

# 1 分组

## 1.1 Problem Statement

小 D 正在招聘员工。

共有  $n$  人报名了招聘，这些人被依次编号为  $1, 2, \dots, n$ 。其中，第  $i$  个人的工作经验为  $w_i$ ，薪水为  $s_i$ 。

小 D 想要招聘  $2k$  个人，并将这些人分为  $k$  组，每组两个人。

每组的两个人中，会有一个人作为组长，而另一个人为组员。对于每个组而言，小 D 都要求组长的工作经验 **不小于** 组员的工作经验。

此外，每个应聘者会有一个意愿  $p_i$ ，表示他应聘的岗位。若  $p_i = 1$ ，则该应聘者应聘的是组长；若  $p_i = 2$ ，则他应聘的是组员；若  $p_i = 3$ ，则他既可以当组长又可以当组员。

小 D 想要知道，在不改变意愿的情况下，是否可以招聘满足条件的  $2k$  个人。如果可以，小 D 还想要知道招聘的人薪水和的最小值。

但是小 D 并不会，请你帮帮他。

## 1.2 Input Format

从标准输入读入数据。

第一行两个整数  $n, k$ ，表示应聘者个数以及小组个数。

接下来  $n$  行，每行三个整数  $w_i, s_i, p_i$ ，依次表示每个人的工作经验，薪水，以及意愿。

## 1.3 Output Format

向标准输出输出答案。

如果无法做到，则输出一行一个整数  $-1$ ，否则输出最小总薪水。

## 1.4 Sample 1

### 1.4.1 Input

```
6 2
20 6 1
6 7 2
4 8 3
3 10 2
8 5 3
```

4 3 1

### 1.4.2 Output

22

### 1.4.3 Explanation

选择如下两组即可：

- 组长：1 号应聘者，组员：5 号应聘者；
- 组长：6 号应聘者，组员：3 号应聘者。

## 1.5 Sample 2

见下发文件 `group/group2.in` 与 `group/group2.ans`。

## 1.6 Sample 3

见下发文件 `group/group3.in` 与 `group/group3.ans`。

## 1.7 Constraints

对于所有测试数据， $1 \leq n \times k \leq 10^5$ ， $1 \leq w_i, s_i \leq 10^9$ ， $1 \leq p_i \leq 3$ 。

- 子任务 1 (20 分)： $n, k \leq 10$ ；
- 子任务 2 (20 分)： $n, k \leq 20$ ；
- 子任务 3 (20 分)： $n, k \leq 500$ ；
- 子任务 4 (20 分)： $n, k \leq 5000$ ；
- 子任务 5 (20 分)：无特殊限制。

## 2 字符串

### 2.1 Problem Statement

小 D 正在研究字符串。

小 D 想了两个仅由 0,1 组成的字符串  $S, T$ ，长度分别为  $n, m$ 。

小 D 称一个  $S$  的下标序列  $1 \leq i_1 < i_2 < \dots < i_m \leq n$  是**好的**，当且仅当  $S_{i_1} S_{i_2} \dots S_{i_m} = T$ 。

小 D 称一个  $S$  的下标序列  $1 \leq i_1 < i_2 < \dots < i_m \leq n$  是**优秀的**，当且仅当他是好的，且对于任意  $0 \leq j \leq m$ ，有  $i_j + 1 = i_{j+1}$  成立，或满足  $S_{i_j+1} = S_{i_j+2} = \dots = S_{i_{j+1}-1}$ 。**值得一提的是，这里我们认为  $i_0 = 0$  而  $i_{m+1} = n + 1$ 。**

小 D 想要知道， $S$  中是否存在一个优秀的下标序列。如果存在，请给出一个解。但是小 D 并不会，请你帮帮他。

### 2.2 Input Format

从标准输入读入数据。

第一行两个整数  $n, m$ ，表示  $S, T$  的长度。

第二行一个长度为  $n$  的 0/1 字符串表示  $S$ 。

第三行一个长度为  $m$  的 0/1 字符串表示  $T$ 。

### 2.3 Output Format

向标准输出输出答案。

如果不存在优秀的下标序列，输出一行一个数 -1。

否则，输出一行  $m$  个整数，依次表示  $i_1, i_2, \dots, i_m$ 。

如果有多个优秀的下标序列，你可以输出任意一个。

### 2.4 Sample 1

#### 2.4.1 Input

```
14 4
00111001110111
0110
```

#### 2.4.2 Output

```
2 5 8 11
```

### 2.4.3 Explanation

该下标序列在  $S$  中对应 001110011101111，容易验证其符合优秀的要求。

## 2.5 Sample 2

### 2.5.1 Input

```
8 4
01010101
0110
```

### 2.5.2 Output

```
-1
```

## 2.6 Sample 3

见下发文件 `string/string3.in` 与 `string/string3.ans`。

## 2.7 Sample 4

见下发文件 `string/string4.in` 与 `string/string4.ans`。

## 2.8 Constraints

对于所有测试数据， $1 \leq m \leq n \leq 3 \times 10^5$ 。

- 子任务 1 (30 分):  $n \leq 20$ ;
- 子任务 2 (30 分):  $n \leq 5000$ ;
- 子任务 3 (20 分):  $n \leq 10^5$ ;
- 子任务 4 (20 分): 无特殊限制。

