

Lucky Numbers (luck)

Sick of losing every lottery he participates in, Giorgio decided to stop relying on chance... and instead relying on a seer to find his lucky number once and for all! However, unexpectedly, the old witch found that Giorgio has actually a whopping K lucky numbers, each of them 4-digits long.

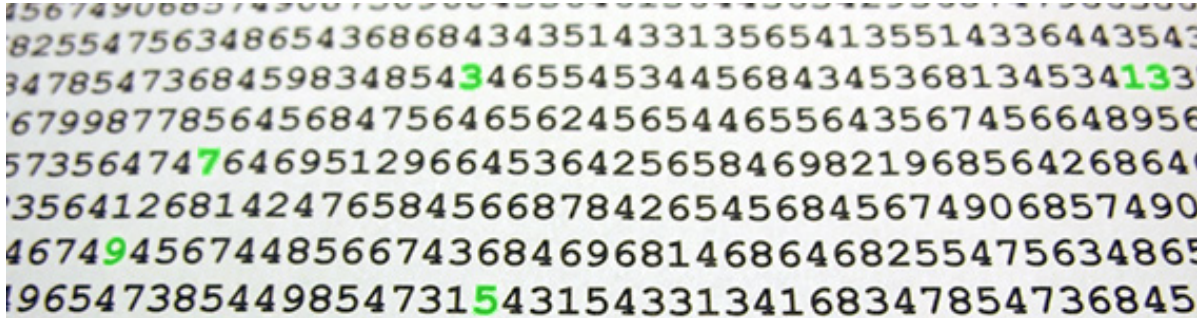


Figure 1: A very long serial number.

Thrilled by this unexpected news, Giorgio can't wait to buy a lottery ticket... but he wants the luckiest one! Given that the serial number of lottery tickets are C -digits long, find out the one with the highest amount of lucky numbers in it.

🔔 Note that a single lucky number can appear more than once and even share digits with other lucky numbers.

📎 Among the attachments of this task you may find a template file `luck.*` with a sample incomplete implementation.

Input

The first line contains two integers K and C . The second line contains K integers L_i , the lucky numbers.

Output

You need to write a single line with an integer of C digits.

Constraints

- $1 \leq K \leq 200$.
- $4 \leq C \leq 10\,000$.
- $1000 \leq L_i \leq 9999$ for each $i = 0 \dots K - 1$.

Scoring

Your program will be tested against several test cases grouped in subtasks. The score of a subtask is the minimum of the scores of its test cases, multiplied by the value of the subtask.

The score of a test case is computed using the following formula. Let O_{found} be the amount of occurrences of lucky numbers found by your solution, and O_{opt} be the optimal amount. Your score will be:

- 0 if you don't output a number of C digits; otherwise,
- 1 if $O_{\text{found}} = O_{\text{opt}}$; otherwise,
- 0 if $O_{\text{found}} \leq \left\lfloor \frac{C}{4} \right\rfloor$, and $\frac{O_{\text{found}} - \lfloor C/4 \rfloor}{O_{\text{opt}} - \lfloor C/4 \rfloor}$ otherwise.

- Subtask 1 (0 points)

Examples.
- Subtask 2 (10 points)

$K = 1$ and all the digits are different.
- Subtask 3 (20 points)

$K = 1$.
- Subtask 4 (30 points)

$K \leq 5, C \leq 30$.
- Subtask 5 (10 points)

$K \leq 5, C \leq 100$.
- Subtask 6 (10 points)

$K \leq 10$.
- Subtask 7 (20 points)

No additional limitations.

Examples

input	output
2 9 1010 1031	101031010
3 9 1122 2333 1111	111111111

Explanation

In the **first sample case**, the lucky numbers inside the serial number are: 101031010 with a total of 3 numbers. Another possible solution is 101010101 with 3 repetitions of the number 1010.

In the **second sample case**, the optimal solution is to repeat the number 1111, obtaining a total of 6 overlapping repetitions of 1111 in the resulting serial number.