

# 第三届环球杯



## 第 16 赛段：南京

2024 年 11 月 9-10 日

这套问题集应包含 13 个问题，共 20 页（有编号）。

### 根据



国际大学生程序设计竞赛（ICPC）

### 主办



### 筹备







## 问题 A. 嘿，你见过我的袋鼠吗？

时间限制                      3 秒 内存限制  
： 128 兆字节                128 兆字节

请注意这个问题的内存限制很不寻常。

继2018年、2019年、2020年、2021年、2022年和2023年的成功举办后，南京航空航天大学将连续第七次承办国际大学生程序设计竞赛（ICPC）南京赛区的比赛。

2018年和2019年，清华大学的*Power of Two*队和*Three Hold Two*队获得冠军。2020年、2021年、2022年，北京大学*倒十字*队获得三连冠。2023 年，来自北京大学的另一支队伍 *Reborn as a Vegetable Dog* 获得冠军。同时，他们还获得了第 46 届 ICPC 世界冠军，时隔 13 年为欧共体地区夺回了冠军奖杯！

今年，约有 335 支队伍参赛。最多将颁发 33 枚金牌、66 枚银牌和 99 枚铜牌（注：这些数字仅供参考）。我们期待看到参赛者的出色表现！我们还要对所有工作人员和志愿者为本次比赛付出的辛勤劳动表示感谢。感谢你们为本次比赛做出的巨大贡献！



在 2023 年国际马联亚洲赛南京地区比赛中拍摄的照片

在 2018 年的竞赛中，问题 K "**袋鼠之谜**"要求参赛者游戏构建一个操作序列：

谜题是一个  $n$  行  $m$  列 ( $1 \leq n, m \leq 20$ ) 的网格，网格中站着几只（至少 2 只）袋鼠。玩家的目标是控制它们走到一起。有些格子里有墙，袋鼠不能进入有墙的格子。其他牢房则是空的。袋鼠可以在四个方向上从一个空格移动到相邻的空格：上、下、左、右。

开始时，每个空格中都有一只袋鼠，玩家可以按键盘上的 U、D、L、R 按钮来控制袋鼠。袋鼠会根据您按下的按钮同时移动。

参赛者需要构建一个最多由  $5 \times 10^4$  步组成的操作序列，其中只包括 U、D、L、R，以实现目标。



在 2020 年的竞赛中，问题 A “**啊，昨日重现**”要求参赛者构建一个输入地图，以破解前述问题的以下代码：

```
#include <bits/stdc++.h>
using namespace std;
string s = "UDLR";
int main()
{
    srand(time(NULL));
    for (int i = 1; i <= 50000; i++) putchar(s[rand() % 4]);
    返回 0;
}
```

此外，在 2021 年竞赛（问题 A，**哎呀，又是昨天的两倍**）、2022 年竞赛（问题 A，**停，昨天请不要再来了**）和 2023 年竞赛（问题 A，**酷，又是昨天的四倍**）中，我们每年都有一个与袋鼠有关的问题！我们本想把所有这些问题都介绍给大家，但如果每年都这样做的话，2024 年竞赛中的一个问题可能就会有 500 页的说明。因此，我们这次就不介绍了。此外，您可能已经在练习赛中见过它们了。

现在，在 2024 年的比赛中，正如大家所期待的那样，袋鼠问题再次出现！我们不知道问题设置者为何对袋鼠如此着迷，但问题如下：

给你一个  $n$  行  $m$  列的网格。有些单元格有墙，袋鼠不能进入有墙的单元格。其他单元格是空的，每个单元格都有一只袋鼠。袋鼠可以向上、下、左、右四个方向从一个空单元格移动到相邻的空单元格。

您可以按键盘上的 U、D、L、R 按钮来控制袋鼠。袋鼠会根据您按下的按钮同时移动。具体来说，对于位于第  $i$  行第  $j$  列单元格中的袋鼠，用  $(i, j)$  表示：

1. 按钮 U：如果  $i > 1$  且  $(i - 1, j)$  不是墙壁，它将移动到  $(i - 1, j)$ 。否则，它将停留在同一小格中。
2. 按钮 D：如果  $i < n$  且  $(i + 1, j)$  不是墙壁，它将移动到  $(i + 1, j)$ 。否则，它将停留在同一单元格中。
3. 按钮 L：如果  $j > 1$  且  $(i, j - 1)$  不是墙壁，它将移动到  $(i, j - 1)$ 。否则，它将停留在同一小格中。
4. 按钮 R：如果  $j < m$  且  $(i, j + 1)$  不是墙壁，它将移动到  $(i, j + 1)$ 。否则，它将停留在同一个单元格中。

