Problem A. Streamer Takahashi

Time Limit 2000 ms

Problem Statement

Streamer Takahashi has decided to stream from L o'clock to R o'clock (using the 24-hour clock).

He has N listeners, and the i-th listener can watch the stream from X_i o'clock to Y_i o'clock.

How many listeners can watch Takahashi's stream from beginning to end?

Constraints

- $1 \leq N \leq 100$
- $0 \leq L < R \leq 23$
- $0 \leq X_i < Y_i \leq 23$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \ L \ R \\
X_1 \ Y_1 \\
X_2 \ Y_2 \\
\vdots \\
X_N \ Y_N
\end{array}
$$

Output

Output the number of listeners who can watch Takahashi's stream from beginning to end.

Sample 1

Input	Output
5 19 22 17 23 20 23 19 22 0 23 12 20	3

The listeners who can watch Takahashi's stream from beginning to end are the 1st, 3rd, and 4th listeners.

Sample 2

Input	Output
3 12 13 0 1 0 1 0 1	0

No listeners can watch Takahashi's stream from beginning to end.

Sample 3

Input	Output
10 8 14 5 20 14 21 9 21 5 23 8 10 0 14 3 8 2 6 0 16 5 20	5

Problem B. String Too Long

Time Limit 2000 ms

Problem Statement

> Restore run-length encoding. If the result is too long, output `Too Long`.

You are given N pairs of characters and integers $(c_1, l_1), (c_2, l_2), \ldots, (c_N, l_N)$.

Let S be the string formed by concatenating l_1 characters c_1, l_2 characters c_2, $\ldots$, and l_N characters c_N in this order.

Output S. However, if the length of S exceeds 100, output `Too Long` instead.

Constraints

- $1 \leq N \leq 100$
- $1 \leq l_i \leq 10^{18}$
- N and l_i are integers.
- Each c_i is a lowercase English letter.
- $c_i \neq c_{i+1}$

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \\
c_1 \ l_1 \\
c_2 \ l_2 \\
\vdots \\
c_N \ l_N
\end{array}
$$

Output

If the length of S is at most 100, output S; otherwise, output `Too Long`.

Sample 1

Input	Output
8 m 1 i 1 s 2 i 1 s 2 i 1 p 2 i 1	mississippi

S is `mississippi`. Since the length of S is not greater than 100, output S.

Sample 2

Input	Output
7 a 1000000000000000000 t 1000000000000000000 c 1000000000000000000 o 1000000000000000000 d 1000000000000000000 e 1000000000000000000 r 1000000000000000000	Too Long

The length of S is 7×10^{18}, so output `Too Long`.

Sample 3

Input	Output
1 a 100	aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa aaaaaaaaaaaaaaaaaa

Sample 4

Input	Output
6 g 4 j 1 m 4 e 4 d 3 i 4	ggggjmmmmeeeedddiiii

Problem C. Palindromic in Both Bases

Time Limit 3000 ms

Problem Statement

The decimal representation of 414 is `414`, which is a palindrome. Also, the octal representation of 414 is `636`, which is also a palindrome. Based on this, solve the following problem.

You are given positive integers A and N. Find the sum of all integers between 1 and N, inclusive, whose decimal representation and base-A representation are both palindromes.

Under the constraints of this problem, it can be proved that the answer is less than 2^{63}.

Constraints

- $2 \leq A \leq 9$
- $1 \leq N \leq 10^{12}$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

A
N

Output

Output the answer in one line.

Sample 1

Input	Output
8 1000	2155

The integers satisfying the condition are $1, 2, 3, 4, 5, 6, 7, 9, 121, 292, 333, 373, 414, 585$ (14 integers), and their sum is 2155.

Sample 2

Input	Output
8 999999999999	914703021014

Sample 3

Input	Output
6 999999999999	283958331810

Problem D. Transmission Mission

Time Limit 2000 ms

Problem Statement

There are N houses numbered from 1 to N on a number line. House i is located at coordinate X_i. Multiple houses may be located at the same coordinate.

You place M base stations at arbitrary real coordinates on the number line. Then, you set a non-negative integer **signal strength** for each base station.

