。

接下来 m 行，每行三个整数 u, v, w ，表示商家 u 向商家 v 卖 w 的虚拟货币。($1 \leq u, v \leq n, u \neq v, 1 \leq w \leq 10^5$)。保证从 u 向 v 的供货关系最多只会有一条。

Output

如果鸡哥能成功出台限币令，在第一行输出一个整数 w ，表示总的最少交易额。

第二行输出一个整数 k ，表示鸡哥限币令下仍然开放的供货关系，接着输出 k 个整数， a_1, a_2, \dots, a_k ，分别表示鸡哥认可的供货关系的编号，中间用空格分隔。如果有多种符合条件的方式，输出任意一种即可。

如果鸡哥不能找到一组满足他要求的供货关系，则输出一行 -1 。

Examples

standard input	standard output
3 5 1 2 1 2 1 2 1 3 3 3 1 4 3 2 5	10 3 2 3 5
3 3 1 2 1 2 3 1 1 3 1	-1

Problem G. 鸡哥的雕像

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 1024 megabytes

7777 年，为了表彰著名币学家尼古拉斯·鸡哥在虚拟货币市场运转、炒币理论方面的杰出贡献，奥林匹斯市决定在韭菜街的路上为他修建雕像。

韭菜街是一条漫长的道路，这条道路上布满了韭菜，但各种韭菜由于发育的不一样，会拥有各种奇怪的姿势和形态，有的韭菜已经成熟，会被修剪和打压，而有的韭菜正在茁壮成长着，等待着成熟。尼古拉斯-鸡哥的雕像就决定以这些韭菜们为背景，修建在这些韭菜中的某一棵上。

现在有 n 棵可供选择的韭菜背景板，每棵韭菜都会有一个发育度，第 i 棵韭菜的发育度为 a_i 。修建雕像当然是为了让行人观赏膜拜，而如果将鸡哥的雕像修建在第 i 根的韭菜上，就能够让行人获得剩下除了这根韭菜以外所有韭菜的发育度乘积的愉悦值。

奥林匹斯市的市长想知道，若将鸡哥的雕像修建在所有韭菜中的某一棵上行人能够获得的愉悦值，请你帮助他。由于答案可能很大，你只需输出答案对 998244353 取模后的结果。

Input

第一行有一个整数 n ($2 \leq n \leq 10^5$)，表示韭菜街上的韭菜数。

第二行有 n 个整数 a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$)，中间以空格分隔，分别表示每根韭菜的发育度。

Output

在一行输出 n 个整数 x_1, x_2, \dots, x_n ，中间以空格分隔。分别代表将鸡哥雕像修建在第 i 根韭菜上行人能获得的愉悦值对 998244353 取模后的结果。

Examples

standard input	standard output
3 2021 7 11	77 22231 14147
4 114 514 1919 810	798956460 177200460 47462760 112445724

Problem H. 鸡哥的 AI 驾驶

Input file: standard input
Output file: standard output
Time limit: 4 seconds
Memory limit: 1024 megabytes

鸡哥最近正在制作一款游戏《狭道赛车手 X》，他为游戏中的车辆编写了一个自动驾驶 AI，使得游戏中的车辆的车辆能够在将要撞到别的车辆时自动绕过对方，且不需要减速。

然而游戏上线后鸡哥突然发现代码中存在一个 bug，会让不同型号的车辆相遇时发生事故（追尾或者相撞）！

鸡哥希望能在玩家发现这个 bug 前推出一个《1.22 热修复补丁》，否则游戏的评分就会非常难看。在他开始编写代码前，鸡哥想要知道玩家还要多久才能发现这个 bug。一旦有事故发生，玩家就会立刻发现这个 bug。

鸡哥将每一辆车看做一条直线上的点，每辆车以（位置，速度，型号）这样的三元组给出。当型号相同的两辆车位于同一坐标时，不会发生事故，但不同型号的两辆车位于同一坐标时，就会发生车祸。

鸡哥觉得这个任务非常简单，于是他将这个任务交给你了，为了给鸡哥一些时间来推送这个补丁，你需要告诉鸡哥的时间是 t ：满足在 $[0, t]$ 时间内没有发生事故，在 $(t, t + 1]$ 的时间内发生了某起（或多起）事故。

Input

第一行有两个整数 n, k ($1 \leq k \leq n \leq 10^5$)，分别表示车辆的数量和不同型号的种数。

接下来的 n 行，第 i 行有三个整数 p_i, v_i, t_i ($-10^9 \leq p_i, v_i \leq 10^9, 1 \leq t_i \leq k$)，分别表示第 i 辆车的位置、速度和型号。