给你一个操作序列  $s_1 s_2 \dots s_k$ ，序列中只有字符 "U"、"D"、"L" 和 "R"。操作将根据序列无限执行。具体来说，如果  $1 \leq t \leq k$ ，则第  $t$  次操作为  $s_t$ ；否则，如果  $t > k$ ，则第  $t$  次操作与第  $(t-k)$ -次操作相同。对于每个  $1 \leq i \leq n \times m$ ，求最小整数  $v_i$ ，使得执行  $v_i$  操作后，最多有  $i$  个单元格包含袋鼠。



## 输入

每个测试文件中只有一个测试用例。

第一行包含三个整数  $n$ 、 $m$  和  $k$  ( $1 \leq n, m \leq 2 \times 10^5$ ,  $1 \leq n \times m \leq 2 \times 10^5$ ,  $1 \leq k \leq 200$ )，表示网格的行数和列数，以及操作序列的长度。

第二行包含一个字符串  $s_1 s_2 \dots s_k$  ( $s_i \in \{'U', 'D', 'L', 'R'\}$ )，表示操作顺序。

对于以下  $n$  行，第  $i$  行包含一个二进制字符串  $a_{i,1} a_{i,2} \dots a_{i,m}$  ( $a_{i,j} \in \{'0', '1'\}$ )。如果  $a_{i,j} = '1'$ ，则单元格  $(i, j)$  为空；否则，如果  $a_{i,j} = '0'$ ，则单元格  $(i, j)$  被封锁，无法输入。可以保证网格中至少有一个空单元格。



输出

输出  $n \times m$  行，其中第  $i$  行包含一个整数  $v_i$ ，表示为使最多  $i$  个单元格包含袋鼠所需的最少操作数。如果不可能，就在这一行输出-1。

实例

标准输入	标准输出
3 3 6 ULDDRR 010 111 010	-1 4 2 1 0 0 0 0 0
3 3 6 ULDDRR 010 111 011	7 4 2 1 1 0 0 0 0
1 5 1 R 11111	4 3 2 1 0



## 问题 B. 生日礼物

时间限制                      1 秒 内存限制  
1024 兆字节

格莱美的生日快到了，她从朋友那里得到一个序列  $A$  作为礼物。这个序列只有 0、1 和 2。格莱美认为这个数列太长了，因此她决定修改  $A$ ，使它变得更短。

从形式上看，格莱美可以执行任意数量的操作。每次她都可以从以下三种操作中选择一种进行：

- 将任意 2 变为 0 或 1。
- 选择两个相邻的 0，擦除它们，然后将其余部分连接起来。
- 选择两个相邻的 1，擦除它们，然后将其余部分连接起来。

计算格莱美能得到的最小序列长度。

### 输入

有多个测试用例。输入的第一行包含一个整数  $T$ ，表示测试用例的数量。对于每个测试用例

第一行也是唯一一行包含长度为  $n$  的字符串 ( $1 \leq n \leq 2 \times 10^5$ )，由数字 0、1 和 2 组成，表示初始序列  $A$ 。

保证所有测试用例的  $n$  之和不超过  $5 \times 10^5$ 。

### 输出

对于每个测试用例，输出一行，其中包含一个整数，表示 Grammy 能得到的最小序列长度。

### 示例

标准输入	标准输出
5	3
0110101	4
01020102	0
0000021111	6
1012121010	0
0100202010	



## 问题 c. 拓扑

时间限制                      2 秒 内存限制  
1024 兆字节

给你一棵由  $n$  个顶点组成、根植于顶点 1 的树。保证每个顶点的索引都小于其所有子顶点的索引。这棵树的拓扑顺序是一个排列  $p_1, p_2, \dots, p_n$ ，它满足以下约束条件：对于所有  $1 \leq i < j \leq n$ ，顶点  $p_j$  不是顶点  $p_i$  的父顶点。

对于每个  $1 \leq i \leq n$ ，计算满足  $p_i = i$  的给定树的拓扑阶数，模数为 998 244 353。

### 输入

每个测试文件中只有一个测试用例。

第一行包含一个整数  $n$  ( $2 \leq n \leq 5\,000$ )，表示树的顶点数。

第二行包含  $(n - 1)$  个整数  $f_2, f_3, \dots, f_n$  ( $1 \leq f_i < i$ )，其中  $f_i$  是顶点  $i$  的父节点。

### 输出

输出一行，包含  $n$  个整数  $a_1, a_2, \dots, a_n$ ，用空格隔开，其中  $a_i$  是满足  $p_i = i$  的拓扑阶数，模数为 998 244 353。