When the signal strength of a base station is set to x, The signal from that base station reaches a house if and only if the distance between the base station and the house is at most $\frac{x}{2}$. Particularly, when $x = 0$, the signal reaches only houses located at the same coordinate as the base station.

Find the minimum possible sum of signal strengths when the positions and signal strengths of the base stations are set such that at least one base station's signal reaches every house.

It can be proved that the answer is an integer for any input satisfying the constraints.

Constraints

- $1 \leq M \leq N \leq 5 \times 10^5$
- $1 \leq X_i \leq 10^{12}$ $(1 \leq i \leq N)$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

```
N M
X_1 ... X_N
```

Output

Output the answer as an integer in one line.

Sample 1

Input	Output
7 3 5 10 15 20 8 14 15	6

By placing three base stations as follows, signals reach all houses.

- Place a base station with signal strength 5 at coordinate 7.5. This base station reaches houses $1, 2, 5$.
- Place a base station with signal strength 1 at coordinate 14.5. This base station reaches houses $3, 6, 7$.
- Place a base station with signal strength 0 at coordinate 20. This base station reaches house 4.

The sum of signal strengths in this case is 6.

It is impossible to satisfy the condition with an arrangement where the sum of signal strengths is smaller than 6, so output 6.

Sample 2

Input	Output
7 7 5 10 15 20 8 14 15	0

Sample 3

Input	Output
7 1 5 10 15 20 8 14 15	15

Problem E. Content Too Large

Time Limit 2000 ms

Problem Statement

Takahashi has N items and one bag.

The size of the i-th $(1 \leq i \leq N)$ item is A_i, and the size of the bag is M.

If and only if the total size of the items he is trying to put in the bag is at most M, he can put all those items in the bag simultaneously.

If he can put all N items in the bag simultaneously, print `Yes`; otherwise, print `No`.

Constraints

- $1 \leq N \leq 100$
- $1 \leq M \leq 10000$
- $1 \leq A_i \leq 100$ $(1 \leq i \leq N)$
- All input values are integers.

Input

The input is given from standard input in the following format:

$$
\begin{array}{l}
N \ M \\
A_1 \ A_2 \ \ldots \ A_N
\end{array}
$$

Output

If Takahashi can put all items in the bag simultaneously, print `Yes`; otherwise, print `No`.

Sample 1

Input	Output
5 15 3 1 4 1 5	Yes

The total size of the 5 items is $3 + 1 + 4 + 1 + 5 = 14$. Since this is not greater than the bag size 15, Takahashi can put all items in the bag simultaneously. Thus, print `Yes`.

Sample 2

Input	Output
5 5 3 1 4 1 5	No

The total size of the 5 items is 14, which is greater than the bag size 5, so he cannot put all items in the bag simultaneously. Thus, print `No`.

Sample 3

Input	Output
1 10000 100	Yes

Problem F. cat 2

Time Limit 2000 ms

Problem Statement

You are given N types of strings $S_1, S_2, \ldots, S_N$.

You perform the following operation once:

- Choose **distinct** integers i and j $(1 \leq i \leq N, 1 \leq j \leq N)$ and concatenate S_i and S_j in this order.

How many different strings can be obtained as a result of this operation?

If different choices of (i, j) result in the same concatenated string, it is counted as one string.

Constraints

- $1 \leq N \leq 100$
- N is an integer.
- S_i is a string of length between 1 and 10, inclusive, consisting of lowercase English letters.
- $S_i \neq S_j$ $(1 \leq i < j \leq N)$

Input

The input is given from standard input in the following format:

$$
\begin{aligned}
&N \\
&S_1 \\
&S_2 \\
&\vdots \\
&S_N
\end{aligned}
$$

Output

Print the number of different strings that can be obtained from the operation.

Sample 1

Input	Output
4 at atco coder der	11

The possible strings are `atatco`, `atcoat`, `atcoder`, `atcocoder`, `atder`, `coderat`, `coderatco`, `coderder`, `derat`, `deratco`, `dercoder`, which are 11 strings.

Thus, print `11`.