输入保证任意两辆车初始时不会在同一位置。

Output

在一行输出一个整数，表示你需要告诉鸡哥的时间 t 。

特别地，如果永远不会有事故发生，输出 -1 。

Examples

standard input	standard output
5 2 0 10 1 5 1 1 12 -1 2 15 -1 2 -10 5 2	1
5 1 10 9 1 100 99 1 99 -10 1 50 40 1 998233 -10000 1	-1

Note

在样例一中，第 2 秒时，1 号车与 3 号车已经相撞，因此你需要告诉鸡哥的时间是 1。

在样例二中，所有车辆都是同一型号，玩家永远也发现不了这个 bug。

Problem I. 对线

Input file: standard input
Output file: standard output
Time limit: 12 seconds
Memory limit: 1024 megabytes

ACPC 是一个新出的对线的战场, 在这个战场每一方各有三道兵线, 作为其中一方的你需要实时观察并调配自己的兵线以维持局势。具体的, 你有三排兵线, 长度均为 n , 每排兵线的位置从左往右编号依次从 1 到 n , 各排同编号的位置对齐。为实时观察和调配兵线, 你需要以下完成四件事情。

- 0、数出第 x 排从 l 位置到 r 位置间所有的士兵数。
- 1、往第 x 排从 l 位置到 r 位置间所有位置各增派 y 名士兵。
- 2、将第 x 排和第 y 排兵线的同编号位置上的士兵进行交换位置。
- 3、你发现你的士兵会分身术, 命令第 x 排从 l 位置到 r 位置所有士兵引导分身术, 并将分身派往第 y 排, 每个分身前往与当前编号相同的位置。

开始时, 每个位置上都没有士兵。只要实时对兵线完成上述的观察和调配, ACPC 战场就可以保持稳定, 稳定才有利于发展!

由于答案可能过大, 你只需要输出答案对 998244353 取模后的结果。

Input

第一行有两个整数 n, q ($1 \leq n, q \leq 3 \times 10^5$), 分别表示兵线的长度和事件的数量。

接下来 q 行, 每行第一个整数 op 代表事件的类型。

- 如果 op 为 0, 接下来会有三个整数 x, l, r ($x \in \{1, 2, 3\}, 1 \leq l \leq r \leq n$) 代表你要数出第 x 排从 l 位置到 r 位置间所有的士兵数。
- 如果 op 为 1, 接下来会有四个整数 x, l, r, y ($x \in \{1, 2, 3\}, 1 \leq l \leq r \leq n, 1 \leq y \leq 10^9$) 代表你要往第 x 排从 l 位置到 r 位置间所有位置各增派 y 名士兵。
- 如果 op 为 2, 接下来会有四个整数 x, y, l, r ($x, y \in \{1, 2, 3\}, 1 \leq l \leq r \leq n$) 代表你要将第 x 排和第 y 排兵线的同编号位置上的士兵进行交换位置。
- 如果 op 为 3, 接下来会有四个整数 x, y, l, r ($x, y \in \{1, 2, 3\}, 1 \leq l \leq r \leq n$) 代表你要命令第 x 排从 l 位置到 r 位置所有士兵引导分身术, 并将分身派往第 y 排, 每个分身前往与当前编号相同的位置。

Output

对于每个 op 为 0 的事件, 在一行输出一个整数代表数出来的士兵数对 998244353 取模后的结果。

Example

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

Problem J. Alice and Bob I

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 1024 megabytes

众所周知, Alice 和 Bob 是很好的朋友, 他们总是喜欢在一起玩有趣的游戏。并且他们每个人都很争强好胜并且高智商, 都想尽力赢下对方。但由于 Alice 是女生, Bob 很有绅士风度, 总是会让 Alice 先手。今天他们遇到的游戏是这样的, 给定 n 张卡片, 每张卡片都有一个数值。他们每个人轮流交替取走一张卡片, 直到取完。

定义手中卡片的价值是所有你取得的卡片数值的和的绝对值, 即假设 Alice 取走的卡为 π_1, \dots, π_k , 那么她所拥有的价值为 $A = \left| \sum_{j=1}^k a_{\pi_j} \right|$, Bob 所拥有的价值为 $B = \left| \sum_{i=1}^n a_i - \sum_{j=1}^k a_{\pi_j} \right|$ 。

Alice 想让她取得的卡片的价值相比于 Bob 所取得的尽可能大, 而 Bob 不想被 Alice 拉开差距, 即 Alice 希望最终 $A - B$ 的值尽可能地大, 而 Bob 希望这个值尽可能地小。

他们都想知道, 在他们都采取最优策略的情况下, 最终 Alice 所取得的卡片价值和 Bob 所取得的卡片价值的差会是多少?

