

## Lisseq

Strong is a very strong weight-lifter. He has had many impressive performances, including one-handedly lifting 15 people. Currently, he is just doing normal weight-lifting.

Strong has  $n$  weights. The  $i^{th}$  weight weighs  $k_i$  kg. Everyday, he picks one **longest increasing subsequence (LIS)** of his weights and lift them. To make it less boring, he doesn't pick the same LIS on any 2 different days. After  $T$  days of lifting, Strong has ran out of ways to choose a LIS, and decided to move on to *Clay pot* lifting.



You want to be as strong as Strong, so you have asked him for the weights.

Strong answered:

“If you want these weights, determine the weight of such weights!”

## Input

Each input contains multiple tests.

The first line of the input contains 3 numbers  $L, R, N$

## Output

For each  $T$  from  $L$  to  $R$ , you have to determine the answer for  $T$ . In other words, you have to find a sequence of length  $N$  that satisfies: Strong could use these weight for a maximum of exactly  $T$  days, if he picks a different LIS everyday. If there are multiple answers, you can print any.

## Constraints

- $L, R \leq 1,000,000,000, R = L + 99$
- $N = 100$
- The weight of the weights must be between 0 and 100.

## Subtask

- Subtask 1 (15%):  $L, R \leq 2,000$
- Subtask 2 (30%):  $L, R \leq 50,000$
- Subtask 3 (55%): No additional constraints.

## Sample

### Input

3 5 5

**Note:** The sample test is intentionally made not satisfactory to the constraints, and is not used for marking.

### Output

2 1 0 3 4  
3 2 1 0 4  
1 1 1 1 1

Explanation: The LIS are:

3 : (1, 4, 5), (2, 4, 5), (3, 4, 5)

4 : (1, 5), (2, 5), (3, 5), (4, 5)

5 : (1), (2), (3), (4), (5)