

纪念品盒

IOI2015开幕式正在进行最后一个环节。按计划开幕式期间，每个代表队都将收到由主办方发放的一个装有纪念品的盒子。然而所有志愿者都被精彩的开幕式所吸引，除Aman外其他人完全忘记了发放纪念品这件事。Aman是一位热情的志愿者，为使得IOI尽量圆满，他要用的最短时间将所有纪念品发放出去。

开幕式的场地是一个圆环，被分为 L 个完全相等的区域，这些区域的编号依次为 0 到 $L-1$ ，也就是说，对于 $0 \leq i \leq L-2$ ，区域 i 与区域 $i+1$ 相邻，且区域 $L-1$ 与区域 0 相邻。场地上共有 N 个代表队，每队坐在上面的一个区域上，每个区域可以包含任意多个代表队，也可以为空。

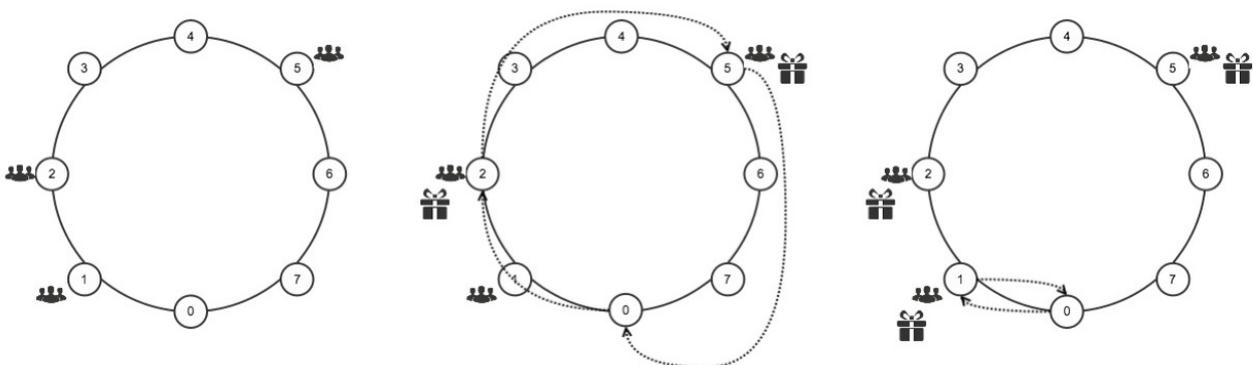
共有 N 个相同的纪念品。开始，Aman和所有纪念品都在区域 0 。Aman应该给每队一个纪念品，并且在发放完最后一个纪念品后他必须回到区域 0 。注意，有些队可能坐在区域 0 。

在任意时刻，Aman只能够携带至多 K 个纪念品。Aman必须从区域 0 取走这些纪念品，且取纪念品不需要时间。纪念品一旦从区域 0 被取走后，Aman只能将其发放给某个代表队或者随身携带。无论何时，Aman携带一个或更多的纪念品到达一个这样的区域，该区域有一个代表队尚未收到纪念品，Aman便可将他携带的一个纪念品发给这个代表队。这种发放也在瞬间完成。他所花的时间都消耗在区域之间的移动上。无论携带多少纪念品，Aman都需要1秒钟从一个区域移动到其相邻的区域（可以顺时针移动也可以逆时针移动）。

你的任务是计算出Aman发放完所有纪念品并返回到他的最初区域所需要的最短时间（秒数）。

样例

这个样例中，代表队的数目 $N=3$ ，Aman携带的纪念品数量 $K=2$ ，区域的数目 $L=8$ 。这些代表队分别位于区域 1 ， 2 ，和 5 。



一个最优解如上图所示。第一步，Aman取两个纪念品，将其中的一个纪念品发给区域 2 的队，然后将另一个纪念品发给区域 5 的队，并且返回到区域 0 。这趟发放共花费 8 秒钟。第二步，Aman取走剩下的纪念品发给区域 1 的队，然后返回到区域 0 。完成第二步他需要 2 秒钟。因此，总用时为 10 秒钟。

任务

对于给定的 N, K, L ，以及所有代表队所在的区域，计算 Aman 发放完所有纪念品并回到区域 0 所需的最短时间（秒数）。你需要实现函数 `delivery`:

- `delivery(N, K, L, positions)` — 评测程序将调用这个函数恰好一次。
 - N : 代表队的数目。
 - K : Aman 在同一时间能够携带纪念品的最大数目。
 - L : 开幕式场地上的区域数目。
 - `positions`: 一个长度为 N 的数组, `positions[0], ..., positions[N-1]` 给出了所有代表队所在区域的编号。 `positions` 的元素按非递减排序。
 - 该函数应当返回 Aman 能够完成这一任务所需的最短时间（秒数）。

子任务

子任务	分数	N	K	L
1	10	$1 \leq N \leq 1,000$	$K = 1$	$1 \leq L \leq 10^9$
2	10	$1 \leq N \leq 1,000$	$K = N$	$1 \leq L \leq 10^9$
3	15	$1 \leq N \leq 10$	$1 \leq K \leq N$	$1 \leq L \leq 10^9$
4	15	$1 \leq N \leq 1,000$	$1 \leq K \leq N$	$1 \leq L \leq 10^9$
5	20	$1 \leq N \leq 10^6$	$1 \leq K \leq 3,000$	$1 \leq L \leq 10^9$
6	30	$1 \leq N \leq 10^7$	$1 \leq K \leq N$	$1 \leq L \leq 10^9$

Sample grader

Sample grader 按照下列格式读取输入：

- 第一行: $N K L$
- 第二行: `positions[0] ... positions[N-1]`

Sample grader 输出函数 `delivery` 的返回值。