Input

第一行有一个整数 n ($1 \leq n \leq 5000$), 表示卡片的数量。

第二行有 n 个整数 a_1, a_2, \dots, a_n ($-10^9 \leq a_i \leq 10^9$), 中间以空格分隔, 分别表示第 i 张卡片上的数值为 a_i 。

Output

在一行输出一个整数, 代表最终 Alice 所取得的卡片价值和 Bob 所取得的卡片价值的差。

Examples

standard input	standard output
1 100	100
3 4 5 1	2
6 1 2 3 4 5 6	3

Problem K. Alice and Bob II

Input file: standard input
Output file: standard output
Time limit: 15 seconds
Memory limit: 1024 megabytes

众所周知, Alice 和 Bob 是很好的朋友, 他们总是喜欢在一起玩有趣的游戏。并且他们每个人都很争强好胜并且高智商, 都想尽力赢下对方。但由于 Alice 是女生, Bob 很有绅士风度, 总是会让 Alice 先手。今天他们遇到了一些仅由小写字母组成的字符串, 他们觉得完整的字符串不好玩儿, 于是决定玩玩游戏把这些字符串瓜分了。

这个游戏怎么玩呢? 游戏的规则非常的简单呐。

由他们两人轮流交替取走这些字符串中的字母, 直到取完, 轮到谁但他无法行动他就输了, 就要接受严峻的惩罚。

取字母的规则如下, 每次轮到行动的人需要选择其中一个操作:

1. 选择一个非空的字符串, 取走其中任意一个字母;
2. 选择一个非空的字符串, 取走其中任意两个不同的字母。

他们谁都不想接受严峻的惩罚, 于是提前问到了你: 在上述规则下, 如果两人都采取最优策略, 谁能够获胜?

Input

第一行有一个整数 T ($T \leq 10$), 表示测试数据的组数。对于每组数据:

第一行有一个整数 n ($1 \leq n \leq 10$), 表示字符串的个数, 接下来 n 行, 每行有一个仅由小写字母组成字符串 s ($1 \leq |s| \leq 40$)。

Output

对于每组数据, 在一行输出 “Alice” 或者 “Bob” (不含引号), 表示获胜的玩家。

Example

standard input	standard output
4	Alice
1	Bob
aaa	Bob
2	Bob
aaa	
aaa	
1	
aabb	
2	
aabb	
aabb	





















































Problem L. 高低奥马哈扑克

Input file: standard input
Output file: standard output
Time limit: 3 seconds
Memory limit: 1024 megabytes

奥马哈扑克 (Omaha Hold'em) 是一种纸牌游戏, 在许多地方类似于德州扑克 (Texas Hold'em)。即使你玩过德州扑克, 也请仔细阅读下面的内容, 因为部分规则与德州扑克不同。

一群牛逼网友在玩高低奥马哈扑克 (Omaha Hi / Lo) — 一种奥马哈扑克的变体。在游戏的一开始, 每位玩家手上会有 4 张底牌。接下来, 和德州扑克相同, 5 张公共牌会在 3 个阶段发出。这三个阶段分别为: 连续发三张 (翻牌, flop), 单独发一张 (转牌, turn) 和最后一张 (河牌, river)。所有玩家都能看到发出的公共牌, 但只能看到自己的底牌。所有的牌都是从去掉大小王的标准的 52 张扑克牌牌堆中抽取的。一副标准的扑克牌包含了 4 种花色: 梅花 (clubs)♣, 方片 (diamonds)♦, 红心 (hearts)♥, 黑桃 (spades)♠; 每种花色 13 张牌, 包括 Ace (A), King(K), Queen(Q), Jack(J) 和从 2 到 10 的序数牌。每张牌有且仅有一张。

Example set of 52 playing cards; 13 of each suit clubs, diamonds, hearts, and spades

	Ace	2	3	4	5	6	7	8	9	10	Jack	Queen	King
Clubs:													
Diamonds:													
Hearts:													
Spades:													

单张牌的排名从大到小如下: A, K, Q, J, 10, 9, 8, 7, 6, 5, 4, 3, 2。每个玩家需要从自己的底牌中选择恰好 2 张牌, 从公共牌中选择恰好 3 张牌, 组成最好的 5 张牌的牌型。

下面根据牌型大小升序介绍可能的 5 张牌的牌型, 每种牌型中牌需按下文描述的顺序排列。下面描述的规则与常规的德州扑克相同。

- 高牌 (Highcard): 不满足以下任何一种牌型。按 $a_1a_2a_3a_4a_5$ 排列, 满足 $a_1 > a_2 > a_3 > a_4 > a_5$ 。(a_i 表示第 i 张牌的排名)。