### 实例

标准输入	标准输出
4 1 1 2	3 2 1 2
9 1 1 2 2 3 3 4 5	672 420 180 160 152 108 120 170 210

### 备注

对于第一个样本测试用例，树的所有拓扑顺序分别为  $\{1, 2, 3, 4\}$ 、 $\{1, 3, 2, 4\}$  和  $\{1, 2, 4, 3\}$ 。其中有 3 个满足  $p_1 = 1$ ，2 个满足  $p_2 = 2$ ，1 个满足  $p_3 = 3$ ，2 个满足  $p_4 = 4$ 。





## 问题 D. 脚趾--嗒嗒嗒

时间限制

1 秒 内存限制

1024 兆字节

爱丽丝和鲍勃在有 3 行 3 列的  $n$  个棋盘上玩 "嘟-嗒-嘟" 游戏。棋盘上的一些单元格最初是空的，而其他单元格中已经有了一些标记。爱丽丝先走，两人轮流选择一块棋盘，并把自己的标记放入棋盘上的空格中。爱丽丝的标记是 "x"，鲍勃的标记是 "o"。

每位玩家必须确保在他/她的移动后，任何棋盘上的任何行、列或对角线上都没有三个相同的标记。轮到自己下棋时，下棋无效的一方输，另一方赢。

给定  $n$  个棋盘的初始状态，假设两位棋手都以最佳方式下棋，您需要确定谁能获胜。

### 输入

有多个测试用例。输入的第一行包含一个整数  $T$ ，表示测试用例的数量。对于每个测试用例

第一行包含一个整数  $n$  ( $1 \leq n \leq 10^5$ )，表示棋局的棋盘数。然后是大小为  $3 \times 3$  的  $n$  个棋盘

。对于每块棋盘

- 如果不是第一块棋盘，则首先会出现空行。
- 在下面的三行中，第  $i$  行包含一个长度为 3 的字符串  $s_{i,1}, s_{i,2}, s_{i,3}$ ，由字符 'x'、'o' 和 '.' 组成，描述了一块大小为  $3 \times 3$  的棋盘。设  $(i, j)$  为第  $i$  行第  $j$  列上的单元格。如果  $s_{i,j} = 'x'$ ，则单元格  $(i, j)$  包含一个标记 'x'；如果  $s_{i,j} = 'o'$ ，则单元格  $(i, j)$  包含一个标记 'o'；如果  $s_{i,j} = '.'$ ，则单元格  $(i, j)$  为空。

保证任何棋盘的任何行、列或对角线上都没有三个相同的标记。同时保证所有测试案例的  $n$  之和不超过 10。<sup>5</sup>

### 输出

对于每个测试案例，如果 Alice 赢得游戏，则输出 Alice；如果 Bob 赢得游戏，则输出 Bob。



示例

标准输入	标准输出
4	爱丽丝
1	爱丽丝
...	
...	鲍勃 鲍
...	
1	勃
...	
oo.	
oo.	
2	
...	
oo.	
oo.	
...	
xx.	
xx.	
2	
..x	
xo.	
...	
xo.	
o..	
.x.	



## 问题 E.左移 3

时间限制 1 秒 内存限制  
1024 兆字节

给定长度为  $n$  的字符串  $S = s_0s_1 \dots s_{n-1}$ ，最多可以将  $S$  向左平移  $k$  次（包括 0 次）。计算操作后字符串中包含的 "南京" 子串的最大数目。

更正式地说，让  $f(S, d)$  是将  $S$  向左移动  $d$  次得到的字符串。即， $f(S, d) = s_{(d+0) \bmod n} s_{(d+1) \bmod n} \dots s_{(d+n-1) \bmod n}$ 。设  $g(f(S, d), l, r) = s_{(d+l) \bmod n} s_{(d+l+1) \bmod n} \dots s_{(d+r) \bmod n}$ 。设  $h(d)$  为整数对  $(l, r)$  的个数，使得  $0 \leq l \leq r < n$ ，且  $g(f(S, d), l, r) = \text{nanjing}$ 。找出一个整数  $d$ ，使得  $0 \leq d \leq k$ ，从而最大化  $h(d)$ ，并输出这个最大化值。

### 输入

有多个测试用例。输入的第一行包含一个整数  $T$ ，表示测试用例的数量。对于每个测试用例  
第一行包含两个整数  $n$  和  $k$  ( $1 \leq n \leq 2 \times 10^5$ ,  $0 \leq k \leq 10^9$ )，表示字符串的长度和可以执行的最大左移次数。  
第二行包含长度为  $n$  的字符串  $s_0s_1 \dots s_{n-1}$ 。该字符串由小写英文字母组成。  
保证所有测试用例的  $n$  之和不超过  $5 \times 10^5$ 。