Sample 2

Input	Output
5 a aa aaa aaaa aaaaa	7

The possible strings are `aaa`, `aaaa`, `aaaaa`, `aaaaaa`, `aaaaaaa`, `aaaaaaaa`, `aaaaaaaaa`, which are 7 strings.

Thus, print `7`.

Sample 3

Input	Output
10 armiearggc ukupaunpiy cogzmjmiob rtwbvmtruq qapfzsitbl vhkihnipny ybonzypnsn esxvgoudra usngxmaqpt yfseonwhgp	90

Problem G. Large Queue

Time Limit 2000 ms

Problem Statement

There is an empty integer sequence $A = ()$. You are given Q queries, and you need to process them in the given order. There are two types of queries:

- Type 1: Given in the format `1 c x`. Add c copies of x to the end of A.
- Type 2: Given in the format `2 k`. Remove the first k elements from A and output the sum of the removed k integers. It is guaranteed that k is at most the length of A at that time.

Constraints

- $1 \leq Q \leq 2 \times 10^5$
- In type 1 queries, $1 \leq c \leq 10^9$.
- In type 1 queries, $1 \leq x \leq 10^9$.
- In type 2 queries, letting n be the length of A at that time, $1 \leq k \leq \min(10^9, n)$.
- All input values are integers.

Input

The input is given from standard input in the following format:

$$
\begin{array}{l}
Q \\
\mathrm{query}_1 \\
\mathrm{query}_2 \\
\vdots \\
\mathrm{query}_Q
\end{array}
$$

where query_i represents the i-th query and is in one of the following formats:

$$1 \ c \ x$$

$$2 \ k$$

Output

Let q be the number of type 2 queries. Output q lines. The i-th line should contain the answer to the i-th type 2 query.

Sample 1

Input	Output
5 1 2 3 1 4 5 2 3 1 6 2 2 5	11 19

- 1st query: Add 2 copies of 3 to the end of A. Then, $A = (3, 3)$.
- 2nd query: Add 4 copies of 5 to the end of A. Then, $A = (3, 3, 5, 5, 5, 5)$.
- 3rd query: Remove the first 3 elements from A. Then, the sum of the removed 3 integers is $3 + 3 + 5 = 11$, so output 11. After removal, $A = (5, 5, 5)$.
- 4th query: Add 6 copies of 2 to the end of A. Then, $A = (5, 5, 5, 2, 2, 2, 2, 2, 2)$.
- 5th query: Remove the first 5 elements from A. Then, the sum of the removed 5 integers is $5 + 5 + 5 + 2 + 2 = 19$, so output 19. After removal, $A = (2, 2, 2, 2)$.

Sample 2

Input	Output
10 1 75 22 1 81 72 1 2 97 1 84 82 1 2 32 1 39 57 2 45 1 40 16 2 32 2 42	990 804 3024

Sample 3

Input	Output
10 1 160449218 954291757 2 17217760 1 353195922 501899080 1 350034067 910748511 1 824284691 470338674 2 180999835 1 131381221 677959980 1 346948152 208032501 1 893229302 506147731 2 298309896	16430766442004320 155640513381884866 149721462357295680

Problem H. Make Geometric Sequence

Time Limit 2000 ms

Problem Statement

You are given an integer sequence $A = (A_1, A_2, \ldots, A_N)$ of length N. It is guaranteed that for any i $(1 \le i \le N)$, A_i is not 0.

Determine whether there exists a permutation $B = (B_1, B_2, \ldots, B_N)$ of A such that B forms a geometric sequence.

A sequence $S = (S_1, S_2, \ldots, S_N)$ is a geometric sequence if there exists a real number r such that $S_{i+1} = rS_i$ for all integers $1 \le i < N$.

Solve T test cases per input file.

Constraints

- $1 \le T \le 10^5$
- $2 \le N \le 2 \times 10^5$
- $-10^9 \le A_i \le 10^9$ $(1 \le i \le N)$
- $A_i \ne 0$ $(1 \le i \le N)$
- The sum of N over all test cases in a single input file is at most 2×10^5.
- All input values are integers.