- 一对 (Pair): 有两张牌的大小相同。按 $a_1a_2a_3a_4a_5$ 排列, 满足 $a_1 = a_2, a_3 > a_4 > a_5, a_1 \neq a_3, a_1 \neq a_4, a_1 \neq a_5$ 。



- 两对 (Two Pairs): 有两对牌的大小相同。按 $a_1a_2a_3a_4a_5$ 排列, 满足 $a_1 = a_2, a_3 = a_4, a_1 > a_3, a_1 \neq a_5, a_3 \neq a_5$ 。



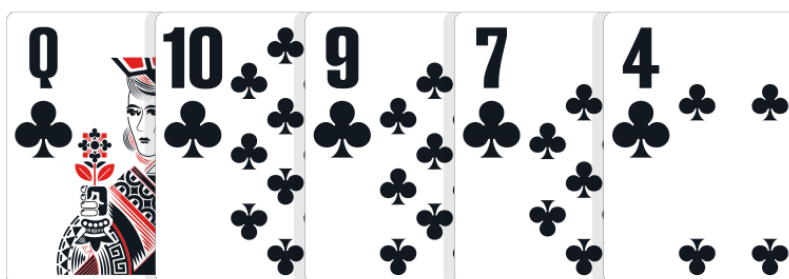
- 三条 (Three of a kind): 有三张牌的大小相同。按 $a_1a_2a_3a_4a_5$ 排列, 满足 $a_1 = a_2 = a_3, a_4 > a_5, a_1 \neq a_4, a_1 \neq a_5$ 。



- 顺子 (Straight): 五张牌的大小连续。按 $a_1a_2a_3a_4a_5$ 排列, 满足对于所有的 $1 \leq i \leq 4$, a_i 恰好比 a_{i+1} 排名高 1。特别地, 如果 a_5 是 Ace, a_4 可以是 2。在这个情况下, Ace 被认为比 2 的排名低。



- 同花 (Flush): 五张牌的花色相同。按 $a_1a_2a_3a_4a_5$ 排列, 满足所有五张牌的花色相同且 $a_1 > a_2 > a_3 > a_4 > a_5$ 。



- 葫芦 (Full house): 三张牌的大小相同, 另外两张牌的大小相同。按 $a_1a_2a_3a_4a_5$ 排列, 满足 $a_1 = a_2 = a_3, a_4 = a_5$ 。



- 四条 (Four of a kind): 四张牌的大小相同。按 $a_1a_2a_3a_4a_5$ 排列, 满足 $a_1 = a_2 = a_3 = a_4$ 。



- 同花顺 (Straight flush): 五张牌的花色相同且大小连续。按 $a_1a_2a_3a_4a_5$ 排列, 满足所有五张牌花色相同, 且对于所有的 $1 \leq i \leq 4$, a_i 恰好比 a_{i+1} 排名高 1。特别地, 如果 a_5 是 Ace, a_4 可以是 2。在这个情况下, Ace 被认为比 2 的排名低。



在比较两手牌时, 先比较两手牌的牌型。例如, 一手牌是四条, 另一手牌是葫芦, 那么四条永远赢葫芦。如果两手牌的牌型一样, 则比较牌的大小。我们需要将牌按上述方式排序, 并一张张地比较它们的大小。即先比较第一张牌的大小, 大的赢。如果第一张牌一样大, 再比较第二张牌的大小, 以此类推。如果所有的五张牌都一样大, 则计为平手。在比较大小时, 不考虑花色。例如 $\clubsuit 5, \diamondsuit 5, \heartsuit 5, \spadesuit 2, \clubsuit 2$ 可以赢 $\diamondsuit 3, \spadesuit 3, \heartsuit 3, \diamondsuit A, \heartsuit A$ 。因为它们都是三条, 我们将先比较相同的三个的大小。

下面的规则是德州扑克中没有的, 请仔细阅读。

在高低奥马哈扑克中, 玩家还需要使用小于等于 8 的牌进行比低牌。每个玩家需要从自己的底牌中选择恰好 2 张牌, 从公共牌中选择恰好 3 张牌, 组成最好的 5 张不成对的牌型。如果无法使用小于等于 8 的牌组成不成对的牌型, 则该玩家不参与比低牌。

注意: 比高和比低是相互独立的, 选择可以不同。在比低牌时的单张牌的排名从大到小为: 8, 7, 6, 5, 4, 3, 2, A。按 a_1, a_2, a_3, a_4, a_5 排列, 满足 $a_1 > a_2 > a_3 > a_4 > a_5$ 。同花和顺子不影响比小。

