

---

# APIO'07

Asia-Pacific Informatics Olympiad

**12th May, 2007**

---

**Duration: 5 hours**  
**(8:15 ~ 13:15)**

**3 questions**

中文参考译文

中国 NOISC 翻译

## 第 1 题

### 风铃 (Mobiles)

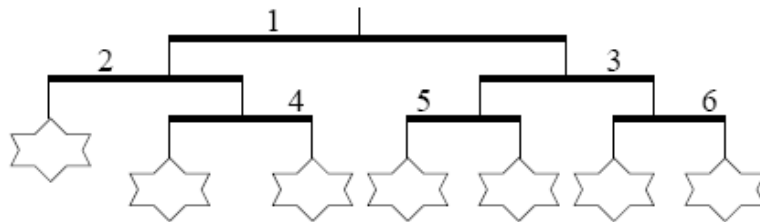
输入文件: mobiles.in

输出文件: mobiles.out

时间限制: 1 秒 内存限制: 32M

你准备给弟弟 Ike 买一件礼物, 但是, Ike 挑选礼物的方式很特别: 他只喜欢那些能按照他的特有方式排成有序的东西。

你准备给 Ike 买一个风铃。风铃是一种多层的装饰品, 一般挂在天花板上。每个风铃都包含一些由竖直的线连起来的水平杆。每根杆的两端都有线连接, 下面或者挂着另一根水平杆, 或者挂着一个玩具。下面是一个风铃的例子:



为使你的弟弟满意, 你需要选一个满足下面两个条件的风铃:

- (1) 所有的玩具都在同一层(也就是说, 每个玩具到天花板之间的杆的个数是一样的)或至多相差一层。
- (2) 对于两个相差一层的玩具, 左边的玩具比右边的玩具要更靠下一点。

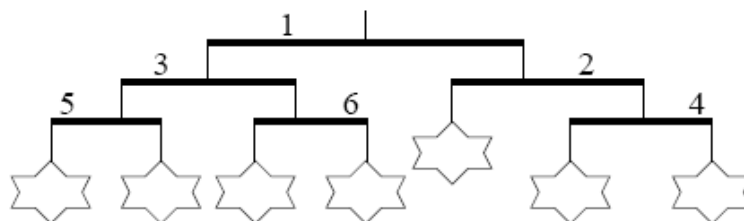
风铃可以按照下面的规则重新排列: 任选一根杆, 将杆两端的线“交换”。也就是解开一根杆左右两端的线, 然后将它们分别绑到杆的另一端。注意这个操作不会改变下面的杆上线的排列顺序。

由于你正在参加信息学奥林匹克的训练, 所以你决定设计一个算法, 判断能否通过重新排列, 将一个给定的风铃变为 Ike 喜欢的样子。

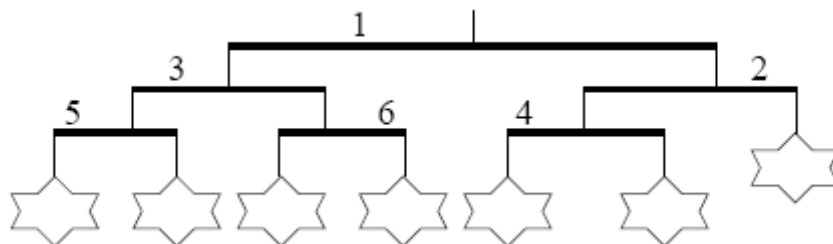
考虑上面的例子, 上图中的风铃满足条件(1), 却不满足条件(2)——最左边的那个玩具比它右边的要高。

但是, 我们可以通过下面的步骤把这个风铃变成一个 Ike 喜欢的形式:

1. 第一步, 将杆 1 的左右两端交换, 这使得杆 2 和杆 3 的位置互换, 交换的结果如下图所示:



2. 第二步,也是最后一步,将杆 2 的左右两端交换,这使得杆 4 到了左边,原来在左边的玩具到了右边,交换的结果如下图所示:



现在这个风铃就满足 Ike 的条件了。

你的任务是:给定一个风铃的描述,求出最少需要多少次交换才能使这个风铃满足 Ike 的条件(如果可能的话)。

### 输入

输入的第一行包含一个整数  $n$  ( $1 \leq n \leq 1\,000\,000$ ), 表示风铃中有多少根杆。

接下来的  $n$  行描述杆的连接信息。这部分的第  $i$  行包含两个由空格分隔的整数  $l_i$  和  $r_i$ , 描述杆  $i$  的左右两端悬挂的东西。如果挂的是一个玩具, 则对应的值为 -1, 否则为挂在下面的杆的编号。