Input

The input is given from standard input in the following format:

$$
\begin{array}{l}
T \\
\mathrm{testcase}_1 \\
\mathrm{testcase}_2 \\
\vdots \\
\mathrm{testcase}_T
\end{array}
$$

where $\mathrm{testcase}_i$ is the i-th test case $(1 \le i \le T)$, and each test case is given in the following format:

N
$A_1\ A_2\ \ldots\ A_N$

Output

Output T lines. The i-th line $(1 \leq i \leq T)$ should contain `Yes` if A can be rearranged to form a geometric sequence in the i-th test case, and `No` otherwise.

Sample 1

Input	Output
3 5 1 8 2 4 16 5 -16 24 54 81 -36 7 90000 8100 -27000 729 -300000 -2430 1000000	Yes No Yes

In the first test case, the rearrangement $(16, 8, 4, 2, 1)$ of A forms a geometric sequence with common ratio $r = \frac{1}{2}$. Thus, print `Yes` on the first line.

In the second test case, no rearrangement of A satisfies the condition. Thus, print `No` on the second line.

Problem I. Reverse 2^i

Time Limit 2000 ms

Problem Statement

You are given a permutation $P = (P_0, P_1, \dots, P_{2^N-1})$ of $(1, 2, 3, \dots, 2^N)$.

You can perform the following operation any number of times (possibly zero):

- Choose non-negative integers a, b satisfying $0 \le a \times 2^b < (a+1) \times 2^b \le 2^N$, and reverse $P_{a\times 2^b}, P_{a\times 2^b+1}, \dots, P_{(a+1)\times 2^b-1}$. Here, reversing $P_{a\times 2^b}, P_{a\times 2^b+1}, \dots, P_{(a+1)\times 2^b-1}$ means simultaneously replacing $P_{a\times 2^b}, P_{a\times 2^b+1}, \dots, P_{(a+1)\times 2^b-1}$ with $P_{(a+1)\times 2^b-1}, P_{(a+1)\times 2^b-2}, \dots, P_{a\times 2^b}$.

Find the lexicographically smallest permutation P that can be obtained by repeating the operation.

You are given T test cases, so find the answer for each.

Constraints

- $1 \le T \le 10^5$
- $1 \le N \le 18$
- P is a permutation of $(1, 2, 3, \dots, 2^N)$.
- For each input file, the sum of 2^N over all test cases is at most 3×10^5.
- All input values are integers.

Input

The input is given from standard input in the following format:

$$\begin{array}{l} T \\ \text{case}_1 \\ \text{case}_2 \\ \vdots \\ \text{case}_T \end{array}$$

case_i represents the i-th test case and is given in the following format:

```
N
P_0 P_1 ... P_{2^N-1}
```

Output

Output T lines. The i-th line $(1 \leq i \leq T)$ should contain the answer to the i-th test case.

Sample 1

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

In the first test case, when no operation is performed on P, $P = (1, 2)$. This is the lexicographically smallest permutation. Thus, the answer is $(1, 2)$.

In the second test case, when we perform the operation with $a = 1, b = 1$, P becomes $(1, 3, 2, 4)$. No matter how many operations we perform on P, we cannot obtain a permutation lexicographically smaller than $(1, 3, 2, 4)$. Thus, the answer is $(1, 3, 2, 4)$.

In the third test case, by performing operations in the following order, we can obtain $P = (1, 4, 2, 3)$:

- Perform the operation with $a = 0, b = 1$. P becomes $(3, 2, 4, 1)$.
- Perform the operation with $a = 0, b = 2$. P becomes $(1, 4, 2, 3)$.

No matter how many operations we perform on P, we cannot obtain a permutation lexicographically smaller than $(1, 4, 2, 3)$. Thus, the answer is $(1, 4, 2, 3)$.

Problem J. No Passage

Time Limit 2500 ms

Problem Statement

There is an $H \times W$ grid. Let (i, j) denote the cell at the i-th row from the top and j-th column from the left. Among these, K cells are goals. The i-th goal $(1 \le i \le K)$ is cell (R_i, C_i).

