A - ARC Arc

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 400 points

Problem Statement

You are given a positive integer N and a sequence $A = (A_1, A_2, \ldots, A_N)$ of length N, consisting of 0 and 1.

We call a string S of length N, consisting only of uppercase English letters, a **good string** if it is possible to perform the following operation any number of times (possibly zero) so that the sequence A contains no 0. Here, $S_i\ (1 \leq i \leq N)$ denotes the i-th character of S, and we define $S_{N+1} = S_1, S_{N+2} = S_2$, and $A_{N+1} = A_1$.

- Perform one of the following operations:
  - Choose an integer i with $1 \leq i \leq N$ such that $S_i =$ `A`, $S_{i+1} =$ `R`, and $S_{i+2} =$ `C`, and replace each of A_i and A_{i+1} with 1.
  - Choose an integer i with $1 \leq i \leq N$ such that $S_{i+2} =$ `A`, $S_{i+1} =$ `R`, and $S_i =$ `C`, and replace each of A_i and A_{i+1} with 1.

Determine whether there exists a good string.

Constraints

- $3 \leq N \leq 200000$
- $A_i \in \{0, 1\}\ (1 \leq i \leq N)$
- All input values are integers.

Input

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

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

Output

If there exists a good string, print `Yes`; otherwise, print `No`.

The judge is case-insensitive; for example, if the correct answer is `Yes`, outputs such as `yes`, `YES`, or `yEs` will also be accepted.

Sample Input 1

```
12
0 1 0 1 1 1 1 0 1 1 1 0
```

Sample Output 1

```
Yes
```

For example, `RARCARCCRAGC` is a good string. This is because it is possible to change all elements of A to 1 by performing the following operations:

- Initially, $A = (0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0)$.
- Perform the first operation with $i = 2$. Then, $A = (0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0)$.
- Perform the first operation with $i = 5$. Then, $A = (0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0)$.
- Perform the second operation with $i = 8$. Then, $A = (0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0)$.
- Perform the second operation with $i = 12$. Then, $A = (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)$.

Since there exists a good string, output `Yes`.

Sample Input 2

```
3
0 0 0
```

Sample Output 2

```
No
```

Good strings do not exist.

Sample Input 3

```
29
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

Sample Output 3

```
Yes
```

Since A already contains no 0, every string of length 29 consisting of uppercase English letters is a good string.

B - Fennec VS. Snuke 2

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 600 points

Problem Statement

Fennec and Snuke are playing a board game.

You are given a positive integer N and a sequence $A = (A_1, A_2, \ldots, A_N)$ of positive integers of length N. Also, there is a set S, which is initially empty.

Fennec and Snuke take turns performing the following operation in order, starting with Fennec.

- Choose an index i such that $1 \leq A_i$. Subtract 1 from A_i, and if $i \notin S$, add i to S.
- If $S = \{1, 2, \ldots, N\}$, the game ends and the player who performed the last operation wins.

Note that it can be proven that until a winner is determined and the game ends, players can always make a move (there exists some i such that $1 \leq A_i$).

Both Fennec and Snuke play optimally to win. Determine who will win.

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq A_i \leq 10^9 \ (1 \leq i \leq N)$
- All input values are integers.

Input

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

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

Output

Print `Fennec` if Fennec wins, or `Snuke` if Snuke wins.

The judge is case-insensitive; for example, if the correct answer is `Fennec`, outputs such as `fennec`, `FENNEC`, or `fEnNeC` will also be accepted.

Sample Input 1

```
3
1 9 2
```

Sample Output 1

```
Fennec
```

For example, the game may proceed as follows:

- Initially, $A = (1, 9, 2)$ and S is empty.
- Fennec chooses index 2. Then, $A = (1, 8, 2)$ and $S = \{2\}$.
- Snuke chooses index 2. Then, $A = (1, 7, 2)$ and $S = \{2\}$.
- Fennec chooses index 1. Then, $A = (0, 7, 2)$ and $S = \{1, 2\}$.
- Snuke chooses index 2. Then, $A = (0, 6, 2)$ and $S = \{1, 2\}$.
- Fennec chooses index 3. Then, $A = (0, 6, 1)$ and $S = \{1, 2, 3\}$. The game ends with Fennec declared the winner.

This sequence of moves may not be optimal; however, it can be shown that even when both players play optimally, Fennec will win.

Sample Input 2

