题目大意

有一个大小为 n 的牌堆，其中有一张已知位置的关键牌，分值为 1。给定 k，双方轮流行动，每人抽 k 张牌，然后选 0 到 1 张移除，剩下的牌放在牌堆底。如果关键牌被任何人放回牌堆底，则分值加 1；如果先手移除关键牌，游戏立即结束且得分为关键牌分值；如果后手移除关键牌，游戏立即结束且得分为 0。先手要最大化分值，后手要最小化分值，求游戏结果。

题解

本题的原型来自于桌游《Gizmos》（中文译名爆珠发明）。

下面称先手为 Alice，后手为 Bob。

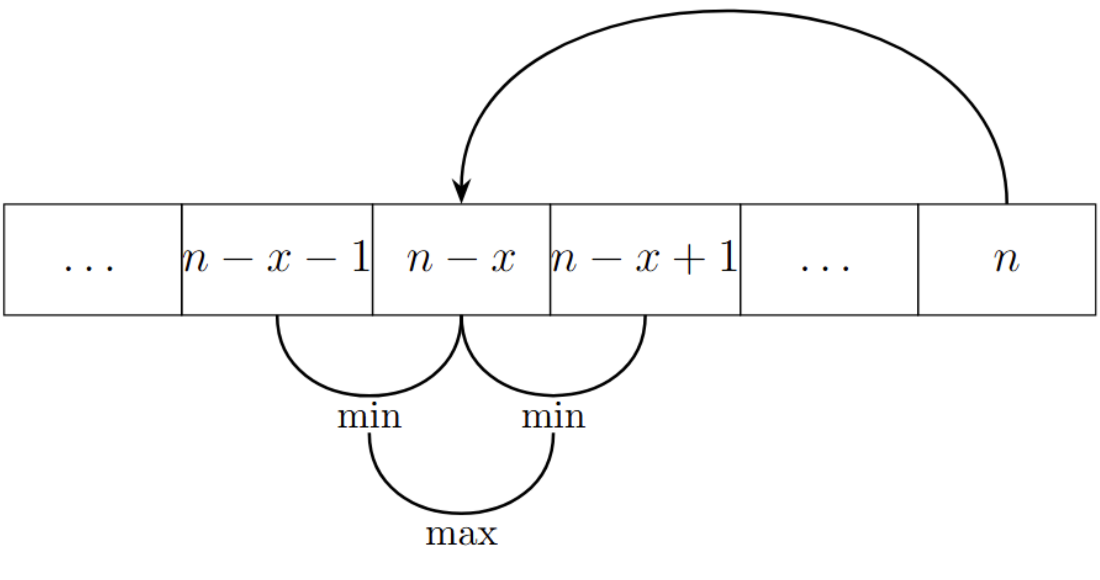
游戏肯定不会无限进行下去。因为当 $n \le k$ 时 Alice 必须立刻结束游戏，否则 Bob 就会在下一回合移除关键牌来使得分变为 0。而且 Bob 可以一直移除牌来使得牌堆越来越薄。所以 Bob 是有能力阻止 Alice 一直将游戏进行下去的。

首先考虑 $s = 1$ 的情况。这时 Alice 必须决定是拿走关键牌还是把它塞回去。一旦把它塞回去，接下来是谁拿到关键牌就只和它的位置模 $2k$ 的结果有关。她可以在连续的最多 k 个位置里选择一个，所以只有当 $n \equiv k \pmod{2k}$ 的时候她无法在下一次拿到关键牌，此时直接移除关键牌是最优的。

否则 Alice 都有办法让自己在下一次拿到这张关键牌，于是塞回去之后的得分一定比直接移除更高。

从 Alice 把关键牌塞回去这一行动开始，双方会各进行 $x = \left\lfloor \frac{n+k}{2k} \right\rfloor$ 次操作，然后又会变成 Alice 能够摸到关键牌的情况。关键的是，牌堆的大小会变成多少呢？

设 $f_k(n)$ 为牌堆大小为 n，$s = 1$，$k = k$ 时的最终分数。接下来的每次操作中，双方都可以选择移除一张牌或不移除牌。我们可以证明，最后的得分会是 $1 + \max\left\{\min\{f_k(n-x-1), f_k(n-x)\}, \min\{f_k(n-x), f_k(n-x+1)\}\right\}$。或者换一种写法，就是，$1 + \min\{\max\{f_k(n-x-1), f_k(n-x+1)\}, f_k(n-x)\}$。



一共有 $2x$ 次操作，可能会移除 $0 \sim 2x$ 张牌。我们移动一下坐标系，看成，一开始最后会剩 $n - x$ 张牌。一个人移除牌就相当于减少 0.5 张牌，不移除就是增加 0.5 张牌。

对于 Alice 来说，她第一步可以选择往左或者往右走 0.5 步，然后让 Bob 在相邻的两个当中选一个。

Bob 一定可以保证得分小于等于二者之间较小的那个，只要之后他所有操作和 Alice 相反就可以。而 Alice 也一定可以保证大于等于这二者之间较小的那个，因为她可以每次都让 Bob 做一次一模一样的选择。所以最后的结果就是 $1+\min\{\max\{f_k(n-x-1), f_k(n-x+1)\}, f_k(n-x)\}$。

但是，有一种情况比较特殊：当 $n \equiv k+1 \pmod{2k}$ 的时候，Alice 为了自己能在下一次摸到关键牌，她唯一的选择是不移除牌并且将关键牌放在牌堆底。所以这时 Alice 的第一步只有往右走一种选择，于是此时答案变成了 $1+\min\left\{f_k(n-x), f_k(n-x+1)\right\}$。

然后我们可以发现一个性质，就是当 $f_k(n-1)<f_k(n)>f_k(n+1)$ 的时候，必有 $n \equiv k-1 \pmod{2k}$。证明在这里略去，可以看 Codeforces 上的题解。此时由于 $f_k(n-x-1)$ 一定大于等于 $f_k(n-x+1)=1$，所以可以写成 $1+\min\left\{f_k(n-x), f_k(n-x-1)\right\}$。于是就有了下面的递推式：

$$f_k(n)=\begin{cases}1, & n \equiv k \pmod{2k} \\ 1+\min\left\{f_k(n-x), f_k(n-x+1)\right\}, & n \equiv k+1 \pmod{2k} \\ 1+\min\left\{f_k(n-x), f_k(n-x-1)\right\}, & n \equiv k-1+x \not\equiv k+1 \pmod{2k} \\ 1+f_k(n-x), & \text{otherwise}\end{cases}$$

注意到当 k 比较大的时候 x 很小，所以对于最后一种情况，可以在 x 相同的地方加速一下。也就是考虑连续跳 $\left\lfloor\frac{(n+k) \bmod 2k}{x}\right\rfloor$ 步。显然途中不会经过前三种情况。如果这个值是 0，那么就正常跳一步。

接下来考虑 $s \neq 1$ 的情况。也就是说，距离关键牌浮出表面，双方还要各行动 $x=\left\lfloor\frac{s-1}{2k}\right\rfloor$ 次。于是按照之前的分析，答案就是 $\min\{\max\{f_k(n-x-1), f_k(n-x+1)\}, f_k(n-x)\}$。

递推用记忆化搜索即可。时间复杂度是 $O(\sqrt{n})$。大致就是每层只会访问到不超过两个状态。具体的证明可以看 Codeforces 的题解。