

1001 小 Q 系列故事——大笨钟

Time Limit: 0.2 Seconds Memory Limit: 65536K

饱尝情感苦恼的小 Q 本打算隐居一段时间，但仅仅在 3 月 25 号一天没有出现，就有很多朋友想念他，所以，他今天决定再出来一次，正式和大家做个告别。

小 Q 近来睡眠情况很差——晚上睡不着，早上又起不来！以前总是全勤的他这个月甚至迟到了好几次，虽然主管没说什么，但是他自己清楚此事的严重性。

爱情没有了，小 Q 不想再失去面包，于是他决定买个闹钟，为求质量可靠，特意挑了个最贵的原装进口货！但是，正所谓屋漏偏逢连夜雨、人倒霉的时候喝凉水都塞牙，小 Q 新买的这个进口闹钟竟然每分钟总是比正确时间慢那么几秒！

可怜的小 Q 愤愤然道：“真是一个大笨钟！”

但是为了充分利用大笨钟，小 Q 还是尽力想办法搞清楚它的性能，希望能将其当作正常时钟使用。

小 Q 从如下三方面去研究大笨钟：

- 1、 假设正常时间走了 t 分钟，计算大笨钟走的时间；
- 2、 假设大笨钟走了 t 分钟，计算正常时间走的时间；
- 3、 小 Q 将大笨钟与当前时间调成正确时间，此时记为第 0 次，计算第 k 次大笨钟显示的时间与正确时间相同需要的时间（大笨钟的结构和普通时钟相同，即分为 12 大格，60 小格）。

Input

输入数据第一行是一个正整数 T ，表示总共有 T 组测试数据；

接下来的每组数据首先输入正整数 x ，表示大笨钟每分钟比正常时钟慢 x 秒；

接下来一行是一个正整数 Q ，表示共有 Q 次询问；

接下来 Q 行，每行首先输入询问方式（1、2 或 3，对应小 Q 研究大笨钟的三方面），如果输入 1 或 2，接下来输入正整数 t ，如果输入 3，接下来输入正整数 k （ t 和 k 的含义见题目）。

[Technical Specification]

$T \leq 100$

$x < 60$

$Q \leq 100$

$t \leq 10000$

$k \leq 10$

Output

请输出要计算的答案，以秒为单位，保留两位小数，每次查询输出一行（参见 Sample）。

Sample Input

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

Sample Output

118.00

122.03

2592000.00

Hint:

小 Q 最后还想说句话：“进口货未必可靠，咱们还是支持国货吧！”

1002 吉哥系列故事——乾坤大挪移

Time Limit: 1.0 Seconds Memory Limit: 65536K

只有进入本次马拉松复赛,你才有机会知道一个秘密:吉哥的真名叫基哥,江湖人称“叭叭哥”。

叭叭哥除了编程,还一直有个武侠梦,他最喜欢的人物是金庸小说《倚天屠龙记》中的张无忌,不仅有美人环绕,而且有一身的好武功,尤其是那神秘的乾坤大挪移,让他梦寐以求:

“乾坤大挪移乃在颠倒一刚一柔、一阴一阳的乾坤二气,随意而行,不用心而无心用,所谓至我逍遥游,以纯阳之身,和纯阴之体,合练双修,不动身,只用意,意动身守.....”

但是,梦毕竟只是梦,平时在编程的空闲时间,叭叭哥也最多只能上网玩一下名为“乾坤大挪移”的游戏聊以自慰而已。

这个“乾坤大挪移”游戏是在 3×3 的方格中进行。

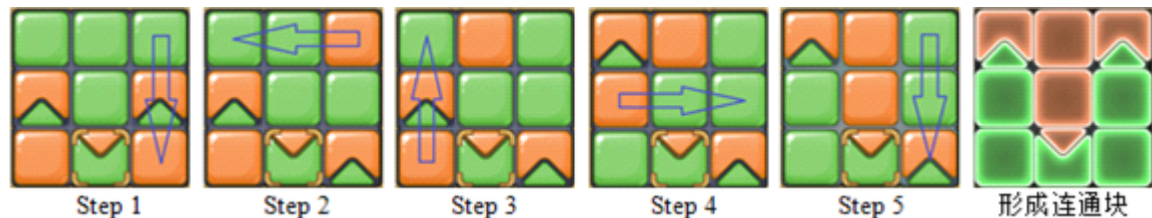
游戏的目标是通过移动,让相同颜色的块形成一个连通块(相邻是指两个块有边相邻,角相邻不算)。

移动规则如下:选择一行(列),向左右(上下)移动一格,方格从一边划出,则从对应的另外一边划入,像履带一样。

如选择第一行向右边移动,最右边的那格会移动到最左边。

游戏中还有一些方格被固定住,这些方格没办法移动(如下图的第三行第二列)。

下图是游戏的一个演示(即 Case 1):



假设现在告诉你初始状态,请问你最少需要几步才能达到目标?

