A - Raise Both Hands

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 100 points

Problem Statement

Takahashi decided to make takoyaki (octopus balls) and serve it to Snuke. Takahashi instructed Snuke to raise only his left hand if he wants to eat takoyaki, and only his right hand otherwise.

You are given the information about which hand Snuke is raising as two integers L and R. He is raising his left hand if and only if $L = 1$, and raising his right hand if and only if $R = 1$. He might not follow the instructions and could raise both hands or not raise any hand at all.

If Snuke is raising only one hand, print `Yes` if he wants to eat takoyaki, and `No` if he does not. If he is raising both hands or not raising any hand, print `Invalid`.

Assume that if Snuke is raising only one hand, he is always following the instructions.

Constraints

- Each of L and R is 0 or 1.

Input

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

$$L \quad R$$

Output

Print `Yes`, `No`, or `Invalid` according to the instructions in the problem statement.

Sample Input 1

```
1 0
```

Sample Output 1

```
Yes
```

Snuke wants to eat takoyaki, so he is raising only his left hand.

Sample Input 2

```
1 1
```

Sample Output 2

```
Invalid
```

Snuke is raising both hands.

B - Binary Alchemy

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 200 points

Problem Statement

There are N types of elements numbered $1, 2, \ldots, N$.

Elements can be combined with each other. When elements i and j are combined, they transform into element $A_{i,j}$ if $i \geq j$, and into element $A_{j,i}$ if $i < j$.

Starting with element 1, combine it with elements $1, 2, \ldots, N$ in this order. Find the final element obtained.

Constraints

- $1 \leq N \leq 100$
- $1 \leq A_{i,j} \leq N$
- All input values are integers.

Input

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

$$
\begin{array}{l}
N \\
A_{1,1} \\
A_{2,1} \quad A_{2,2} \\
\vdots \\
A_{N,1} \quad A_{N,2} \quad \ldots \quad A_{N,N}
\end{array}
$$

Output

Print the number representing the final element obtained.

Sample Input 1

```
4
3
2 4
3 1 2
2 1 2 4
```

Sample Output 1

```
2
```

- Combining element 1 with element 1 results in element 3.
- Combining element 3 with element 2 results in element 1.
- Combining element 1 with element 3 results in element 3.
- Combining element 3 with element 4 results in element 2.

Therefore, the value to be printed is 2.

Sample Input 2

```
5
5
5 5
5 5 5
5 5 5 5
5 5 5 5 5
```

Sample Output 2

```
5
```

Sample Input 3

```
6
2
1 5
1 6 3
2 6 1 4
2 1 1 1 6
5 6 1 2 2 5
```

Sample Output 3

```
5
```

C - Word Ladder

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 300 points

Problem Statement

You are given two strings S and T consisting of lowercase English letters. Here, S and T have equal lengths.

Let X be an empty array, and repeat the following operation until S equals T:

- Change one character in S, and append S to the end of X.

Find the array of strings X with the minimum number of elements obtained in this way. If there are multiple such arrays with the minimum number of elements, find the lexicographically smallest one among them.

▶ What is lexicographical order on arrays of strings?

Constraints

- S and T are strings consisting of lowercase English letters with length between 1 and 100, inclusive.
- The lengths of S and T are equal.

Input

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

$$S$$
$$T$$

Output

Let M be the number of elements in the desired array. Print $M + 1$ lines.

The first line should contain the value of M.

The $i + 1$-th line $(1 \leq i \leq M)$ should contain the i-th element of the array.

Sample Input 1

```
adbe
bcbc
```

Sample Output 1

```
3
acbe
acbc
bcbc
```

Initially, $S =$ adbe.

We can obtain $X = ($ acbe , acbc , bcbc $)$ by performing the following operations:

- Change S to acbe and append acbe to the end of X.
- Change S to acbc and append acbc to the end of X.
- Change S to bcbc and append bcbc to the end of X.

Sample Input 2

```
abcde
abcde
```

Sample Output 2

```
0
```

Sample Input 3

```
afwgebrw
oarbrenq
```

Sample Output 3

```
8
aawgebrw
aargebrw
aarbebrw
aarbebnw
aarbebnq
aarbeenq
aarbrenq
oarbrenq
```

D - Cross Explosion

Time Limit: 4 sec / Memory Limit: 1024 MB

Score : 400 points

Problem Statement

There is a grid with H rows and W columns. Let (i, j) denote the cell at the i-th row from the top and j-th column from the left.
Initially, there is one wall in each cell.
After processing Q queries explained below in the order they are given, find the number of remaining walls.