### 输出

对于每个测试用例，输出一行，其中包含一个整数，表示字符串中包含的 "南京" 子串的最大数量。

### 示例

标准输入	标准输出
4	2
21 10	1
京ICP备050321号-1 京公网安备050321号-1	3
21 0	0
京ICP备050321号-1 京公网安备050321号-1	
21 3	
南京南京南京	
4 100	
icpc	

### 备注

对于第一个示例测试用例，我们可以将字符串向左移动 6 次，得到字符串 "pcnanjingsuananijingic"。其中有两个 "nanjing" 子串。



对于第二个样本测试用例，由于  $k = 0$ ，我们无法进行任何左移操作。原始字符串中有一个 "南京 " 子串。



问题 F. 地铁

时间限制 2.5 秒 内存限制 1024 兆字节

在 Pigeland，地铁系统相当先进。地铁线  $i$  依次经过站点  $x_{i,1}, x_{i,2}, \dots, x_{i,p_i}$ ，其中  $x_{i,j}$  是地铁线  $i$  经过的第  $j$  个站点。地铁线  $i$  从站点  $x_{i,j}$  到站点  $x_{i,j+1}$  需要  $w_{i,j}$  个单位的时间。

当多条线路在同一站点交汇时，乘客可以在线路之间换乘。如果乘客在  $x$  号线的某个站点，而  $y$  号线也经过该站点，那么他/她可以花费  $a_y \times b_x$  个单位的时间从  $x$  号线换乘到  $y$  号线，其中  $a_y$  和  $b_x$  是  $y$  号线和  $x$  号线的给定系数。换乘后，乘客仍在同一站点，但在  $y$  号线上。

从站点 1 开始。在所有  $2 \leq s \leq n$  的情况下，求到达站点  $s$  所需的最短时间。保证从站点 1 可以到达所有站点。

输入

每个测试文件中只有一个测试用例。

第一行包含两个整数  $n$  和  $k$  ( $2 \leq n \leq 2 \times 10^5, 1 \leq k \leq 2 \times 10^5$ )，表示站点数和地铁线路数。

第二行包含  $k$  个整数  $a_1, a_2, \dots, a_k$  ( $1 \leq a_i \leq 10^6$ )。第

三行包含  $k$  个整数  $b_1, b_2, \dots, b_k$  ( $1 \leq b_i \leq 10^6$ )。

对于下面的  $k$  行，第  $i$  行首先包含一个整数  $p_i$  ( $2 \leq p_i \leq n$ )，表示第  $i$  行经过的站点数。然后是  $(2p_i - 1)$  个整数  $x_{i,1}, w_{i,1}, x_{i,2}, \dots, x_{i,p_i-1}, w_{i,p_i-1}, x_{i,p_i}$  遵循 ( $1 \leq x_{i,j} \leq n, 1 \leq w_{i,j} \leq 10^9$ )，其中  $x_{i,j}$  是第  $i$  条线路经过的第  $j$  个站点， $w_{i,j}$  是第  $i$  条线路上从站点  $x_{i,j}$  到站点  $x_{i,j+1}$  的旅行时间。地铁线路经过的站点是不同的。

可以保证  $\sum_{i=1}^k (p_i - 1) \leq 2 \times 10^5$

输出

输出一行包含  $(n - 1)$  个整数  $d_2, d_3, \dots, d_n$ ，中间用空格隔开，其中  $d_i$  是站点 1 到站点  $i$  的最小时间成本。

实例

标准输入	标准输出
6 3 1 5 1 5 5 1 3 1 2 2 3 3 3 5 1 2 1 4 3 3 4 5 4 6	2 5 21 14 18
6 3 1 5 1	2 31 43 37 136



5 5 1	
5 1 2 2 100 3 100 6 1 4	
5 1 100 2 4 3 100 5 1 4	
2 3 1 5	



## 问题 G. 二进制树

时间限制

1 秒 内存限制

1024 兆字节

这是一个互动问题。

给定一棵有  $n$  个顶点的二叉树，你的任务是在树中找到一个特殊的顶点  $s$ ，该顶点最多有  $p = \lfloor \log_2 n \rfloor$  查询。也就是说， $p$  是  $2^p \leq n$  的最大整数。

每个查询由两个不同的顶点  $u$  和  $v$  组成。交互者将输出一个整数  $t$  ( $0 \leq t \leq 2$ ) 作为答案。假设  $d(a, b)$  是顶点  $a$  到顶点  $b$  的简单路径上的边数。