Takahashi and Aoki play a game using this grid and a single piece placed on the grid. Takahashi and Aoki repeatedly perform the following series of operations until the piece reaches a goal cell:

- Aoki chooses an integer a between 1 and 4, inclusive.
- Then, Takahashi chooses an integer b between 1 and 4, inclusive, where $a \neq b$ must be satisfied. Let (i, j) be the cell where the piece is placed before the operation. Based on the chosen integer b and the piece's position, move the piece.
  - When $b = 1$: If $(i - 1, j)$ is within the grid, move the piece from cell (i, j) to cell $(i - 1, j)$; if it is outside the grid, do nothing.
  - When $b = 2$: If $(i + 1, j)$ is within the grid, move the piece from cell (i, j) to cell $(i + 1, j)$; if it is outside the grid, do nothing.
  - When $b = 3$: If $(i, j - 1)$ is within the grid, move the piece from cell (i, j) to cell $(i, j - 1)$; if it is outside the grid, do nothing.
  - When $b = 4$: If $(i, j + 1)$ is within the grid, move the piece from cell (i, j) to cell $(i, j + 1)$; if it is outside the grid, do nothing.

Takahashi's objective is to minimize the number of moves until the piece reaches a goal. Aoki's objective is to prevent the piece from reaching the goal; if that is impossible, his objective is to maximize the number of moves until the piece reaches a goal.

For all pairs of integers (i, j) satisfying $1 \le i \le H, 1 \le j \le W$, solve the following problem and find the sum of all solutions:

> Start the game with the piece at cell (i, j). Assume both players act optimally toward their respective objectives. If Takahashi can make the piece reach a goal, the solution is the minimum number of moves; otherwise, the solution is 0.

Constraints

- $2 \le H \le 3000$
- $2 \le W \le 3000$
- $1 \le K \le \min(HW, 3000)$
- $1 \le R_i \le H$
- $1 \le C_i \le W$
- $(R_i, C_i) \ne (R_j, C_j) \, (1 \le i < j \le K)$
- All input values are integers.

Input

The input is given from standard input in the following format:

$$
\begin{array}{l}
H \ W \ K \\
R_1 \ C_1 \\
R_2 \ C_2 \\
\vdots \\
R_K \ C_K
\end{array}
$$

Output

Print the answer.

Sample 1

Input	Output
2 3 2 1 2 2 1	2

When $(i, j) = (1, 2), (2, 1)$, the starting cell is a goal, so the solution is 0.

When $(i, j) = (1, 1), (2, 2)$, no matter which a Aoki chooses, Takahashi can make the piece reach a goal in 1 move from the starting cell, so the solution is 1.

When $(i, j) = (1, 3), (2, 3)$, Takahashi cannot reach a goal, so the solution is 0.

The sum of these is $0 \times 2 + 1 \times 2 + 0 \times 2 = 2$. Thus, print 2.

Sample 2

Input	Output
9 3 9 1 3 6 1 4 1 1 2 2 1 7 1 9 3 8 1 9 2	43

Sample 3

Input	Output
10 10 36 3 8 5 10 3 10 6 10 2 10 2 8 7 10 1 10 1 8 7 6 7 8 2 5 1 6 8 8 7 5 2 4 9 8 7 4 4 3 10 10 10 8 8 10 10 6 6 2 4 2 10 5 8 3 1 2 2 1 4 1 10 4 10 3 8 1 6 1 10 2 9 1	153

Problem K. Big Banned Grid

Time Limit 2000 ms

Problem Statement

There is an $H \times W$ grid. Let (i, j) denote the cell at the i-th row $(1 \le i \le H)$ from the top and j-th column $(1 \le j \le W)$ from the left.

Each cell in the grid either has an obstacle placed on it or has nothing placed on it. There are K cells with obstacles: cells $(r_1, c_1), (r_2, c_2), \ldots, (r_K, c_K)$.

Takahashi is initially at cell $(1, 1)$ and wants to reach cell (H, W) by repeatedly moving to an adjacent cell (up, down, left, right) that has nothing placed on it.

