Problem HIKINGTRIPS: Hiking Trips

Marty and Erica go hiking every weekend. Each of them has a knapsack. As experienced hikers they share items like a map to reduce weight. Furthermore, they distribute all items over the two knapsacks in a greedy manner: The items are in an initial order, which will not be changed. For each item they weigh the two knapsacks and put the item in the lighter knapsack. Obviously, this approach will not necessarily result in an optimal distribution of the items, i.e. where the knapsack weights are as equal as possible. Marty is a real gentleman and therefore always picks the heavier knapsack.

Someday Marty thinks about the process and reckons that sorting the items from heaviest to lightest might result in a smaller difference in weight between the two knapsacks. Erica is not delighted by Marty's idea and plans to convince him to go back to the greedy approach. Her plan is to present Marty a series of item weights a_i in an order far from descending, where the greedy strategy leads to equal weights.

Erica is not very successful in generating those sequences, so she needs your help. Create a sequence consisting of unique positive integers such that the greedy approach leads to equal knapsack weights. Erica wants the *smallest* sequence, because she thinks this series is farthest from descending order. A series $a_0, a_1, \ldots, a_N$ is considered smaller than a series $b_0, b_1, \ldots, b_N$, if there is an integer p such that $a_i = b_i$ for all $i < p$ and $a_p < b_p$.

Input

The first line contains two integers N and Q. N represents the number of items in the sequence you have to generate ($3 \leq N \leq 2^{60}$) and Q the number of positions Erica wants to know ($1 \leq Q \leq 100\,000$). The following Q lines contain a single integer i each ($0 \leq i < N$), the 0-based index of the position Erica wants to know. No two items may have the same weight and all weights have to be positive integers.

Output

Print Q lines of output. Each line should contain the weight of the item at the requested position.

Sample Input 1	Sample Output 1
7 2	1
0	9
5	

Sample Input 2	Sample Output 2
4 5	1
0	2
1	4
2	3
3	4
2	

Sample Input 3	Sample Output 3
987654321123456789 1	987654321124
987654321123	