A - Not Acceptable

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 100 points

Problem Statement

Takahashi had a report whose deadline was B minutes past A o'clock on May 17, 2025. He submitted it at D minutes past C o'clock on May 17, 2025. It is guaranteed that "B minutes past A o'clock" and "D minutes past C o'clock" are different times.

Output `Yes` if Takahashi submitted the report before the deadline, and `No` otherwise.

Constraints

- $0 \leq A, C \leq 23$
- $0 \leq B, D \leq 59$
- $(A, B) \neq (C, D)$
- $A, B, C,$ and D are integers.

Input

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

$$A \quad B \quad C \quad D$$

Output

If Takahashi submitted the report before the deadline, output `Yes`; otherwise, output `No`.

Sample Input 1

```
22 40 22 30
```

Sample Output 1

```
Yes
```

The deadline is 22:40, and he submitted at 22:30, so he submitted before the deadline. Hence, output `Yes`.

Sample Input 2

```
22 40 22 45
```

Sample Output 2

```
No
```

The deadline is 22:40, and he submitted at 22:45, so he submitted after the deadline. Hence, output `No`.

Sample Input 3

```
12 0 11 30
```

Sample Output 3

```
Yes
```

B - Product Calculator

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 200 points

Problem Statement

Takahashi has a calculator that initially displays 1.
He will perform N operations on the calculator.
In the i-th operation $(1 \leq i \leq N)$, he multiplies the currently displayed number by a positive integer A_i.
However, the calculator can display at most K digits. If the result of the multiplication has $(K+1)$ or more digits, the display shows 1 instead; otherwise, the result is shown correctly.

Find the number showing on the calculator after the N operations.

Constraints

- $1 \leq N \leq 100$
- $1 \leq K \leq 18$
- $1 \leq A_i < 10^K$
- All input values are integers.

Input

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

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

Output

Output the number shown on the calculator after the N operations.

Sample Input 1

```
5 2
7 13 3 2 5
```

Sample Output 1

```
10
```

This calculator can display at most two digits and initially shows 1. Takahashi operates as follows:

- The 1st operation multiplies by 7. Since $1 \times 7 = 7$, the calculator shows 7.
- The 2nd operation multiplies by 13. Since $7 \times 13 = 91$, the calculator shows 91.
- The 3rd operation multiplies by 3. Since $91 \times 3 = 273$, which has three digits, the calculator shows 1.
- The 4th operation multiplies by 2. Since $1 \times 2 = 2$, the calculator shows 2.
- The 5th operation multiplies by 5. Since $2 \times 5 = 10$, the calculator shows 10.

Sample Input 2

```
2 1
2 5
```

Sample Output 2

```
1
```

C - ~

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 350 points

Problem Statement

For an integer sequence $A = (A_1, A_2, \ldots, A_{|A|})$, we say that A is **tilde-shaped** if it satisfies all of the following four conditions:

- The length $|A|$ is at least 4.
- $A_1 < A_2$.
- There exists exactly one integer i with $2 \leq i < |A|$ such that $A_{i-1} < A_i > A_{i+1}$.
- There exists exactly one integer i with $2 \leq i < |A|$ such that $A_{i-1} > A_i < A_{i+1}$.

You are given a permutation $P = (P_1, P_2, \ldots, P_N)$ of $(1, 2, \ldots, N)$. Find the number of (contiguous) subarrays of P that are tilde-shaped.

Constraints

- $4 \leq N \leq 3 \times 10^5$
- $P = (P_1, P_2, \ldots, P_N)$ is a permutation of $(1, 2, \ldots, N)$.
- All input values are integers.

Input

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

$$N$$
$$P_1 \quad P_2 \quad \ldots \quad P_N$$

Output

Output the answer.

Sample Input 1

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

Sample Output 1

```
2
```

Among the subarrays of $(1, 3, 6, 4, 2, 5)$, exactly two are tilde-shaped: $(3, 6, 4, 2, 5)$ and $(1, 3, 6, 4, 2, 5)$.

Sample Input 2

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

Sample Output 2

```
0
```

Sample Input 3

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

Sample Output 3

```
4
```

D - Garbage Removal

Time Limit: 2.5 sec / Memory Limit: 1024 MiB

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 the j-th column from the left.

There are N pieces of trash on the grid; the i-th piece is at cell (X_i, Y_i).

You are given Q queries, which you must process in order. Each query is of one of the following types:

- Type 1: Given in the format `1 x` in the input. Report the number of trash pieces in the x-th row. Then, all trash pieces in the x-th row are removed from the grid.
- Type 2: Given in the format `2 y` in the input. Report the number of trash pieces in the y-th column. Then, all trash pieces in the y-th column are removed from the grid.

Constraints

- $1 \leq H, W, N \leq 2 \times 10^5$
- $1 \leq X_i \leq H$
- $1 \leq Y_i \leq W$
- If $i \neq j$, then $(X_i, Y_i) \neq (X_j, Y_j)$.
- $1 \leq Q \leq 2 \times 10^5$
- For a type 1 query, $1 \leq x \leq H$.
- For a type 2 query, $1 \leq y \leq W$.
- All input values are integers.