## 2 折纸

### 2.1 Problem Statement

小 D 正在折纸。

小 D 现在有一张  $n \times m$  的方格纸，每个方格有一个颜色，用小写字母表示。具体地，位于第  $i$  行第  $j$  列的方格颜色为  $c_{i,j}$ 。

小 D 想要进行折纸。具体地，他可以选择一个水平或竖直的**不穿过任何小方格内部**的直线，并将整张纸沿这条线对折。并且在折时，小 D 要求在下面的那部分必须**不小于**上面的那部分。也就是说，我们把较小的那部分折到较大的那部分上面。但如果两部分一样大，那么向两个方向折都是允许的。

为了最终结果的美观，小 D 要求每次折纸时，**对应位置的颜色必须一样**。

小 D 想要知道，在若干次对折后，他最终可以折出多少种本质不同的结果。

我们认为两个结果是本质不同的，当且仅当他们的最下层在**原方格纸中对应的区域不同**。也就是说，颜色一样的两个结果可能是不同的。

但是小 D 并不会，请你帮帮他。

**注：** 你可以认为在一次折纸后，上下两部分会合成一张纸，即你不用考虑上面的部分在之后的折纸过程中，被折到下面那部分之间导致的不合法情况。

### 2.2 Input Format

从标准输入读入数据。

第一行两个整数  $n, m$ ，表示方格纸的大小。

接下来  $n$  行，每行一个长度为  $m$  的小写字母组成的字符串，表示第  $i$  行所有方格的颜色。

### 2.3 Output Format

向标准输出输出答案。

输出一行一个整数表示答案。

### 2.4 Sample 1

#### 2.4.1 Input

```
5 7
baabbaa
cbbccbb
```

```
ababbab
cabccba
bccaaacc
```

#### 2.4.2 Output

2

#### 2.4.3 Explanation

第一种结果是不折，第二种结果是沿第四、第五列之间对折。

### 2.5 Sample 2

见下发文件 `origami/origami2.in` 与 `origami/origami2.ans`。

### 2.6 Sample 3

见下发文件 `origami/origami3.in` 与 `origami/origami3.ans`。

### 2.7 Constraints

对于所有测试数据， $1 \leq n \times m \leq 10^6$ 。

- 子任务 1 (30 分):  $n, m \leq 30$ ;
- 子任务 2 (20 分):  $n = 1$ ;
- 子任务 3 (20 分):  $n, m \leq 1000$ ;
- 子任务 4 (30 分): 无特殊限制。

## 3 集合

### 3.1 Problem Statement

小 D 正在研究集合。

小 D 想要维护一个**可重集合**  $S$ 。

小 D 想要支持如下四种操作：

- 在  $S$  中插入一个元素  $x$ ；
- 删除  $S$  中某个元素  $x$  的**一次出现**，保证  $x \in S$ ；
- 给  $S$  中的所有元素加上 1；**为了避免整数溢出，小 D 在每次加法后会将所有数字对  $2^{30}$  取模。**
- 给  $S$  中的所有元素异或上  $x$ 。

小 D 想要知道，所有操作都进行完后  $S$  中的所有元素。

但是小 D 并不会，请你帮帮他。

### 3.2 Input Format

从标准输入读入数据。

第一行两个整数  $n, q$ ，表示  $S$  中初始的元素个数，以及小 D 的操作次数。

第二行  $n$  个空格隔开的整数  $a_1, a_2, \dots, a_n$ ，表示  $S$  中的初始元素。

接下来  $q$  行每行 1 或 2 个整数，表示一次操作：

- 1  $x$ ：插入一个元素  $x$ ；
- 2  $x$ ：删除元素  $x$  的一次出现；
- 3：给所有元素 +1；
- 4  $x$ ：给所有元素异或上  $x$ 。

### 3.3 Output Format

向标准输出输出答案。

输出一行若干个空格隔开的整数，表示最终  $S$  中的元素。你需要按照**从小到大的顺序**进行输出。

### 3.4 Sample 1

#### 3.4.1 Input

```
2 4
5 10
```

3  
1 6  
4 2  
2 4

### 3.4.2 Output

4 9

### 3.4.3 Explanation

集合  $S$  的变化如下：

- 初始时， $S = \{5, 10\}$ ；
- 第一次操作后， $S = \{6, 11\}$ ；
- 第二次操作后， $S = \{6, 6, 11\}$ ；
- 第三次操作后， $S = \{4, 4, 9\}$ ；
- 第四次操作后， $S = \{4, 9\}$ ；

## 3.5 Sample 2

见下发文件 `set/set2.in` 与 `set/set2.ans`。

## 3.6 Sample 3

见下发文件 `set/set3.in` 与 `set/set3.ans`。

## 3.7 Constraints

对于所有测试数据， $1 \leq n, q \leq 3 \times 10^5$ ， $0 \leq a_i, x < 2^{30}$ 。

- 子任务 1 (25 分)：  $n, q \leq 5000$ ；
- 子任务 2 (15 分)： 没有操作 3；
- 子任务 3 (15 分)： 没有操作 4；
- 子任务 4 (25 分)：  $n, q \leq 10^5$ ；
- 子任务 5 (20 分)： 无特殊限制。