Pure Straight

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Given a sequence of N positive integers $A = (A_1, A_2, \ldots, A_N)$, where each A_i is 1, 2, ..., or K.

You can do the following operation on this sequence any number of times.

- Swap two adjacent elements, that is, choose i and j such that $|i - j| = 1$ and swap A_i and A_j.

Find the minimum number of operations needed to make A satisfy the following condition

- A contains $(1, 2, \ldots, K)$ as a contiguous subsequence, that is, there is a positive integer x at most $N - K + 1$ such that $A_x = 1, A_{x+1} = 2, \ldots, A_{x+K-1} = K$

It is guaranteed that the input given will be such that it is possible to satisfy the condition after a finite number of operations.

Input

The first line contains 2 integers N and K ($K \leq N \leq 200$, $2 \leq K \leq 16$) — the size of the sequence and the range of A_i.

The second line contains N integers A_i ($1 \leq A_i \leq K$) — the elements of the sequence

Output

Print a single integer representing the minimum number of operations needed to make A satisfy the condition.

Examples

standard input	standard output
4 3 3 1 2 1	2
5 5 4 1 5 2 3	5
8 4 4 2 3 2 4 2 1 4	5

Note

For the first example, one optimal sequence of operations is as follows:

- Swap A_1 and A_2
- Swap A_2 and A_3