```
2
25 29
```

Sample Output 2

```
Snuke
```

Sample Input 3

```
6
1 9 2 25 2 9
```

Sample Output 3

```
Snuke
```

C - Range Sums 2

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 600 points

Problem Statement

This problem is interactive.

You are given a positive integer N.

Snuke secretly holds a permutation $P = (P_1, P_2, \ldots, P_N)$ of $(1, 2, \ldots, N)$ and a sequence $A = (A_1, A_2, \ldots, A_N)$ of positive integers of length N. **It is guaranteed that $P_1 < P_2$.**

You may send up to $2N$ queries to Snuke. A query is of the following form:

- Specify two **distinct** positive integers s and t with $1 \leq s, t \leq N$, and you will be given the value of $\sum_{i=\min(P_s,P_t)}^{\max(P_s,P_t)} A_i$.

Determine P and A.

Constraints

- $3 \leq N \leq 5000$
- $1 \leq P_i \leq N \, (1 \leq i \leq N)$
- $P_i \neq P_j$ for $i \neq j$.
- $P_1 < P_2$
- $1 \leq A_i \leq 10^9 \, (1 \leq i \leq N)$
- N, P_i, and A_i are integers.
- P and A are fixed before the interaction with the judge.

Input/Output

This problem is interactive.

First, N is given from Standard Input:

```
N
```

Next, you may send at most $2N$ queries to Snuke. When sending a query by specifying two **distinct** positive integers s, t, output in the following format. Do not forget to include a newline at the end.

```
? s t
```

After sending a query, you will receive a response from Snuke in the following format:

```
X
```

Here, X is an integer:

- If $X \neq -1$, then X is the value of $\displaystyle\sum_{i=\min(P_s,P_t)}^{\max(P_s,P_t)} A_i$.
- If $X = -1$, then either s, t do not satisfy the constraints or you have sent more than $2N$ queries.
  - In this case, your program is judged as incorrect and must terminate immediately.

Once you have determined P and A, output your answer in the following format. This output does not count as a query.

```
! P_1 P_2 ... P_N A_1 A_2 ... A_N
```

Note that $P_1 < P_2$. After this output, terminate your program immediately.

If you output anything that does not follow the above formats, `-1` will be given as input.