在比较两手牌时, 先比较第一张牌的大小, 小的赢。如果第一张牌一样大, 再比较第二张牌的大小, 以此类推。如果所有的五张牌都一样大, 则计为平手。因此在比小时最好的牌型是: 5,4,3,2,A, 最差的牌型是: 8,7,6,5,4。

当游戏进行到最后的比大小阶段, 当前的底池为 p 个筹码。如果没有玩家能参与比低牌, 则奖池全部给在比高中获胜的玩家。如果有至少一名玩家可以比低, 则奖池的一半 $\lceil \frac{p}{2} \rceil$ 分给比高中获胜的玩家, 奖池的另一半 $\lfloor \frac{p}{2} \rfloor$ 分给比低中获胜的玩家。

如果在比高或比低时打平需要平分底池但无法平分, 多余的筹码给位置靠前的玩家。即如果底池为 p , 给 n 个人平分, 则位置最靠前的玩家赢得剩余的 $p - \lfloor \frac{p}{n} \rfloor$ 个筹码。

现在, 其它玩家已经弃牌, Alice 和 Bob 进行到了最后的比大小阶段。Alice 位置靠前, 底池总筹码数为 p 。已知两人的底牌和 5 张公共牌。你需要分别计算两人最后得到的筹码数。

Input

第一行有一个整数 T ($T \leq 500$), 表示测试数据的个数。对于每组数据:

第一行有一个整数 p ($10 \leq p \leq 1000$), 表示底池的大小。第二行为 Alice 的底牌, 第三行为 Bob 的底牌, 第四行为公共牌。

牌之间用空格分隔, 每张牌由两个字符组成, 分别表示大小和花色。大小用 2,..., 9, T, J, Q, K, A 表示 (T 代表序数牌 10), 花色用 C, D, H, S 表示。

Output

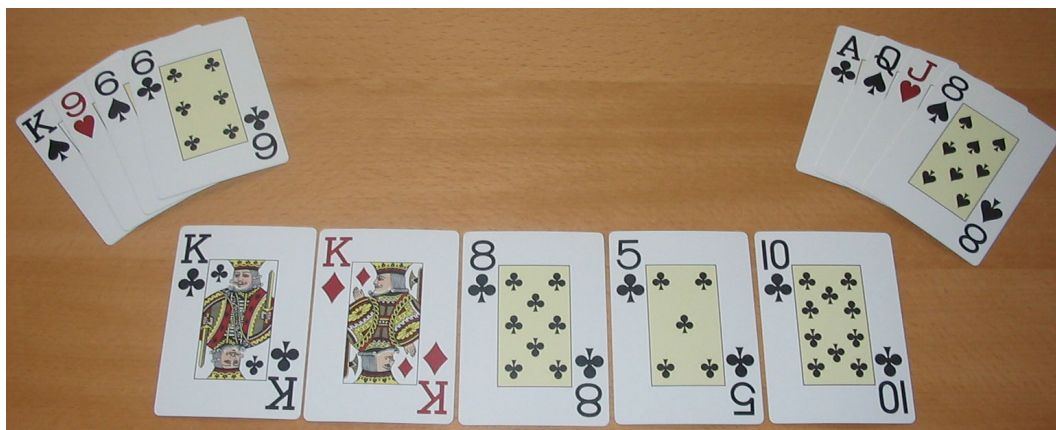
对于每组数据, 在一行输出两个整数, 中间以空格分隔, 分别表示 Alice 和 Bob 赢得的筹码数。

Example

standard input	standard output
2	233 0
233	116 117
KS 9H 6S 6C	
AC QS JH 8S	
KC KD 8C 5C TC	
233	
AS 2C 4H KH	
AC 2D 5D 5C	
2S 3H JH JD 5H	

Note

第一个样例中，Alice 在比高中的牌型为 $\clubsuit K, \diamondsuit K, \spadesuit K, \clubsuit T, \heartsuit 9$ （三条），Bob 在比高中的牌型为 $\clubsuit K, \diamondsuit K, \clubsuit 8, \spadesuit 8, \clubsuit A$ （两对）。因此 Alice 在比高中取胜。两人均无法比低。所以 Alice 赢得所有底池。



在第二个样例中，Alice 以 $\spadesuit A, \spadesuit 2, \heartsuit 3, \heartsuit 4, \heartsuit 5$ 的牌型在比低中取胜，Bob 以 $\clubsuit 5, \diamondsuit 5, \heartsuit 5, \diamondsuit J, \heartsuit J$ （葫芦）的牌型在比高中取胜。因此两人将平分底池。