Input

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

$$
\begin{array}{l}
H \quad W \quad N \\
X_1 \quad Y_1 \\
X_2 \quad Y_2 \\
\vdots \\
X_N \quad Y_N \\
Q \\
\text{query}_1 \\
\text{query}_2 \\
\vdots \\
\text{query}_Q
\end{array}
$$

Here, query_i denotes the i-th query, which is given in one of the following formats:

$$1 \quad x$$

$$2 \quad y$$

Output

Output Q lines. The i-th line should contain the response to the i-th query.

Sample Input 1

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

Sample Output 1

```
2
1
0
1
0
```

Initially, trash pieces are at cells $(1,2)$, $(1,3)$, $(3,4)$, $(3,1)$, $(2,2)$.

In the 1st query, the 1st row contains two pieces of trash at $(1,2)$ and $(1,3)$, so report 2. These pieces are then removed; the remaining trash is at $(3,4)$, $(3,1)$, $(2,2)$.

In the 2nd query, the 2nd row contains one piece of trash at $(2,2)$, so report 1. This piece is then removed; the remaining trash is at $(3,4)$, $(3,1)$.

In the 3rd query, the 2nd column contains no trash, so report 0.

In the 4th query, the 4th column contains one piece of trash at $(3,4)$, so report 1. This piece is then removed; the remaining trash is at $(3,1)$.

In the 5th query, the 2nd row contains no trash, so report 0.

Sample Input 2

```
1 2 1
1 2
7
2 1
2 1
2 1
2 1
2 1
2 1
2 1
```

Sample Output 2

```
0
0
0
0
0
0
0
```

Sample Input 3

```
4 4 16
1 1
1 2
1 3
1 4
2 1
2 2
2 3
2 4
3 1
3 2
3 3
3 4
4 1
4 2
4 3
4 4
7
2 1
1 1
2 2
1 2
2 3
1 3
2 4
```

Sample Output 3

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

E - Popcount Sum 3

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 450 points

Problem Statement

You are given positive integers N and K.

Find the **sum**, modulo 998244353, of all positive integers x that do not exceed N and satisfy the following condition:

- the popcount of x is exactly K.

You are given T test cases; solve each of them.

▶ What is popcount?

Constraints

- $1 \leq T \leq 100$
- $1 \leq N < 2^{60}$
- $1 \leq K \leq 60$
- T, N, and K are integers.

Input

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

$$T$$
$$\mathrm{case}_1$$
$$\mathrm{case}_2$$
$$\vdots$$
$$\mathrm{case}_T$$

case_i denotes the i-th test case. Each test case is given in the following format:

$$N \quad K$$

Output

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

Sample Input 1

```
2
20 2
1234567890 17
```

Sample Output 1

```
100
382730918
```

For the first test case, there are nine positive integers not exceeding 20 whose popcount is 2: $3, 5, 6, 9, 10, 12, 17, 18, 20$. Their sum is 100.

The remainder when 100 is divided by 998244353 is 100, so output `100` on the first line.

Remember to output the sum modulo 998244353.

F - Compare Tree Weights

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 500 points

Problem Statement

There is a tree T with N vertices. The vertices are labeled as vertex 1, vertex $2, \ldots$, vertex N, and the edges are labeled as edge 1, edge $2, \ldots$, edge $(N-1)$.
Edge i $(1 \leq i \leq N-1)$ connects vertices U_i and V_i.
Each vertex has a weight; initially, the weight of every vertex is 1.

You are given Q queries to process in order. Each query is of one of the following two types:

- `1 x w` : Increase the weight of vertex x by w.
- `2 y` : If edge y were removed, the tree T would be split into two subtrees (connected components). For each subtree, let its weight be the sum of the weights of its vertices. Output the difference between the weights of the two subtrees.

For a query of the second type, it can be proved that removing any edge of T always splits it into exactly two subtrees.
Note that queries of the second type do not actually delete remove the edge.

Constraints

- $2 \leq N \leq 3 \times 10^5$
- $1 \leq U_i, V_i \leq N$
- $1 \leq Q \leq 3 \times 10^5$
- $1 \leq x \leq N$
- $1 \leq w \leq 1000$
- $1 \leq y \leq N-1$
- All input values are integers.
- The given graph is a tree.
- There is at least one query of the second type.

Input

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

$$
\begin{array}{l}
N \\
U_1 \quad V_1 \\
U_2 \quad V_2 \\
\vdots \\
U_{N-1} \quad V_{N-1} \\
Q \\
\mathrm{query}_1 \\
\mathrm{query}_2 \\
\vdots \\
\mathrm{query}_Q
\end{array}
$$

Each query query_i $(1 \leq i \leq Q)$ is given in one of the following formats:

```
1 x w
```

```
2 y
```

Output

Let K be the number of queries of the second type. Output K lines; the i-th line $(1 \leq i \leq K)$ should contain the answer to the i-th query of the second type.

