2020 Team Formation Test (Round 1) Paper 2

Task Overview

ID	Name	Time Limit	Memory Limit	Subtasks
T2004	Divisibility	1.000 s	256 MB	8 + 8 + 16 + 28 + 40
T2005	Find the Sample Input	1.000 s	256 MB	5 + 11 + 15 + 13 + 22 + 34
T2006	Paper Cutting	1.000 s	256 MB	7 + 7 + 28 + 16 + 42

Notice:
Unless otherwise specified, inputs and outputs shall follow the format below:

- One space between a number and another number or character in the same line.
- No space between characters in the same line.
- Each string shall be placed in its own separate line.
- Outputs will be automatically fixed as follows: Trailing spaces in each line will be removed and an end-of-line character will be added to the end of the output if not present. All other format errors will not be fixed.

C++ programmers should be aware that using C++ streams (`cin` / `cout`) may lead to I/O bottlenecks and substantially lower performance.

For some problems 64-bit integers may be required. In Pascal it is `int64`. In C/C++ it is `long long` and its token for `scanf` /`printf` is `%lld`.

All tasks are divided into subtasks. You need to pass all test cases in a subtask to get points.

T2004 - DIVISIBILITY

Time Limit: 1.000 s / Memory Limit: 256 MB

The second semester has started and Marisa registered the course MATH3145, "Introduction to Number Theory". Of course, this course is very hard and many students cannot handle the course.

One day, Professor Cirno discussed the topic "Divisibility". Since Cirno wanted to challenge her students, she came up with a task that expected no students sucessfully solving it and said that whoever solves the task will get an A grade from MATH3145. The task is as follows:

Given a string S with N numerical digits (`0` to `9` inclusive). You are also given an integer K, and have to split S into exactly K **non-empty** substrings $p_1, p_2, \ldots, p_K$. A splitting is considered *nice* if and only if every of the substrings $p_1, p_2, \ldots, p_K$, when treated as a base-10 number, is divisible by D. Your task is to determine if a *nice* splitting exists.

Professor Cirno reminds you that:

- The order of the digits must not be rearranged, that is splitting $S =$ `12345` into $p_1 =$ `45` and $p_2 =$ `123` is not allowed.
- Leading zeros are **ALLOWED** in the splitting. For example, $N = 8$, $S =$ `70007000`, $K = 3$ and $D = 7$. Splitting S into $p_1 =$ `7`, $p_2 =$ `00` and $p_3 =$ `07000` is allowed.

For example, $N = 5$, $S =$ `12345`, $K = 2$ and $D = 3$:
We can split S into $p_1 =$ `123` and $p_2 =$ `45`. As both the numbers 123 and 45 are divisible by D, this is a nice splitting.
Another possible nice splitting is $p_1 =$ `12` and $p_2 =$ `345`.

Consider another example where $N = 4$, $S =$ `1024`, $K = 2$ and $D = 3$. All possible splittings are:

- $p_1 =$ `1` and $p_2 =$ `024`
- $p_1 =$ `10` and $p_2 =$ `24`
- $p_1 =$ `102` and $p_2 =$ `4`

However, none of the three splittings are considered nice. p_1 in the first two splittings and p_2 in the third splitting are not divisible by D.

Marisa wants to impress Professor Cirno by solving this task so that she can get an A grade for this course. Can you help Marisa solve this task?

INPUT

The first line contains an integer N.
The second line contains a string S which has N digits.
The third line contains an integer K.
The fourth line contains an integer D.

OUTPUT

If it is impossible to obtain a nice splitting, output `Impossible`.

Otherwise, output exactly K lines. The i-th line should contain the string p_i.
If there exists multiple nice splittings, output any one of them.

SAMPLE TESTS

	Input	Output
1	5 12345 2 3	123 45

The splitting is nice as 123 and 45 are divisible by 3.

	Input	Output
2	8 70007000 3 7	7 00 07000

Leading zeros are allowed for answer.

