Walking In The Park (sumpartition)

Bence is casually walking in the park when he stumbles over something. When he looks down he is stunned, he stumbled over two arrays!

Figure 1: Bence after stumbling over the arrays.

The first array A contains N integers while the second array B contains M **positive** integers. Bence wants to take revenge on the arrays and destroy them. However, he wants to destroy them in a very specific way: he wants to split each array into K non-empty contiguous subarrays, such that the sum of elements within each corresponding pair of subarrays is precisely the same.

More formally, Bence wants to partition A into subarrays $X_0, \ldots, X_{K-1}$ and B into subarrays $Y_0, \ldots, Y_{K-1}$, such that the sum of elements in X_i equals the sum of elements in Y_i for each $i = 0 \ldots K-1$.

Help Bence to find such a partition, or tell him if it is impossible.

Input

The input file consists of:

- a line containing integers N, M, K.
- a line containing the N integers $A_0, \ldots, A_{N-1}$.
- a line containing the M integers $B_0, \ldots, B_{M-1}$.

Output

If there exists a suitable partition, the output file must contain **two lines**:

- a line containing $K-1$ integers: the starting indices of the subarrays $X_1, \ldots, X_{K-1}$.
- a line containing $K-1$ integers: the starting indices of the subarrays $Y_1, \ldots, Y_{K-1}$.

If there is no suitable partition, you have to **print a single line** containing -1.

Constraints

- $2 \leq N, M \leq 200\,000$.
- $2 \leq K \leq \min(N, M)$.
- $-10^{12} \leq A_i \leq 10^{12}$ for each $i = 0 \ldots N-1$.
- $1 \leq B_i \leq 10^6$ for each $i = 0 \ldots M-1$.

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

– **Subtask 1** (0 points) Examples.

– **Subtask 2** (19 points) $A_i > 0$ for each $i = 0 \ldots N-1$.

– **Subtask 3** (23 points) $N, M \leq 100$.

– **Subtask 4** (16 points) $N, M \leq 1000$.

– **Subtask 5** (42 points) No additional limitations.

Examples

input	output
5 4 3 1 3 6 -3 9 1 9 2 4	1 3 1 2
5 4 3 2 3 6 -3 9 1 10 2 4	-1

Explanation

In the **first sample case**, Bence has to partition A into $[1]$, $[3, 6]$, $[-3, 9]$ and B into $[1]$, $[9]$, $[2, 4]$.

In the **second sample case**, there is no way to split both arrays into 3 subarrays each such that the sequences of their sums are equal.