如果杆  $i$  下面挂有其它杆, 则这些杆的编号将严格大于  $i$ 。杆 1 位于风铃的顶部。

### 输出

输出仅包含一个整数。表示最少需要多少次交换能使风铃满足 Ike 的条件。如果不可能满足, 输出 -1。

### 样例输入

```
6
2 3
-1 4
5 6
-1 -1
-1 -1
-1 -1
```

### 样例输出

```
2
```

### 说明

上面的样例输入给出了前面描述的示例情形。

### 评分

对于每一个测试点, 如果你得到了正确的结果, 得到该测试点 100% 的分数, 否则得 0 分。

## 第 2 题

### 数据备份 (Backup)

输入文件: backup.in

输出文件: backup.out

时间限制: 1 秒 内存限制: 32M

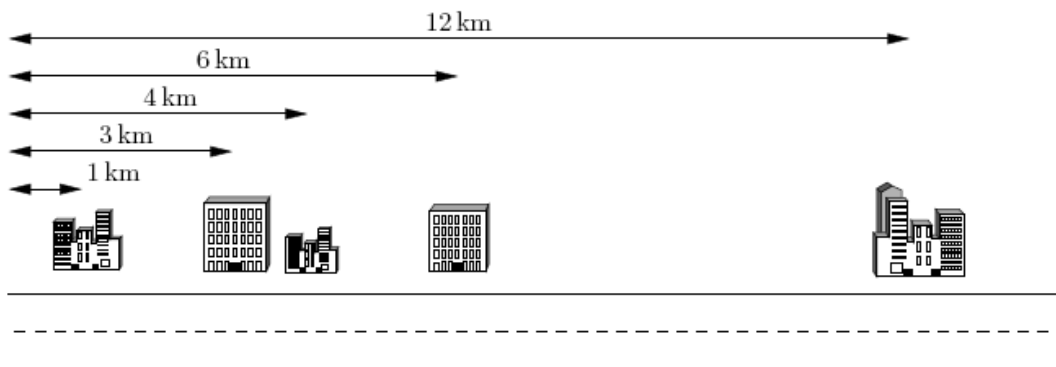
你在一家 IT 公司为大型写字楼或办公楼 (offices) 的计算机数据做备份。然而数据备份的工作是枯燥乏味的, 因此你想设计一个系统让不同的办公楼彼此之间互相备份, 而你则坐在家中尽享计算机游戏的乐趣。

已知办公楼都位于同一条街上。你决定给这些办公楼配对 (两个一组)。每一对办公楼可以通过在这两个建筑物之间铺设网络电缆使得它们可以互相备份。

然而, 网络电缆的费用很高。当地电信公司仅能为你提供  $K$  条网络电缆, 这意味着你仅能为  $K$  对办公楼 (或总计  $2K$  个办公楼) 安排备份。任一个办公楼都属于唯一的配对组 (换句话说, 这  $2K$  个办公楼一定是相异的)。

此外, 电信公司需按网络电缆的长度 (公里数) 收费。因而, 你需要选择这  $K$  对办公楼使得电缆的总长度尽可能短。换句话说, 你需要选择这  $K$  对办公楼, 使得每一对办公楼之间的距离之和 (总距离) 尽可能小。

下面给出一个示例, 假定你有 5 个客户, 其办公楼都在一条街上, 如下图所示。这 5 个办公楼分别位于距离大街起点 1km, 3km, 4km, 6km 和 12km 处。电信公司仅为你提供  $K=2$  条电缆。



上例中最好的配对方案是将第 1 个和第 2 个办公楼相连, 第 3 个和第 4 个办公楼相连。这样可按要求使用  $K=2$  条电缆。第 1 条电缆的长度是  $3\text{km} - 1\text{km} = 2\text{km}$ , 第 2 条电缆的长度是  $6\text{km} - 4\text{km} = 2\text{km}$ 。这种配对方案需要总长  $4\text{km}$  的网络电缆, 满足距离之和最小的要求。

#### 输入

输入的第一行包含整数  $n$  和  $k$ , 其中  $n$  ( $2 \leq n \leq 100\,000$ ) 表示办公楼的数目,  $k$  ( $1 \leq k \leq n/2$ ) 表示可利用的网络电缆的数目。

