Subtask 1 (16 points):

In this subtask we can sort the cards in 51 swaps, so the second player can choose $target$-th card and be sure that the guess is correct.

Subtask 2 (20 points):

Here, since we have S swaps and $N - S$ guesses, the first player can put first S cards on corresponding indexes and left all others in arbitrary places, if $target \leq S$, the second player will reveal $target$-th card and guess first try. Otherwise in $N - S$ guesses the unsorted part can be fully searched.

Subtask 3 (22 points):

In this subtask we have $\frac{n}{4}$ swaps and $\frac{n}{2} + 1$ guesses. We can divide our deck in two predetermined parts (for example, odd and even indexes or first and second halves). For simplicity, we take odd and even indexes. Now we should observe the deck and see how many odd-valued cards are in even indexes (it's easy to see that this number will be equal to the number of even-valued cards on odd indexes). It's obvious, that either that number will be less or equal to $\frac{n}{4}$, or the number of even-valued cards on even indexes. So, in at most $\frac{n}{4}$ swaps, we can either put all the even-valued cards, or all the odd-valued cards on even indexes (after that, all the values of the odd indexed cards will have the same parity too).

On the first guess, the second player should check if odd indexes hold odd-valued cards by revealing the first card and after that, we have $\frac{n}{2}$ tries and $\frac{n}{2}$ possible locations of $target$, so the problem can be solved by full search in corresponding half.

Subtask 4 (18 points):

In this subtask we have 1 swap and $\frac{n}{2}$ guesses. If we interpret our deck as a graph, where i-th card is connected to the v_i-th one (where v_i is the value of the i-th card), we will get a graph, where every card is a part of exactly one simple cycle. If we follow the cycle of card t by first revealing $target$-th card, then v_t-th, v_{v_t}-th and so on, we will find the card $target$ in exactly s_t guesses (where s_t is the size of that cycle), because the last card of the cycle should point to index $target$, which means it has a value of $target$.

The main issue of this solution is that we can have cycles larger than $\frac{n}{2}$, but we can't have more than one such cycle for obvious reasons. If such cycle exists, we can swap $\frac{n}{2}+1$-th and the first element of cycle (you can enumerate the elements of the cycle any way you want as long as i-th card points to $i+1$-th and the last one points to the first). After the swap, $\frac{n}{2}$-th card will still point to the index of $\frac{n}{2}+1$-th card, but now that card points to the second one, so now this means that the size of one of newly formed cycle is exactly $\frac{n}{2}$, while the size of remaining cycle is $s_i - \frac{n}{2} \leq \frac{n}{2}$, so now the second player can definitely find the desired card in time.

Subtask 5 (24 points):

This subtask is just a generalization of the last one. If we have S swaps and T guesses, on each swap we can take one of the cycles with size greater than T (if one exists) and cut the T-sized cycle out of it. Algorithm of the second player remains exactly the same.