More precisely, he can repeat the following operation as many times as he likes:

- Choose one of the following four operations and perform the chosen operation:
  - If $1 < i$ and cell $(i-1, j)$ has nothing placed on it, move to cell $(i-1, j)$. Otherwise, do not move.
  - If $1 < j$ and cell $(i, j-1)$ has nothing placed on it, move to cell $(i, j-1)$. Otherwise, do not move.
  - If $i < H$ and cell $(i+1, j)$ has nothing placed on it, move to cell $(i+1, j)$. Otherwise, do not move.
  - If $j < W$ and cell $(i, j+1)$ has nothing placed on it, move to cell $(i, j+1)$. Otherwise, do not move.

Determine whether he can move from cell $(1, 1)$ to cell (H, W).

Constraints

- $1 \le H \le 2 \times 10^5$
- $1 \le W \le 2 \times 10^5$
- $0 \le K \le 2 \times 10^5$
- $1 \le r_i \le H$ $(1 \le i \le K)$
- $1 \le c_i \le W$ $(1 \le i \le K)$
- $(r_i, c_i) \ne (1, 1)$ $(1 \le i \le K)$
- $(r_i, c_i) \ne (H, W)$ $(1 \le i \le K)$
- $(r_i, c_i) \ne (r_j, c_j)$ $(1 \le i < j \le K)$

- All input values are integers.

Input

The input is given from standard input in the following format:

$$
\begin{array}{l}
H \ W \ K \\
r_1 \ c_1 \\
r_2 \ c_2 \\
\vdots \\
r_K \ c_K
\end{array}
$$