接下来的  $n$  行每行仅包含一个整数 ( $0 \leq s \leq 1000\,000\,000$ ), 表示每个办公楼

到大街起点处的距离。这些整数将按照从小到大的顺序依次出现。

## 输出

输出应由一个正整数组成，给出将  $2K$  个相异的办公楼连成  $k$  对所需的网络电缆的最小总长度。

## 样例输入

5 2  
1  
3  
4  
6  
12

## 样例输出

4

## 说明

上面的样例输入给出了前面描述的示例情形。

## 评分

对于每一个测试点，如果写到输出文件中的答案正确，则得到该测试点 100% 的分数，否则得零分。30% 的输入数据满足  $n \leq 20$ 。60% 的输入数据满足  $n \leq 10\,000$ 。

## 第 3 题

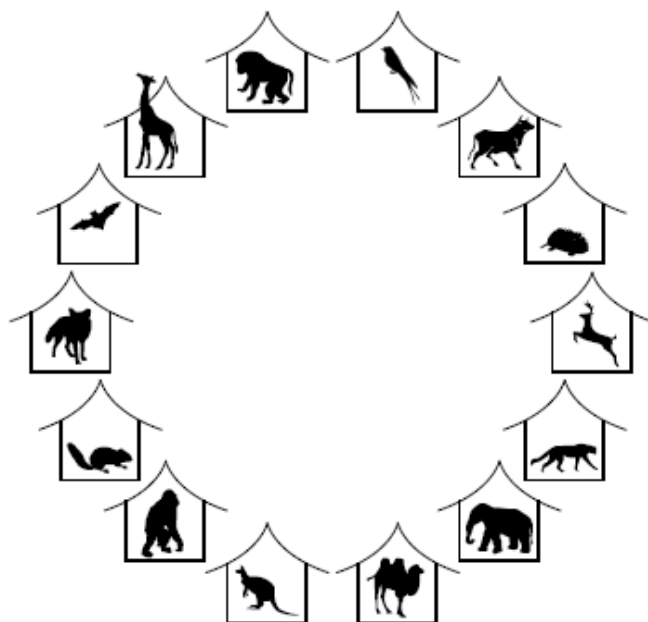
### 动物园 (Zoo)

输入文件: zoo.in

输出文件: zoo.out

时间限制: 1 秒 内存限制: 32M

新建的圆形动物园是亚太地区的骄傲。圆形动物园坐落于太平洋的一个小岛上，它包含一大圈围栏，每个围栏里有一种富有异国风情的动物。如下图所示：



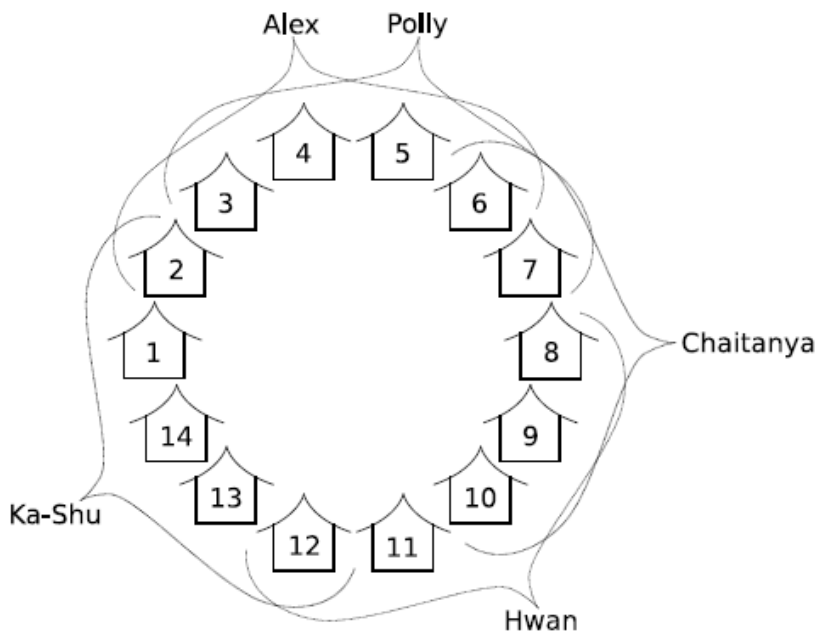
你是动物园的公关主管。你要做的是，让每个参观动物园的游客都尽可能高兴。今天有一群小朋友来到动物园参观，你希望能让他们在动物园度过一段美好的时光。但这并不是一件容易的事——有些小朋友喜欢某些动物，而有些小朋友则害怕某些动物。例如，Alex 喜欢可爱的猴子和考拉，而害怕拥有锋利牙齿的狮子。而 Polly 会因狮子有美丽的鬃毛而喜欢它，但害怕有臭味的考拉。

