A. 选石头

问题描述

有 n 种石头，种类的编号为1~ n。

小A想将石头排成一排， 组成一个魔法阵。

只有 m 组石头可以放在相邻的位置，分别是 $(a_1, b_1), (a_2, b_2), \ldots, (a_m, b_m)$，其余种类的石头不可以放在相邻的位置。

请判断是否存在这样一种排列方式，使得其中包含 $c_1, c_2, \ldots, c_k$ 中的每一种石头。

如果存在，输出最少需要的石头的数量；否则输出-1.

输入格式

第一行两个整数 n 和 m。

接下来 m 行，每行两个数，表示 a_i 和 b_i。

接下来一行一个整数 k。

接下来一行 k 个整数 $c_1, c_2, \ldots, c_k$。

输入1

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

输出1

```
5
```

解释1

1 4 2 4 3 为一种合法的方案，这里的数字为石头的种类。

输入2

```
10 10
3 9
3 8
8 10
2 10
5 8
6 8
5 7
6 7
1 6
2 4
4
1 2 7 9
```

输出2

```
11
```

解释2

2 10 8 3 9 3 8 5 7 6 1 为一种合法的方案

数据范围

$n \le 10^5; 0 \le m \le 10^5; 1 \le a_i < b_i \le n; 1 \le k \le 17$

$1 \le c_1 < c_1 \cdots < c_k \le n$

保证 $i \ne j$ 时有 $(a_i, b_i) \ne (a_j, b_j)$

B. 游戏

问题描述

两个人在玩游戏。初始有一个数字 $x = 0$ 。

给两个长为 n 的字符串 s 和 t，字符串下标从1开始。

s 由数字 $0, 1,$ $\ldots, 9$ 组成。

t 由小写字母 a 和 b 组成。

有 n 轮游戏。

第 i 轮游戏，若 $t_i = a$ ，则由alice操作；若 $t_i = b$，则由bob操作。

第 i 轮游戏，操作的人可以选择将 x 变成 $10x + s_i$ 或者 $10x$。

如果最后 x 是7的倍数，则 bob 胜利；否则 alice胜利。

两人都绝顶聪明，采取最优策略，请输出获胜者。

输入格式

第一行一个整数 n 。

第二行一个字符串 s。

第三行一个字符串 t。

输入1

```
2
14
ab
```

输出1

```
bob
```

解释1

如果alice 将x变成1，则bob将x变成14。

如果alice 将x变成0，则bob将x变成0。

输入2

```
5
12345
aaaab
```

输出2

```
alice
```

数据范围

$1 \le n \le 2 * 10^5$

C. 移动

问题描述

有一个 $H+1$ 行 W 列的矩阵，你每步可以在矩阵中向右或向下移动一个格子。

但是在第 $i\,(1 \le i \le H)$行中， 你无法 第 A_i 至 B_i 列的格子向下走。

对于每一个 $k,(1 \le k \le H)$，求出从第 1 行的任意一个格子出发移动到第 $k+1$ 行的最少步数，若无法移动到第$k+1$行则输出 `-1`。

输入格式

第一行两个整数 H 和 W，

接下来 H 行每行两个整数 A_i 和 B_i。

输出格式

共 H 行，每行一个整数。

输入1

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

输出1

```
1
3
6
-1
```

解释1

k=1 时，其中一种答案最小的移动顺序为 (1,1)→(2,1)；

k=2 时，一种移动顺序为 (1,1)→(2,1)→(2,2)→(3,2)；

k=3 时，一种移动顺序为 (1,1)→(2,1)→(2,2)→(3,2)→(3,3)→(3,4)→(4,4)

k=4 时，无法从第 1 行移动到第 5 行。

数据范围

$1 \leq H, W \leq 2 * 10^5; 1 \leq A_i \leq B_i \leq W$

D. 数字操作

给你一个长度为 n 的非负整数数组 $a_1, a_2, \ldots, a_n$。

你可以选择一个非负整数 x。

令 $b_i = a_i$ xor x。 xor 为异或运算，把两个操作数按二进制位运算，如果两个数二进制下第k位相同，则运算结果的二进制下第k位为0，否则为1。

$m = max(b_1, b_2, \ldots, b_n)$

问 m 的最小值是多少。

输入格式

第一行一个整数 n。

第二行 n 个非负整数 $a_1, a_2, \ldots, a_n$。

输出格式

一行一个整数。

输入1

```
3
1 2 3
```

输出1

```
2
```

解释1

选择 $x = 3$， $b_1 = 2, b_2 = 1, b_3 = 0$， $m = 2$。

数据范围

$1 \le n \le 150000; 0 \le a_i < 2^{30}$

E. 排序

有 n 个人，编号1~n。

n 个人排成一排， 从左到右第 i 个人的编号为 p_i。

你的目标是将这些人按编号升序排好，最左边的编号最小，最右边的编号最大。

你可以以任意顺序重复以下三种操作任意次：

1. 选择一个数 i，支付 a_i 的代价，将编号为 i 的人移动到任意位置。
2. 选择一个数 i，支付 b_i 的代价，将编号为 i 的人移动到最左边。
3. 选择一个数 i，支付 c_i 的代价，将编号为 i 的人移动到最右边。

问最小代价是多少，才能使得这些人从左到右按编号升序排列。

输入格式

第一行一个整数 n。

第二行 n 个整数 $p_1, p_2, ..., p_n$。

接下来 n 行，每行 3 个整数，表示 a_i, b_i, c_i。

输出格式

一个整数表示最小代价。

样例输入1

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

样例输出1

```
6
```

样例解释1

将编号为3的人移动到最右边，花费代价$c_3 = 6$。

样例输入2

```
6
2 6 5 3 4 1
10 8 16
30 2 10
10 17 8
11 27 22
8 6 5
15 29 2
```

样例输出2

```
15
```

花费 $b_1 = 8$ 将 编号为1的人移动到最左边。126534

花费 $c_5 = 5$ 将 编号为5的人移动到最右边。126345

花费 $c_6 = 2$ 将 编号为6的人移动到最右边。123456

数据范围

$1 \le n \le 2 * 10^5; 1 \le p_i \le n; 1 \le a_i, b_i, c_i \le 10^9$

p 为1~n的排列。

F. 树上路径

有一棵 n 个点的树，第 i 条边连接 a_i 和 b_i ，第 i 个点颜色为 c_i。

对于每个 $k = 1, 2, \ldots, n$ ，计算经过颜色 k 的简单路径的数目。

路径 i 到 j 与 路径 j 到 i 视为相同的路径。

输入格式

第一行一个整数 n。

第二行 n 个整数 $c_1, c_2, \ldots, c_n$。

接下来 $n-1$ 行，每行两个整数 a_i, b_i，表示一条边。

输出格式

n 行，每行一个整数。

样例输入1

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

样例输出1

```
5
4
0
```

样例解释1

用 $p_{i,j}$ 表示端点为 i 和 j 的简单路径

经过颜色1的简单路径有 5条：$p_{1,1}, p_{1,2}, p_{1,3}, p_{2,3}, p_{3,3}$

经过颜色2的简单路径有 4条：$p_{1,2}, p_{1,3}, p_{2,2}, p_{2,3}$。

没有经过颜色3的简单路径。

样例输入2

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

样例输出2

```
18
15
0
14
23
0
23
0
```

数据范围

$1 \le n \le 2*10^5$；$1 \le a_i, b_i, c_i \le n$