

1 翻牌

(flip.cpp)

1.1 问题描述

给定一行中有 n 张卡片，其中 n 是奇数。每张卡片的两面都有数字。对于第 i 张卡片，正面朝上的数字是 a_i ，反面朝上的数字是 b_i 。

Grammy 想要最大化所有朝上的数字的中位数。为此，她最多可以进行以下操作一次：

选择一个区间 $[l, r]$ ，并翻转该区间内的所有卡片。翻转后， b_i 会变为朝上，而 a_i 会变为朝下，适用于所有 $i \in [l, r]$ 。

请帮助 Grammy 计算在她的最优策略下，所有朝上的数字的中位数。

请记住，一个序列的中位数是该序列中第 $\frac{n+1}{2}$ 大的数字。

1.2 输入

第一行包含两个整数 n ($1 \leq n < 3 \times 10^5$ ，且 $n \bmod 2 = 1$)，表示卡片的数量。

接下来的 n 行中，每行包含两个整数 a_i 和 b_i ($1 \leq a_i, b_i \leq 10^9$)，分别表示初始时卡片的正面和反面的数字。

1.3 输出

输出一个整数，表示在 Grammy 最优策略下，所有朝上的数字的中位数。

1.4 输入输出样例 1

1.4.1 输入样例

```
1
2 1
```

1.4.2 输出样例

```
2
```

1.5 输入输出样例 2

1.5.1 输入样例

```
5
3 6
5 2
4 7
6 4
```

2 8

1.5.2 输出样例

6

1.6 输入输出样例 3

见下发文件中的 `flip.in` 和 `flip.out`。

1.7 约定和数据范围

对所有测试点数据, 保证 $1 \leq n < 3 \times 10^5$, 且 $n \bmod 2 = 1$, $1 \leq a_i, b_i \leq 10^9$ 。
每个测试点的具体限制见下表:

测试点编号	$n <$
1 ~ 2	500
3 ~ 4	1000
5 ~ 8	5000
9 ~ 12	10^4
13 ~ 14	5×10^4
15 ~ 17	10^5
18 ~ 20	3×10^5