- 如果  $t = 0$ ，则顶点  $u$  更靠近特殊顶点。也就是说， $d(u, s) < d(v, s)$ 。
- 如果  $t = 1$ ，那么  $u$  和  $v$  到特殊顶点的距离相同。也就是说， $d(u, s) = d(v, s)$ 。
- 如果  $t = 2$ ，则顶点  $v$  更靠近特殊顶点。也就是说， $d(u, s) > d(v, s)$ 。

请注意，交互器是自适应的，这意味着每个测试用例的答案都不是预先确定的。只要其答案与之前的查询和答案不冲突，交互式程序就能根据您的查询确定特殊顶点。

### 输入

有多个测试用例。输入的第一行包含一个整数  $T$ ，表示测试用例的数量。对于每个测试用例

第一行包含一个整数  $n$  ( $2 \leq n \leq 10^5$ )，表示二叉树的顶点数。

对于下面的  $n$  行，第  $i$  行包含两个整数  $x_i$  和  $y_i$  ( $0 \leq x_i, y_i \leq n$ )，表示第  $i$  个顶点的左子顶点和右子顶点。如果  $x_i = 0$ ，则第  $i$  个顶点没有左子顶点；如果  $y_i = 0$ ，则第  $i$  个顶点没有右子顶点。

保证所有测试用例的  $n$  之和不超过  $2 \times 10^5$ 。

### 互动协议

要进行查询，只需输出一行。首先输出"?"，后面跟一个空格，然后打印两个不同的整数  $u$  和  $v$  ( $1 \leq u, v \leq n$ )，中间用空格隔开。刷新输出后，程序应读取一个整数  $t$ ，表示查询的答案。

如果您想猜测特殊顶点，请输出一行。首先输出!之后的空格，然后打印整数  $s$  ( $1 \leq s \leq n$ )，表示特殊顶点。刷新输出后，程序应继续处理下一个测试用例，如果没有测试用例，则立即退出。请注意，您的猜测不算是一次查询。

要清除输出，可以使用

- 在 C 和 C++ 中，`fflush(stdout)` (如果使用 `printf`) 或 `cout.flush()` (如果使用 `cout`)。



- Java 中的 `System.out.flush()`。
- 在 Python 中使用 `stdout.flush()`。





示例

标准输入	标准输出
2	
5	
0 0	
1 5	
2 4	
0 0	
0 0	
	?5 1
1	
	?1 4
0	
	!2
2	
0 2	
0 0	
	?2 1
2	
	!1



问题 H. 边境跳跃 2

时间限制 2 秒 内存限制 1024 兆字节

给你一个由小写英文字母组成的字符串  $S$ 。您需要对  $S$  进行一些操作，直到它变成空字符串。每次您都可以执行以下三种操作之一：

- 1. 删除  $S$  的第一个字符。
- 2. 删除  $S$  的最后一个字符。
- 3. 从  $S$  中选择一个好的子串  $S^r$ ，然后用  $S^r$  替换  $S$ 。

当且仅当  $S^r \neq S$ ， $S^r$  是  $S$  的前缀，且  $S^r$  的反向是  $S$  的后缀时，非空字符串  $S^r$  称为字符串  $S$  的好子串。长度为  $k$  的字符串  $p_1 p_2 \cdots p_k$  的反向是长度为  $k$  的另一个字符串  $p_k p_{k-1} \cdots p_1$ 。

您最多可以执行多少次第 3 类操作？

输入

有多个测试用例。输入的第一行包含一个整数  $T$ ，表示测试用例的数量。对于每个测试用例第一行也是唯一一行包含一个字符串  $S$  ( $1 \leq |S| \leq 10^5$ )，由小写英文字母组成。保证所有测试用例的  $|S|$  总和不超过  $2 \times 10^5$ 。

输出

对于每个测试用例，输出一行，其中包含一个整数，表示类型 3 的最大数量您可以执行的操作。

示例

标准输入	标准输出
3	3
aaaa	4
阿爸阿妈	0
xy	

备注

第一个示例测试用例：aaaa  $\xrightarrow{\text{op.3}}$  aaa  $\xrightarrow{\text{op.3}}$  aa  $\xrightarrow{\text{op.3}}$  a  $\xrightarrow{\text{op.2}}$   $\emptyset$ 。

第二个示例测试用例：abbaabba  $\xrightarrow{\text{op.1}}$  abbaabb  $\xrightarrow{\text{op.3}}$  bbaabb  $\xrightarrow{\text{op.3}}$  bbaab  $\xrightarrow{\text{op.1}}$  baab  $\xrightarrow{\text{op.3}}$  咿  $\xrightarrow{\text{op.1}}$  aa  $\xrightarrow{\text{op.3}}$  a  $\xrightarrow{\text{op.1}}$   $\emptyset$ 。