In the q-th query, you are given two integers R_q and C_q.
You place a bomb at (R_q, C_q) to destroy walls. As a result, the following process occurs.

- If there is a wall at (R_q, C_q), destroy that wall and end the process.
- If there is no wall at (R_q, C_q), destroy the first walls that appear when looking up, down, left, and right from (R_q, C_q). More precisely, the following four processes occur simultaneously:
  - If there exists an $i < R_q$ such that a wall exists at (i, C_q) and no wall exists at (k, C_q) for all $i < k < R_q$, destroy the wall at (i, C_q).
  - If there exists an $i > R_q$ such that a wall exists at (i, C_q) and no wall exists at (k, C_q) for all $R_q < k < i$, destroy the wall at (i, C_q).
  - If there exists a $j < C_q$ such that a wall exists at (R_q, j) and no wall exists at (R_q, k) for all $j < k < C_q$, destroy the wall at (R_q, j).
  - If there exists a $j > C_q$ such that a wall exists at (R_q, j) and no wall exists at (R_q, k) for all $C_q < k < j$, destroy the wall at (R_q, j).

Constraints

- $1 \leq H, W$
- $H \times W \leq 4 \times 10^5$
- $1 \leq Q \leq 2 \times 10^5$
- $1 \leq R_q \leq H$
- $1 \leq C_q \leq W$
- All input values are integers.

Input

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

$$
\begin{array}{l}
H \quad W \quad Q \\
R_1 \quad C_1 \\
R_2 \quad C_2 \\
\vdots \\
R_Q \quad C_Q
\end{array}
$$

Output

Print the number of remaining walls after processing all queries.

Sample Input 1

```
2 4 3
1 2
1 2
1 3
```

Sample Output 1

```
2
```

The process of handling the queries can be explained as follows:

- In the 1st query, $(R_1, C_1) = (1, 2)$. There is a wall at $(1, 2)$, so the wall at $(1, 2)$ is destroyed.
- In the 2nd query, $(R_2, C_2) = (1, 2)$. There is no wall at $(1, 2)$, so the walls at $(2, 2), (1, 1), (1, 3)$, which are the first walls that appear when looking up, down, left, and right from $(1, 2)$, are destroyed.
- In the 3rd query, $(R_3, C_3) = (1, 3)$. There is no wall at $(1, 3)$, so the walls at $(2, 3), (1, 4)$, which are the first walls that appear when looking up, down, left, and right from $(1, 3)$, are destroyed.

After processing all queries, there are two remaining walls, at $(2, 1)$ and $(2, 4)$.

Sample Input 2

```
5 5 5
3 3
3 3
3 2
2 2
1 2
```

Sample Output 2

```
10
```

Sample Input 3

```
4 3 10
2 2
4 1
1 1
4 2
2 1
3 1
1 3
1 2
4 3
4 2
```

Sample Output 3

```
2
```

E - Avoid K Partition

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 475 points

Problem Statement

You are given a sequence $A = (A_1, A_2, \dots, A_N)$ of length N and an integer K.
There are 2^{N-1} ways to divide A into several contiguous subsequences. How many of these divisions have no subsequence whose elements sum to K? Find the count modulo 998244353.

Here, "to divide A into several contiguous subsequences" means the following procedure.

- Freely choose the number k $(1 \leq k \leq N)$ of subsequences and an integer sequence $(i_1, i_2, \dots, i_k, i_{k+1})$ satisfying $1 = i_1 < i_2 < \dots < i_k < i_{k+1} = N+1$.
- For each $1 \leq n \leq k$, the n-th subsequence is formed by taking the i_n-th through $(i_{n+1} - 1)$-th elements of A, maintaining their order.

Here are some examples of divisions for $A = (1, 2, 3, 4, 5)$:

- $(1, 2, 3), (4), (5)$
- $(1, 2), (3, 4, 5)$
- $(1, 2, 3, 4, 5)$

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $-10^{15} \leq K \leq 10^{15}$
- $-10^9 \leq A_i \leq 10^9$
- All input values are integers.

