

Problem 1. keys

Input file: `keys.in`
Output file: `keys.out`
Time limit: 1 second
Memory limit: 256MB

现在有 n 个人, k 把钥匙, 还有一个办公室, 他们在 x 轴上.

其中, 第 i 个人的位置用一个整数 a_i 表示, 第 i 把钥匙的位置用一个整数 b_i 表示, 办公室的位置用整数 p 表示. 保证钥匙的数量大于等于人的数量.

现在, 他们每个人都需要先去拿到一把钥匙 (每个人必须拿不同的钥匙), 然后前往办公室. 他们每走单位长度花费一秒钟, 假设他们同时出发, 请问他们最少需要多少时间, 可以所有人都到达办公室?

Input

第 1 行包含 3 个整数 n , k 和 p ;

接下来 1 行包含 n 个整数 a_1, a_2, \dots, a_n ;

接下来 1 行包含 k 个整数 b_1, b_2, \dots, b_n ;

Output

输出距离和的最小值.

Sample

<code>keys.in</code>	<code>keys.out</code>
2 4 50 20 100 60 10 40 80	50

样例解释: 位置为 20 的人应该去拿位置为 40 的钥匙, 然后前往办公室 50, 走的距离是 $20 + 10 = 30$, 位置为 100 的人应该去位置位 80 的钥匙然后前往办公室 50, 走的距离是 50, 故花费 $\max(30, 50) = 50$ 秒就可以了.

<code>keys.in</code>	<code>keys.out</code>
1 2 10 11 15 7	7

样例解释: 选择位置为 7 的钥匙最优秀, 时间为: $(11 - 7) + (10 - 7) = 7$.

Note

- 对于 30% 的数据, $1 \leq n \leq 100$, $n \leq k \leq 200$;
- 对于 100% 的数据, $1 \leq n \leq 5000$, $n \leq k \leq 10000$, $1 \leq a_i, b_i, p \leq 10^9$.

Problem 2. cards

Input file: `cards.in`
Output file: `cards.out`
Time limit: 1 second
Memory limit: 256 MB

给你 n 张卡牌, 每张卡牌上有一个整数, 整数可以相同.

现在把这 n 张卡牌按顺序叠成一叠, 每次, 我们将最上面的一张拿出来查看, 如果它上面的数是当前剩余卡牌的最小值, 则把它拿出来, 否则将它放到卡堆的最后.

问我们需要查看多少次, 才能将所有卡牌都拿完.

Input

第 1 行包含一个整数 n 表示桌上的卡牌数;

下一行包含 n 个数 a_1, a_2, \dots, a_n 表示卡牌堆从上到下的数字.

Output

输出一个整数表示需要查看多少次牌.

Sample

<code>cards.in</code>	<code>cards.out</code>
4 6 3 1 2	7

样例说明: 依次查看:6, 3, 1(拿出), 2(拿出), 6, 3(拿出), 6(拿出), 故需要查看 7 次.

<code>cards.in</code>	<code>cards.out</code>
1 1000	1
<code>cards.in</code>	<code>cards.out</code>
7 3 3 3 3 3 3 3	7

Note

- 对于 30% 的数据, $1 \leq n \leq 5000$.
- 对于 100% 的数据, $1 \leq n \leq 10^5$, $1 \leq a_i \leq 10^5$.

Problem 3. bamboo

Input file: bamboo.in
Output file: bamboo.out
Time limit: 1 second
Memory limit: 256 MB

我们来种竹子吧.

现在有 n 棵竹子被种下了, 为了美观, 我们希望最后竹子的高度分别为 a_1, a_2, \dots, a_n .

现在已知, 每天竹子都会长高 1cm, 如果竹子被砍过一次, 那么它就不会再生长.

我们在种下竹子后, 每 d 天去看它们一次, 如果发现某棵竹子高度够了 (当前高度大于等于我们希望的高度), 我们就将它多余的部分砍掉 (如果刚好合适, 我们也算作砍过), 我们希望我们砍下来的竹节的长度小于等于 k , 请问我们的 d 最大可以取多大?

Input

第 1 行包含 2 个整数 n 和 k ;

第 2 行包含 n 个整数 a_1, a_2, \dots, a_n .

Output

输出一行, 包含最大的整数 d .

Sample

bamboo.in	bamboo.out
3 4 1 3 5	3

样例解释: 第 3 天的时候, 三个竹子的高度分别为 3, 3, 3, 我们发现前两棵够了, 分别砍下 2, 0 长度的竹节, 第 6 天的时候, 第三棵竹子长到了 6 的高度, 我们砍下 1 长度的竹节, 最终砍下了 3 长度的竹节.

bamboo.in	bamboo.out
3 40 10 30 50	32

Note

- 对于 30% 的数据, $1 \leq a_i \leq 100$, $1 \leq k \leq 10000$;
- 对于 100% 的数据, $1 \leq n \leq 100$, $1 \leq a_i \leq 10^9$, $1 \leq k \leq 10^{11}$, 保证每个点不相同.