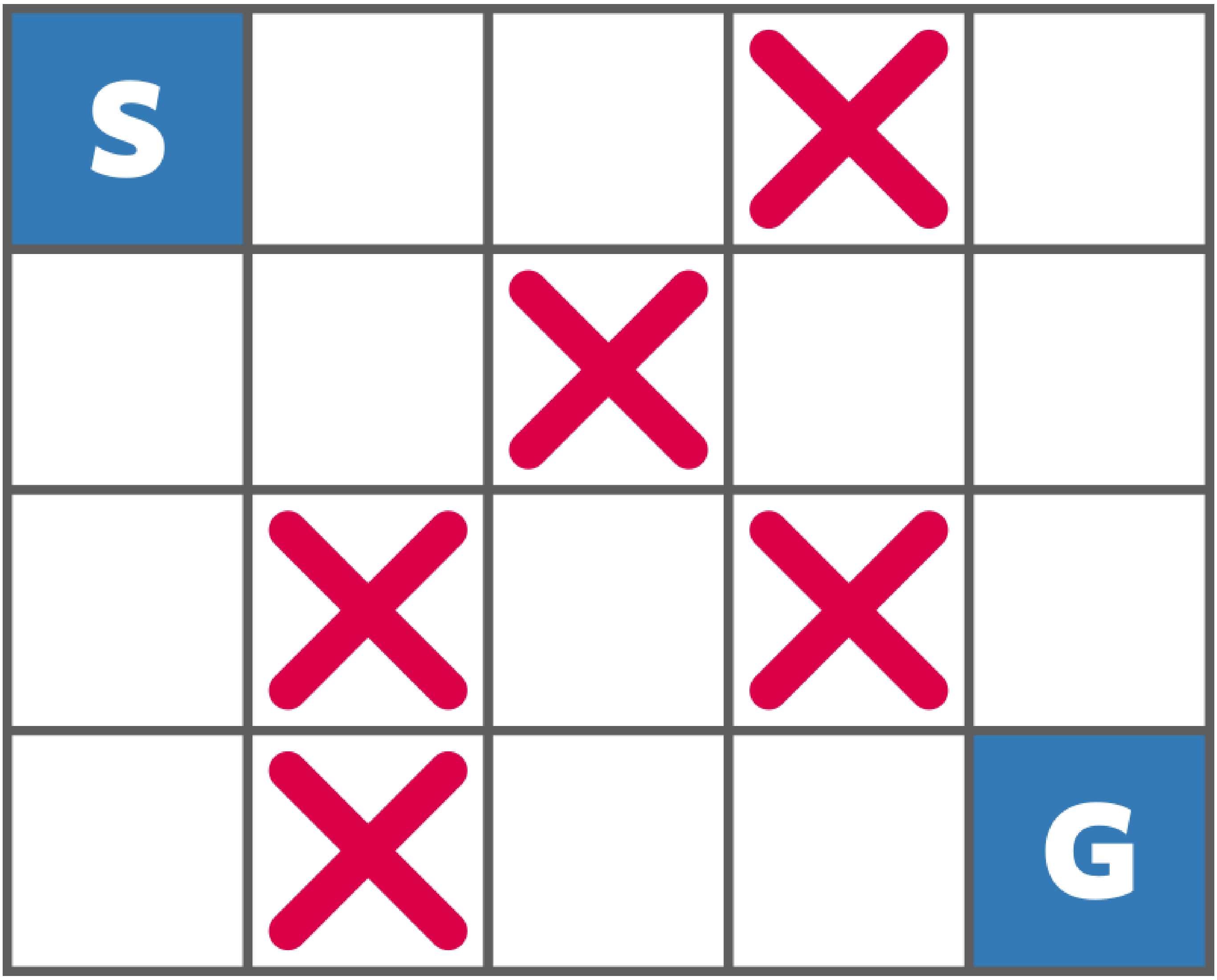
Output

If Takahashi can move from cell $(1, 1)$ to cell (H, W) by repeating the operation described in the problem, print `Yes`; otherwise, print `No`.

Sample 1

Input	Output
4 5 5 1 4 2 3 3 2 3 4 4 2	No

The grid looks as follows:

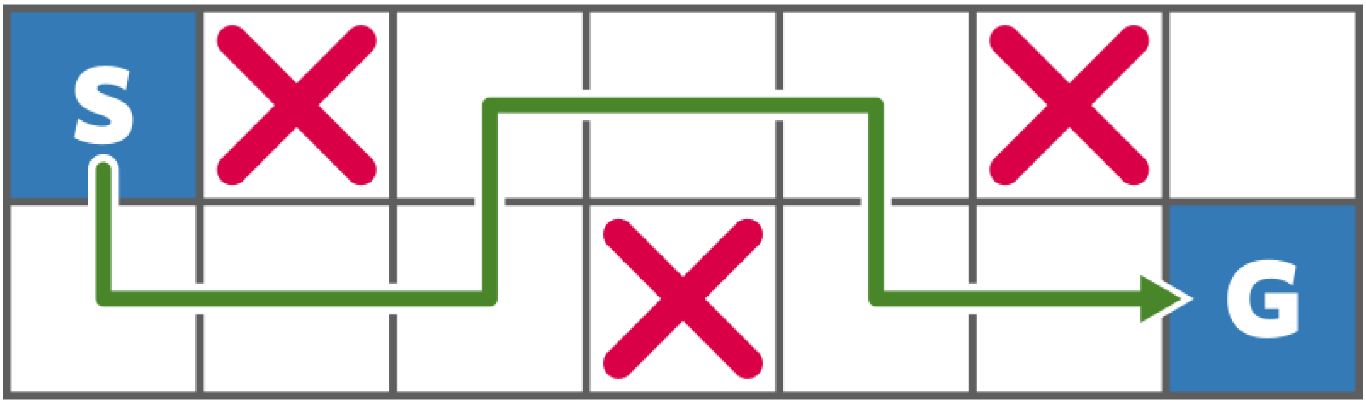


Takahashi cannot move from cell $(1, 1)$ to cell $(4, 5)$.

Sample 2

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

The grid looks as follows:



He can move from cell $(1, 1)$ to cell $(2, 7)$ by moving as shown in the figure.

Sample 3

Input	Output
`1 1 0`	`Yes`

Note that there may be cases where he does not need to move or where no obstacles are placed.

Sample 4

Input	Output

`10 12 20`
`8 3`
`1 11`
`6 4`
`3 7`
`10 4`
`5 7`
`4 7`
`5 5`
`4 3`
`6 1`
`1 6`
`2 7`
`6 7`
`1 3`
`6 3`
`2 12`
`9 6`
`7 3`
`3 11`
`9 7`
	`Yes`