Input

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

$$
\begin{array}{l}
N \quad K \\
A_1 \quad A_2 \quad \dots \quad A_N
\end{array}
$$

Output

Print the count modulo 998244353.

Sample Input 1

```
3 3
1 2 3
```

Sample Output 1

```
2
```

There are two divisions that satisfy the condition in the problem statement:

- $(1), (2, 3)$
- $(1, 2, 3)$

Sample Input 2

```
5 0
0 0 0 0 0
```

Sample Output 2

```
0
```

Sample Input 3

```
10 5
-5 -1 -7 6 -6 -2 -5 10 2 -10
```

Sample Output 3

```
428
```

F - Cake Division

Time Limit: 5 sec / Memory Limit: 1024 MB

Score : 575 points

Problem Statement

There is a circular cake divided into N pieces by cut lines. Each cut line is a line segment connecting the center of the circle to a point on the arc.

The pieces and cut lines are numbered $1, 2, \ldots, N$ in clockwise order, and piece i has a mass of A_i. Piece 1 is also called piece $N + 1$.

Cut line i is between pieces i and $i + 1$, and they are arranged clockwise in this order: piece 1, cut line 1, piece 2, cut line 2, $\ldots$, piece N, cut line N.

We want to divide this cake among K people under the following conditions. Let w_i be the sum of the masses of the pieces received by the i-th person.

- Each person receives one or more **consecutive** pieces.
- There are no pieces that no one receives.
- Under the above two conditions, $\min(w_1, w_2, \ldots, w_K)$ is maximized.

Find the value of $\min(w_1, w_2, \ldots, w_K)$ in a division that satisfies the conditions, and the number of cut lines that are never cut in the divisions that satisfy the conditions. Here, cut line i is considered cut if pieces i and $i + 1$ are given to different people.

Constraints

- $2 \leq K \leq N \leq 2 \times 10^5$
- $1 \leq A_i \leq 10^4$
- All input values are integers.

Input

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

$$N \quad K$$
$$A_1 \quad A_2 \quad \ldots \quad A_N$$

Output

Let x be the value of $\min(w_1, w_2, \dots, w_K)$ in a division that satisfies the conditions, and y be the number of cut lines that are never cut. Print x and y in this order, separated by a space.

Sample Input 1

```
5 2
3 6 8 6 4
```

Sample Output 1

```
13 1
```

The following divisions satisfy the conditions:

- Give pieces $2, 3$ to one person and pieces $4, 5, 1$ to the other. Pieces $2, 3$ have a total mass of 14, and pieces $4, 5, 1$ have a total mass of 13.
- Give pieces $3, 4$ to one person and pieces $5, 1, 2$ to the other. Pieces $3, 4$ have a total mass of 14, and pieces $5, 1, 2$ have a total mass of 13.

The value of $\min(w_1, w_2)$ in divisions satisfying the conditions is 13, and there is one cut line that is not cut in either division: cut line 5.

Sample Input 2

```
6 3
4 7 11 3 9 2
```

Sample Output 2

```
11 1
```

Sample Input 3

```
10 3
2 9 8 1 7 9 1 3 5 8
```

Sample Output 3

```
17 4
```

G - Divisible by 3

Time Limit: 4 sec / Memory Limit: 1024 MB

Score : 650 points

Problem Statement

We call a positive integer n a good integer if and only if the sum of its positive divisors is divisible by 3.
You are given two positive integers N and M. Find the number, modulo 998244353, of length-M sequences A of positive integers such that the product of the elements in A is a good integer not exceeding N.

Constraints

- $1 \leq N \leq 10^{10}$
- $1 \leq M \leq 10^5$
- N and M are integers.

Input

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

$N \quad M$

Output

Print the answer.

Sample Input 1

```
10 1
```

Sample Output 1

```
5
```

There are five sequences that satisfy the conditions:

- (2)
- (5)
- (6)
- (8)
- (10)

Sample Input 2

```
4 2
```

Sample Output 2

```
2
```

There are two sequences that satisfy the conditions:

- $(1, 2)$
- $(2, 1)$

Sample Input 3

```
370 907
```

Sample Output 3

```
221764640
```

Sample Input 4

```
10000000000 100000
```

Sample Output 4

```
447456146
```