```
-1
```

In that case, your program is judged as incorrect and must terminate immediately.

Notes

- **After every output, be sure to end with a newline and flush the standard output. Failure to do so may result in a TLE.**
- When you output your answer, or receive `-1` from standard input, terminate your program immediately. Otherwise, the outcome is indeterminate.
- Note that extra newlines may be considered as an output format error.
- **The judge system for this problem is not adaptive.** That is, P and A are determined before the interaction with the judge and will not be changed at any point.

Sample Interaction

For $N = 6, P = (2, 4, 6, 5, 3, 1)$, and $A = (1, 9, 2, 25, 2, 9)$, here is an example of an interaction.

Input	Output	Explanation
`6`		First, the integer N is given from standard input.
	`? 1 2`	A query is sent to Snuke with $s = 1, t = 2$.
`36`		Since the query satisfies the constraints, the value 36, which is equal to $\sum_{i=\min(P_1,P_2)}^{\max(P_1,P_2)} A_i$, is returned.
	`? 2 5`	A query is sent to Snuke with $s = 2, t = 5$.
`27`		Since the query satisfies the constraints, the value 27, which is equal to $\sum_{i=\min(P_2,P_5)}^{\max(P_2,P_5)} A_i$, is returned.
	`! 2 4 6 5 3 1 1 9 2 25 2 9`	You report that P and A have been determined. After this output, the program should terminate immediately, and it will be judged as correct.

Note that this is just one example of an interaction. In particular, it is not guaranteed that P and A can be uniquely determined from the queries and responses shown above.

D - Fraction Line

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 600 points

Problem Statement

For a positive rational number x, define $f(x)$ as follows:

> Express x as $\frac{P}{Q}$ using coprime positive integers P and Q. $f(x)$ is defined as the value $P \times Q$.

You are given a positive integer N and a sequence $A = (A_1, A_2, \ldots, A_{N-1})$ of positive integers of length $N - 1$.

We call a sequence $S = (S_1, S_2, \ldots, S_N)$ of positive integers of length N a **good sequence** if it satisfies all of the following conditions:

- For every integer i with $1 \leq i \leq N-1$, it holds that $f\left(\frac{S_i}{S_{i+1}}\right) = A_i$.
- $\gcd(S_1, S_2, \ldots, S_N) = 1$.

Define the **score** of a sequence as the product of all its elements.

It can be proved that there are finitely many good sequences. Find the sum, modulo 998244353, of the scores of all good sequences.

Constraints

- $2 \leq N \leq 1000$
- $1 \leq A_i \leq 1000 \ (1 \leq i \leq N-1)$
- All input values are integers.

Input

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

$$
\begin{array}{l}
N \\
A_1 \quad A_2 \quad \ldots \quad A_{N-1}
\end{array}
$$

Output

Print the sum, modulo 998244353, of the scores of all good sequences.

Sample Input 1

```
6
1 9 2 2 9
```

Sample Output 1

```
939634344
```

For example, both $(2, 2, 18, 9, 18, 2)$ and $(18, 18, 2, 1, 2, 18)$ are good sequences, and both have a score of 23328.

There are a total of 16 good sequences, and the sum of the scores of all of them is 939634344.

Sample Input 2

```
2
9
```

Sample Output 2

```
18
```

There are 2 good sequences, both with a score of 9.

Sample Input 3

```
25
222 299 229 22 999 922 99 992 22 292 222 229 992 922 22 992 222 222 99 29 92 999 2 29
```

Sample Output 3

```
192457116
```

Do not forget to compute the sum modulo 998244353.

E - Snuke's Kyoto Trip

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 800 points

Problem Statement

You are given integers W, H, L, R, D, U.

A town of Kyoto is on the two-dimensional plane.

In the town, there is exactly one block at each lattice point (x, y) that satisfies all of the following conditions. There are no blocks at any other points.

- $0 \leq x \leq W$
- $0 \leq y \leq H$
- $x < L$ or $R < x$ or $y < D$ or $U < y$

Snuke traveled through the town as follows.

- First, he chooses one block and stands there.
- Then, he performs the following operation any number of times (possibly zero):
  - Move one unit in the positive direction of the x-axis or the positive direction of the y-axis. However, the point after moving must also have a block.

Print the number, modulo 998244353, of possible paths that Snuke could have taken.

Constraints

- $0 \leq L \leq R \leq W \leq 10^6$
- $0 \leq D \leq U \leq H \leq 10^6$
- There is at least one block.
- All input values are integers.

Input

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

$$W \quad H \quad L \quad R \quad D \quad U$$

Output

Print the answer.

Sample Input 1

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

Sample Output 1

```
192
```

The following are examples of possible paths. Here, a path is represented by listing the lattice points visited in order.

- $(3,0)$
- $(0,0) \to (1,0) \to (2,0) \to (2,1) \to (3,1) \to (3,2) \to (4,2) \to (4,3)$
- $(0,1) \to (0,2)$

There are 192 possible paths.

Sample Input 2

```
10 12 4 6 8 11
```

Sample Output 2

```
4519189
```

Sample Input 3

```
192 25 0 2 0 9
```

Sample Output 3

```
675935675
```

Do not forget to print the number of paths modulo 998244353.