	Input	Output
3	5 12345 2 5	Impossible
4	7 8172232 2 2	8172 232
5	7 8172232 4 2	8 172 2 32
6	7 8172232 5 2	Impossible

SUBTASKS

For all cases:
$1 \leq N \leq 10^5$
$1 \leq K \leq N$
$1 \leq D \leq 10^8$

	Points	Constraints
1	8	$D = 10$
2	8	$D = 3$
3	16	$D = 4$
4	28	$N \leq 9$
5	40	No additional constraints

T2005 - FIND THE SAMPLE INPUT

Time Limit: 1.000 s / Memory Limit: 256 MB

HSOI (Heung Shing Olympiad in Informatics) final is coming soon. Every problem setter of HSOI is busy preparing problem statements and test data. Alice is one of the problem setters of HSOI. Today, she is working on preparation of the sample test for an interesting problem. The problem is the following:

- Given an sequence of N integers $A_1, A_2, \ldots, A_N$, where $0 \le A_i \le B$
- You need to output an sequence of N' integers $S_1, S_2, \ldots, S_{N'}$, where $N' = N - W + 1$, so that S_i equals to the sum of the W consecutive elements of A starting from the i^{th} one, i.e. $S_i = A_i + A_{i+1} + \cdots + A_{i+W-1}$.

Alice prepared a sample test case for this problem with N, W, B, sequence A as input and sequence S as output. Unfortunately, after a while, due to system error, she found that the data of the sequence A in the sample input was corrupted. However, as Alice is lazy, she doesn't want to prepare the whole sample test case from scratch again. Therefore, She wants to find a sequence A' that will produce the same output, i.e. sequence S, with given variables N, W, and B. Note that the values A'_i still have to satisfy the constraint $0 \le A'_i \le B$.

As she has already worked on the problem preparation for 40 hours and she has to slap her face to stay awake. Yet, she is not able to find such an sequence A'. Therefore, she asks you to find it for her. Can you help her?

INPUT

The first line consists of three integers N, W, and B.
The second line consists of N' integers S_i, the sum of the W consecutive elements of sequence A starting from the i^{th} one. Recall that $N' = N - W + 1$.

OUTPUT

If it is impossible to find a sequence A' that will produce the output $S_1, S_2, \ldots, S_{N'}$, output `No` on a single line.

Otherwise, output `Yes` on the first line.
On the second line, output N integers $A'_1, A'_2, \ldots, A'_N$, the elements of the sequence A'.
If there are multiple solutions, you can output any of them.

SAMPLE TESTS

	Input	Output
1	4 2 5 6 7 8	Yes 2 4 3 5

$A'_1 + A'_2 = 2 + 4 = 6 = S_1$
$A'_2 + A'_3 = 4 + 3 = 7 = S_2$
$A'_3 + A'_4 = 3 + 5 = 8 = S_3$
So $2, 4, 3, 5$ is a valid solution.

	Input	Output
2	4 2 4 8 7 8	No

SUBTASKS

For all cases:
$1 \le W \le N \le 100000$
$1 \le B \times W \le 10^9$
$0 \le S_i \le 10^9$

	Points	Constraints
1	5	$N = W = 2$
2	11	$1 \le W \le N \le 20$ $B = 1$
3	15	$2 \le N \le 100000$ $W = N - 1$
4	13	$2 \le N \le 100$ $W = 2$ $1 \le B \le 100$
5	22	$1 \le W \le N \le 2000$ $1 \le B \le 2000$
6	34	No additional constraints