问题 I. 宾果

时间限制 1 秒 内存限制 1024 兆字节

给定两个整数  $n$ 、 $m$  和一个长度为  $n \times m$  的整数序列  $a_1, a_2, \dots, a_{nm}$ ，我们将在一个  $n$  行  $m$  列的网格中填入序列中的整数。更具体地说，假设  $(i, j)$  是第  $i$  行第  $j$  列的单元格，我们将把序列中的第  $((i - 1) \times m + j)$  个元素（即  $a_{(i-1) \times m + j}$ ）填入该单元格。

如果在填满所有单元格后，至少满足以下两个条件中的一个，我们就说一个整数  $k$  是序列中的 "宾果整数"。

- 至少有一行，该行单元格中的所有整数都小于或等于  $k$ 。
- 至少有一列，该列单元格中的所有整数都小于或等于  $k$ 。

不难看出，一个序列可能有多个宾果整数，但在这个问题中，我们只对最小的宾果整数感兴趣。

计算给定序列的所有  $(nm)!$  排列的最小宾果整数之和。由于答案可能很大，请输出答案的模数 998 244 353。

输入

有多个测试用例。输入的第一行包含一个整数  $T$ ，表示测试用例的数量。对于每个测试用例

第一行包含两个整数  $n$  和  $m$  ( $1 \leq n, m \leq 2 \times 10^5, 1 \leq n \times m \leq 2 \times 10^5$ )，表示网格的行数和列数。

第二行包含  $n \times m$  整数  $a_1, a_2, \dots, a_{nm}$  ( $0 \leq a_i < 998\,244\,353$ )，表示给定的序列。

保证所有测试用例的  $n \times m$  之和不超过  $4 \times 10^5$ 。

输出

针对每个测试用例，输出一行包含一个整数的答案。

示例

标准输入	标准输出
4	56
2 2	60
1 3 2 4	60
3 1	855346687
10 10 10	
1 3	
20 10 30	
3 4	



1 1 4    5 1    4 1 9 1 9 8 10	
--------------------------------	--

备注

对于第一个测试案例，如果 1 和 2 不在同一行或同一列，那么最小的宾果整数将是 3，否则最小的宾果整数将是 2。1 和 2 不在同一行或列的排列有 8 种，因此答案是  $8 \times 3 + (4! - 8) \times 2 = 56$ 。

对于第二个示例测试用例，最小的宾果整数总是 10，所以答案是  $3! \times 10 = 60$ 。



## 问题 J. 社交媒体

时间限制

1 秒 内存限制

1024 兆字节

在社交媒体平台上，用户可以在他人的帖子下发表评论，表达自己的想法。但是，这些评论并不是每个人都能看到的。具体来说，用户  $C$  要想在用户  $B$  的帖子下看到用户  $A$  的评论，必须同时与  $A$  和  $B$  成为好友。如果用户在自己的帖子下发表评论，他/她的所有好友都能看到该评论。

作为该平台的活跃用户，您希望看到尽可能多的评论。平台上有  $k$  个用户（不包括您），编号从 1 到  $k$ 。平台上也有  $m$  条评论，但您可能无法看到所有评论，因为您只有  $n$  个好友。由于您需要参加 2024 年国际大学生计算机竞赛亚洲南京区域赛，您没有时间结交太多新朋友。如果您在平台上最多认识两个新朋友，您最多可以看到多少条评论？

### 输入

有多个测试用例。输入的第一行包含一个整数  $T$ ，表示测试用例的数量。对于每个测试用例

第一行包含三个整数  $n$ 、 $m$  和  $k$  ( $1 \leq n \leq k \leq 2 \times 10^5$ ,  $1 \leq m \leq 2 \times 10^5$ )，表示平台上您的好友数量、评论数量和用户数量（不包括您）。

第二行包含  $n$  个不同的整数  $f_1, f_2, \dots, f_n$  ( $1 \leq f_i \leq k$ )，表示您在平台上的好友。

对于下面的  $m$  行，第  $i$  行包含两个整数  $a_i$  和  $b_i$  ( $1 \leq a_i, b_i \leq k$ )，表示用户  $a_i$  在用户  $b_i$  的帖子下所写的评论。

保证所有测试用例的  $k$  和  $m$  之和都不会超过  $2 \times 10^5$ 。

### 输出

对于每个测试用例，输出一行包含一个整数的数据，表示如果您在平台上最多交到两个新朋友，您能看到的最大评论数。



示例

标准输入	标准输出
5	9
4 12 7	5
5 7 3 6	1
3 6	1
2 2	1
1 4	
2 4	
1 3	
7 6	
4 1	
5 4	
1 1	
1 1	
2 1	
3 7	
2 7 6	
2 4	
1 2	
3 2	
2 5	
5 4	
2 6	
4 6	
2 6	
1 1 2	
1	
1 2	
2 1 2	
1 2	
1 2	
2 1 100	
24 11	
11 24	