你可以选择将一些动物从围栏中移走以使得小朋友不会害怕。但你移走的动物也不能太多，否则留给小朋友们观赏的动物就所剩无几了。

每个小朋友站在大围栏圈的外面，可以看到连续的 5 个围栏。你得到了所有小朋友喜欢和害怕的动物信息。当出现下面两种情形之一时，小朋友就会高兴：

- 至少有一个他害怕的动物（从视线可见的范围内）被移走，或
- 至少有一个他喜欢的动物没有被（从视线可见的范围内）移走。

例如，考虑下图中的小朋友和动物：



小朋友	可见的围栏	害怕的动物	喜欢的动物
Alex	2,3,4,5,6	围栏 4	围栏 2,6
Polly	3,4,5,6,7	围栏 6	围栏 4
Chaitanya	6,7,8,9,10	围栏 9	围栏 6,8
Hwan	8,9,10,11,12	围栏 9	围栏 12
Ka-Shu	12,13,14,1,2	围栏 12,13, 2	-

假如你将围栏 4 和 12 的动物移走。Alex 和 Ka-Shu 将很高兴，因为至少有一个他们害怕的动物被移走了。这也会使 Chaitanya 高兴，因为他喜欢的围栏 6 和 8 中的动物都保留了。但是，Polly 和 Hwan 将不高兴，因为他们看不到任何他们喜欢的动物，而他们害怕的动物都还在。这种安排方式使得三个小朋友高兴。

现在换一种方法，如果你将围栏 4 和 6 中的动物移走，Alex 和 Polly 将很高兴，因为他们害怕的动物被移走了。Chaitanya 也会高兴，因为虽然他喜欢的动物 6 被移走了，他仍可以看到围栏 8 里面他喜欢的动物。同样的 Hwan 也会因可以看到自己喜欢的围栏 12 里的动物而高兴。唯一不高兴的只有 Ka-Shu。

如果你只移走围栏 13 中的动物，Ka-Shu 将高兴，因为有一个他害怕的动物被移走。Alex, Polly, Chaitanya 和 Hwan 也会高兴，因为他们都可以看到至少一个他们喜欢的动物。所以有 5 个小朋友会高兴。这种方法使得最多的小朋友高兴。

### 输入

输入的第一行包含两个整数  $N, C$ ，用空格分隔。 $N$  是围栏数( $10 \leq N \leq 10\,000$ )， $C$  是小朋友的个数( $1 \leq C \leq 50\,000$ )。围栏按照顺时针的方向编号为  $1, 2, 3, \dots, N$ 。

接下来的  $C$  行，每行描述一个小朋友的信息，以下面的形式给出：

$EFLX_1 X_2 \cdots X_F Y_1 Y_2 \cdots Y_L$

其中：

$E$  表示这个小朋友可以看到的第一个围栏的编号( $1 \leq E \leq N$ )，换句话说，该小朋友可以看到的围栏为  $E, E+1, E+2, E+3, E+4$ 。注意，如果编号超过  $N$  将继续从 1 开始算。如：当  $N=14, E=13$  时，这个小朋友可以看到的围栏为 13,14,1, 2 和 3。

$F$  表示该小朋友害怕的动物数。 $L$  表示该小朋友喜欢的动物数。

围栏  $X_1, X_2, \cdots, X_F$  中包含该小朋友害怕的动物。

围栏  $Y_1, Y_2, \cdots, Y_L$  中包含该小朋友喜欢的动物。

$X_1, X_2, \cdots, X_F, Y_1, Y_2, \cdots, Y_L$  是两两不同的整数，而且所表示的围栏都是该小朋友可以看到的。

小朋友已经按照他们可以看到的第一个围栏的编号从小到大的顺序排好了(这样最小的  $E$  对应的小朋友排在第一个，最大的  $E$  对应的小朋友排在最后一个)。注意可能有多于一个小朋友对应的  $E$  是相同的。

## 输出

仅输出一个数，表示最多可以让多少小朋友高兴。

### 样例输入 1

```
14 5
2 1 2 4 2 6
3 1 1 6 4
6 1 2 9 6 8
8 1 1 9 12
12 3 0 12 13 2
```

### 样例输出 1

```
5
```

### 样例输入 2

```
12 7
1 1 1 1 5
5 1 1 5 7
5 0 3 5 7 9
7 1 1 7 9
9 1 1 9 11
9 3 0 9 11 1
11 1 1 11 1
```

### 样例输出 2

```
6
```

## 说明

上面的样例 1 给出了前面描述的示例情形。它使得所有小朋友 ( $C=5$ ) 高兴。样例 2 给出了这样的情形，要使得所有小朋友 ( $C=7$ ) 都高兴是不可能的。

## 评分

对于每一个测试点，如果你的答案正确，则该测试点得满分，否则得 0 分。