T2006 - PAPER CUTTING

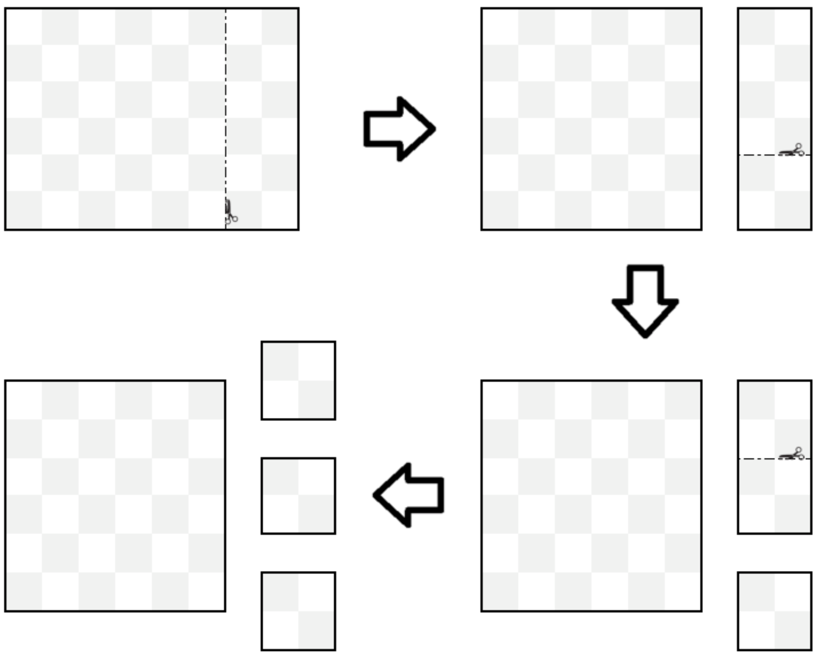
Time Limit: 1.000 s / Memory Limit: 256 MB

Alice loves squares. She loves things with square shape, e.g., square mirror, square phone and square monitor.

Today, Alice fell asleep in the class. When she woke up, she saw two pieces of rectangular paper on top of the table. She will be very annoyed if there is any piece of paper not being a square. So, she has decided to cut them into squares.

Alice perform paper cutting with the following procedure:
Suppose the dimensions of a piece of paper is L by W.
If $L = W$ then end the procedure.
If $L < W$ then cut the paper into a L by L square and a L by $(W - L)$ rectangle, repeat the procedure with the rectangle piece.
If $L > W$ then cut the paper into a W by W square and a $(L - W)$ by W rectangle, repeat the procedure with the rectangle piece.

For example, assume there is a piece of 6 by 8 paper, Alice will cut it into a 6 by 6 square and a 2 by 6 rectangle, and then cut the 2 by 6 rectangle into a 2 by 2 square and a 2 by 4 rectangle, and then cut the 2 by 4 rectangle into two 2 by 2 squares.



Alice was so satisfied to see that all pieces of the paper had become squares. Suddenly, a strong wind blew the paper away. When she was trying to get the paper back, she saw some text on the back of the paper. She realised the paper she cut was useful, so she wanted to recover the paper back. After a while, she got some squares with N different sizes back but others may be lost. She had no clue on how to recover them.

As the friend of Alice, please help her by giving a set of the possible dimensions of the two original pieces of paper by using **ALL** the squares she has, so that she may try to recover them.

INPUT

The first line consists of an integer N.

In the next N lines, the i^{th} line consists of two integers S_i and C_i, stating that Alice has C_i pieces of S_i by S_i squares.

OUTPUT

If it is impossible to recover the two orgiinal pieces of paper back, output `Impossible`.

Otherwise, output two lines. The i^{th} of the two lines contains 2 integers, L_i and W_i, stating that the dimensions of the i^{th} piece of original paper is L_i by W_i.

If there are several possible solutions, output any of them.

SAMPLE TESTS

	Input	Output
1	4 1 4 2 3 4 2 6 1	6 8 4 9
2	3 1 5 2 1 3 1	3 4 3 2
3	3 1 5 2 1 4 1	Impossible

SUBTASKS

For all cases:
$1 \le N \le 100$
$1 \le S_1 < S_2 < S_3 < \cdots < S_N \le 10^9$
$1 \le C_i \le 10^9$
If an answer exists, then it is guaranteed that there exists an answer with the dimensions of the original two pieces of paper no larger than 10^9.

	Points	Constraints
1	7	$N = 1$
2	7	$N = 2$
3	28	There exists an answer with one of the original pieces of paper being a square.
4	16	Sum of $C_i \le 8$
5	42	No additional constraints