备注

在第一个示例测试用例中，您可以与用户 1 和用户 4 交朋友。在第二个示例测试用例中，您可以与用户 5 和用户 6 交朋友。在第三个示例测试用例中，您可以与用户 2 交朋友。

对于第四和第五个示例测试用例，您不需要结交新朋友，因为您已经可以看到所有评论。



问题 K 条

时间限制 1 秒 内存限制 1024 兆字节

一排有  $w$  个单元格，从左到右编号为 1 到  $w$ 。其中  $n$  个单元格是红色的， $m$  个单元格是黑色的，剩下的  $(w - n - m)$  个单元格是白色的。

您需要用一些条带覆盖所有红细胞。每个条带必须覆盖  $k$  个连续的细胞。请找出一种既能覆盖所有红细胞，又能满足以下所有约束条件的方法：

- 每个红细胞上都有一个条带。
- 没有黑格被条带覆盖。
- 没有两个条带覆盖同一个单元格。也就是说，每个单元格最多被一个条带覆盖。
- 使用的条带数量越少越好。

输入

有多个测试用例。输入的第一行包含一个整数  $T$ ，表示测试用例的数量。对于每个测试用例

第一行包含四个整数  $n$ 、 $m$ 、 $k$  和  $w$  ( $1 \leq n, m \leq 10^5, 1 \leq k \leq w \leq 10^9, n + m \leq w$ )，表示红色单元格数、黑色单元格数、每个条带的长度和单元格总数。

第二行包含  $n$  个整数  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq w$ )，表示单元格  $a_i$  是红色的。第三行包含  $m$  个整数  $b_1, b_2, \dots, b_m$  ( $1 \leq b_i \leq w$ )，表示单元格  $b_i$  为黑色。

保证给定的  $(n + m)$  个单元格是不同的。还可以保证  $n$  所有测试用例的  $m$  之和也不会超过  $2 \times 10^5$ 。

输出

对于每个测试用例

如果可以在满足所有限制条件的同时覆盖所有红细胞，首先输出一行，其中包含一个整数  $c$ ，表示使用的最小条带数。然后再输出一行，包含  $c$  个整数  $l_1, l_2, \dots, l_c$  ( $1 \leq l_i \leq w - k + 1$ )，中间用空格隔开，其中  $l_i$  是第  $i$  条带覆盖的最左边的单元格。如果有多个有效答案，可以输出任意一个。

如果无法这样做，就在一行中输出 -1。

示例

标准输入	标准输出
4	4



第三届环球杯

第 16 赛段：南京，2024 年 11 月 9-10 日

5 2 3 16	6 2 14 9
7 11 2 9 14	-1
13 5	2
3 2 4 11	1 4
6 10 2	-1
1 11	
2 1 2 6	
1 5	
3	
2 1 2 6	
1 5	
2	





问题 L. $P \oplus Q = R$

时间限制                      4 秒 内存限制  
1024 兆字节

爱丽丝想训练自己解决建设性问题。于是，她的朋友、超级人工智能主为爱丽丝生成了下面的问题。

给定整数  $n$ ，构造两个  $0, 1, \dots, (n - 1)$  的排列组合  $P = p_1, p_2, \dots, p_n$  和  $Q = q_1, q_2, \dots, q_n$ ，使得序列  $R = r_1, r_2, \dots, r_n$  仍然是  $0, 1, \dots, (n - 1)$  的排列组合。这里， $r_i = p_i \oplus q_i$ 。这里， $x \oplus y$  表示  $x$  和  $y$  的比特排他-或。

爱丽丝用她强大的计算能力解决了这个问题，她决定与大家分享这个问题。你能解决这个问题吗？

输入

有多个测试用例。输入的第一行包含一个整数  $T$ ，表示测试用例的数量。对于每个测试用例

第一行也是唯一一行包含一个整数  $n$  ( $1 \leq n \leq 2 \times 10^5$ )，表示置换的长度。保证所有测试用例的  $n$  之和不超过  $2 \times 10^6$ 。

输出

对于每个测试用例

如果存在两个满足约束条件的排列组合，首先在一行中输出 "是"。然后输出第二行，其中包含  $n$  个整数  $p_1, p_2, \dots, p_n$ ，中间用空格隔开。最后输出第三行，包含  $n$  个整数  $q_1, q_2, \dots, q_n$ ，中间用空格隔开。如果有多个有效答案，可以输出其中任何一个。