Input

第一行一个整数 T 代表接下去有 T 组数据;

每组数据由 3×3 的模块组成,每个模块表示的小正方形是由上下左右四个小三角形组成;每个模块有 5 个字符,前四个字符分别表示组成正方形的上下左右四个小三角形的颜色,第五个字符表示该格子能否移动 (0 表示能移动,1 表示不能移动)。

[Technical Specification]

$0 < T < 100$

代表颜色的字符一定是 RGB0 的其中一个;

代表能否移动的字符一定是 0 或者 1;

Output

首先输出 case 数,接着输出最小的移动步数使得游戏达到目标状态(见 sample)。

数据保证有解。

Sample Input

2

```
GGGG0 GGGG0 GGGG0
OG000 GGGG0 OG000
00000 OGGG1 00000
RRRR0 00000 00000
00000 00000 00000
00000 00000 RRRR0
```

Sample Output

Case #1: 5

Case #2: 2

1003 湫秋系列故事——安排座位

Time Limit: 10.0 Seconds Memory Limit: 65536K

为了给腾讯公司找到更多优秀的人才，HR 湫秋最近去某高校组织了一次针对该校所有系的聚会，邀请了每个系的一些优秀学生来参加。

作为组织者，湫秋要安排他们的座位。这并不是一件很简单的事情，因为只有一排位置，并且位置总数恰好等于参加聚会的人数。为了促进交流，两个来自相同系的同学不可以座位相邻。湫秋现在希望知道有多少种不同的合理安排座位的方法（任意两个合理的安排方法，只要有一个位置的同学不同，都被认为是不同的）。

Input

输入第一行为 T ，表示有 T 组测试数据。

每组数据一个 N 开始，表示一共有多少个系。下面的一行包含 N 个整数 A_i ，表示每个系的到场人数。

[Technical Specification]

1. $1 \leq T \leq 47$
2. $1 \leq N, A_i \leq 47$
3. $1 \leq \text{Sum}(A_i) \leq 447$

Output

对每组数据，先输出为第几组数据，然后输出结果。由于结果可能很大，输出对 1 000 000 007 取余后的结果。

Sample Input

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

Sample Output

```
Case 1: 2
Case 2: 0
Case 3: 120
```

1004 威威猫系列故事——晒被子

Time Limit: 1.0 Seconds Memory Limit: 65536K

因为马拉松初赛中吃鸡腿的题目让不少人抱憾而归，威威猫一直觉得愧对大家，这几天他悄悄搬到直角坐标系里去住了。

生活还要继续，太阳也照常升起，今天，威威猫在第一象限晒了 N 条矩形的被子，被子的每条边都和坐标轴平行，不同被子的某些部分可能会叠在一起。这时候，在原点处突然发了场洪水，时间 t 的时候，洪水会蔓延到 (t, t) ，即左下角为 $(0, 0)$ ，右上角为 (t, t) 的矩形内都有水。

悲剧的威威猫想知道，在时间 $t_1, t_2, t_3 \dots t_x$ 的时候，他有多少面积的被子是湿的？

Input

输入数据首先包含一个正整数 T ，表示有 T 组测试数据；

每组数据的第一行首先是一个整数 N ，表示有 N 条被子；

接下来 N 行，每行包含四个整数 x_1, y_1, x_2, y_2 ，代表一条被子的左下角和右上角的坐标；

然后接下来一行输入一个整数 x ，表示有 x 次询问；

再接下来 x 行，输入 x 个严格单调递增的整数，每行一个，表示威威猫想知道的时间 t_i 。

[Technical Specification]

$T \leq 5$

$0 < N \leq 20000$

$1 \leq x_1 < x_2 \leq 200000$

$1 \leq y_1 < y_2 \leq 200000$

$1 \leq x \leq 20000$

$1 \leq t_i \leq 200000 \quad (1 \leq i \leq x)$

Output

对于每次询问，请计算并输出 t_i 时有多少面积的被子是湿的，每个输出占一行。

Sample input

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

Sample output

```
0
1
5
8
8
```

1005 郑厂长系列故事——新闻净化

Time Limit: 2.0 Seconds Memory Limit: 65536K

郑厂长不是正厂长
也不是副厂长
他根本就不是厂长
他曾经是腾讯公司的码农
一个业余时间喜欢下棋的码农
但现在
他神秘失踪了.....

众所周知，在太平洋某岛国，新闻审查是很严格的，而郑厂长的失踪就与该国的新闻审查有关。别担心，他不是喝茶去，而且被秘密邀请承担该国净化新闻的工作了。

这份工作的主要内容是这样的，对于一篇即将发表的新闻稿，郑厂长需要对它做最后的订正工作：只通过删除一些字母，使其符合“相关要求”。这些要求有，一些词语必须作为子串出现，一些词语必须不能作为子串出现，另一些词语作为子串出现有相应的分数加成，需要注意的是，这个加成分数可能是负的。

郑厂长要删除最少字母使文章符合要求，并让加成分之和尽可能高。如果一个带有加分的单词出现了多次，结果也计算多次。

Input

输入第一行为 T ，表示有 T 组测试数据。

每组数据一个 N 开始，表示有 N 个在“相关要求”中的单词。为了简化输入，给每个单词都指定一个加成分，加成分为 999 的，表示“必须作为子串出现”的，加成分为 -999 的，表示“必须不能作为子串出现”的。

接下来的 N 行里，每行有一个单词 S_i 和对应的加成分 G_i 。最后一行是原稿内容 S_{ori} 。

[Technical Specification]

1. $1 \leq T \leq 47$
2. $1 \leq N \leq 100$
3. $-999 \leq G_i \leq 999$
4. G_i 为 999 的单词数目不大于 8
5. G_i 为 -999 的单词数目不大于 8
6. $1 \leq |S_i| \leq 16, 1 \leq |S_{ori}| \leq 100, |S|$ 表示字符串 S 的长度
7. S_i 与 S_{ori} 只由小写字母 'a'-'z' 组成，不会出现相同的

Output

对每组数据，先输出为第几组数据，如果无法满足要求，输出“Banned”，否则输出两个数字，删除的最少字母和最大分数，用一个空格隔开。

Sample Input

```
3
2
he 999
sh -999
she
2
she 999
he -999
shelovesyou
4
ab 999
cd -999
abd 1
abc -1
abcdefg
```

Sample Output

```
Case 1: 1 0
Case 2: Banned
Case 3: 1 1
```

Hint

对于第三组样例，“abdefg”和“abcefg”都符合要求，但前者的加成分比后者高。