如果不存在满足约束条件的两种排列组合，只需在一行中输出 "否"。

示例

标准输入	标准输出
2	没有
3	是
4	0 2 1 3 3 2 0 1

备注

对于第二个测试案例， $R = \{3, 0, 1, 2\}$  仍然是  $0, 1, 2, 3$  的排列。





## 问题 M. 无尽审判的裁决者

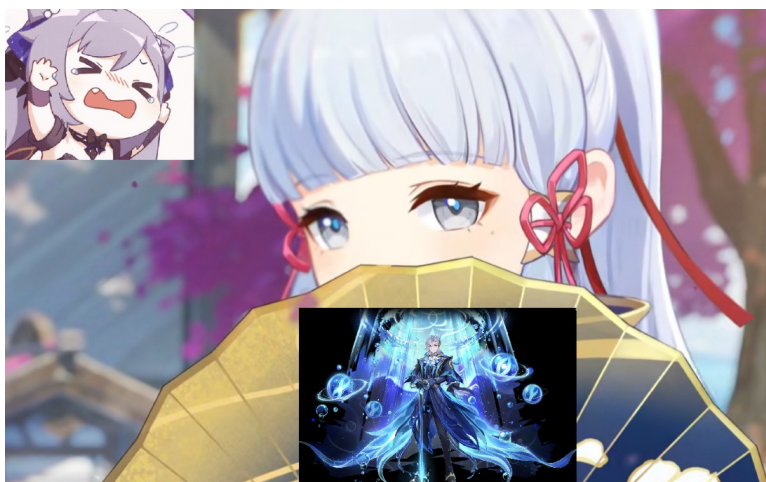
时间限制

1 秒 内存限制

1024 兆字节

Neuvillette 是被称为 "Iudex " 的 Fontaine 首席法官，以公正无私而闻名。他是世界著名游戏《源氏冲击》中的可玩角色，以其强大的带电攻击而闻名，这种攻击可以击中特定范围内的敌人。

由于他非常强大，许多玩家在挑战几乎所有任务时都会使用他。然而，泰瓦特的每个人对此都不满意，尤其是其他 ADC（攻击伤害携带者）角色，包括上里绫香、可青等。他们决定一起说服 Mihoyo 在游戏中削弱 Neuvillette。为此，他们必须提交一份报告，说明 Neuvillette 在几种情况下的总伤害。



根据《源氏冲击》官方资料制作

每个战斗场景都发生在一个二维平面上。新维莱特最初站在  $(0, 0)$  上，面向  $(x_0, y_0)$ ，进行持续  $t$  个单位时间的带电攻击，并在每个单位时间内逆时针旋转 1 拉德。也就是说，纽维莱特在  $2\pi$  单位时间内逆时针转了一圈。

考虑一条从  $(0, 0)$  射向 Neuvillette 方向的射线。如果形状为凸多边形的目标与攻击范围内的点有公共点，那么它将在单位时间内持续受到 1 次伤害。

作为一名经验丰富的程序员，你被 Ayaka 召唤。这次，你的任务是计算目标在前  $t$  个单位时间内造成的伤害。

### 输入

每个测试文件中只有一个测试用例。

第一行包含五个整数  $n$ 、 $x_0$ 、 $y_0$ 、 $d$  和  $t$  ( $3 \leq n \leq 100$ 、 $-10^4 \leq x_0, y_0 \leq 10^4$ 、 $x^2 + y^2 > 0$ )、  
 $1 \leq d, t \leq 10^4$ ).

对于下面的  $n$  行，第  $i$  行包含两个整数  $x_i$  和  $y_i$  ( $-10^4 \leq x_i, y_i \leq 10^4$ )，表示凸多边形第  $i$  个顶点的坐标。



所有  $n$  个顶点都按逆时针顺序给出，其中任意三个顶点都不相交。也就是说，在凸多边形的内部或边界上不存在同时在圆的内部或边界上的点。

## 输出

输出一行包含一个实数的数据，表示目标在前  $t$  个单位时间内受到的伤害。



如果绝对误差或相对误差不超过  $10^{-6}$ ，您的答案将被视为正确。从形式上讲，假设您的输出结果是  $a$ ，而评委的答案是  $b$ ，那么您的输出结果在且仅在以下条件下被接受

$$\frac{|a-b|}{\max(1,|b|)} \leq 10^{-6}.$$

实例

标准输入	标准输出
3 1 0 1 1 1 2 2 1 2 2	1.000000000000
3 1 0 1 2 1 2 2 1 2 2	1.570796326795
3 1 0 1 10000 1 2 2 1 2 2	2500.707752257475

备注

下图同时显示了示例测试用例的初始状态。