Sample Input 1

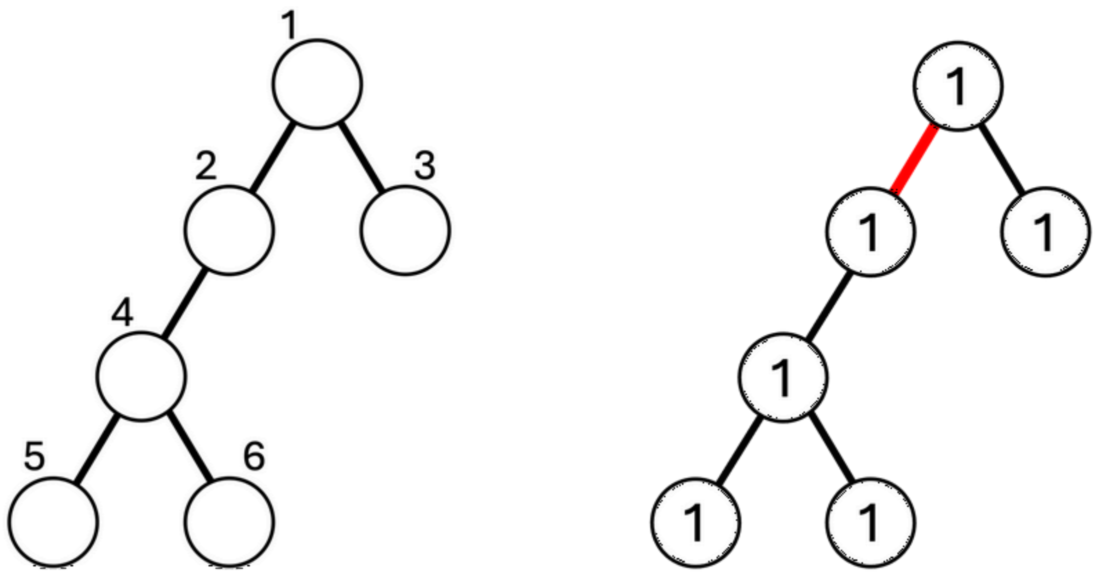
```
6
1 2
1 3
2 4
4 5
4 6
5
2 1
1 1 3
2 1
1 4 10
2 5
```

Sample Output 1

```
2
1
17
```

The structure of the tree T and the vertex numbering are as shown on the left of the figure below. Initially, the weight of every vertex is 1.

In the 1st query, consider deleting edge 1.
The tree splits into the subtree containing vertex 1 and the subtree containing vertex 2. The subtree with vertex 1 has weight 2; the subtree with vertex 2 has weight 4. Output the difference, 2 (figure below, right).

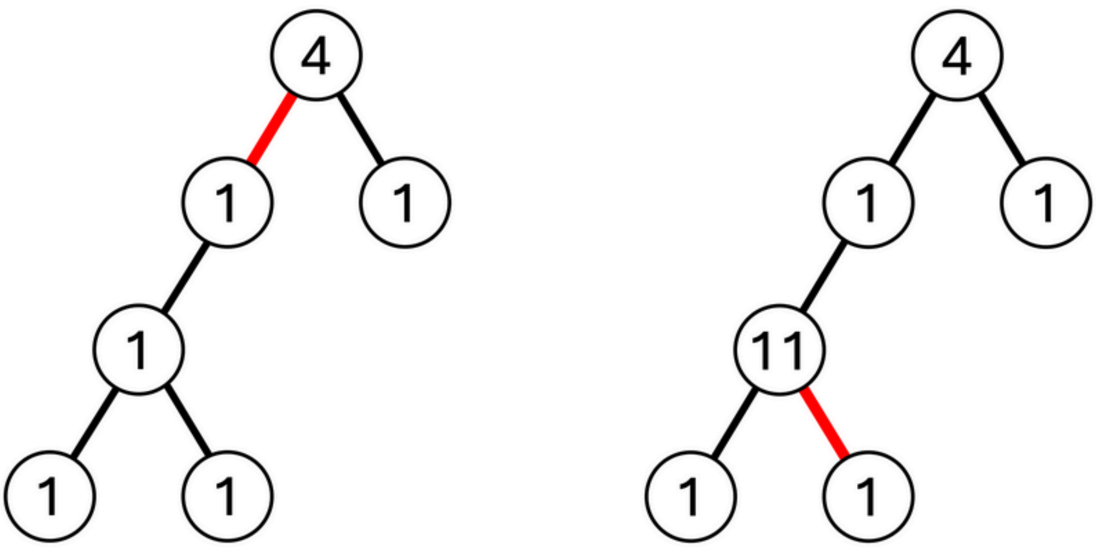


The 2nd query increases the weight of vertex 1 by 3.

In the 3rd query, consider deleting edge 1.
The subtree with vertex 1 has weight 5; the subtree with vertex 2 has weight 4. Output the difference, 1 (figure below, left).

The 4th query increases the weight of vertex 4 by 10.

In the 5th query, consider deleting edge 5.
The tree splits into the subtree containing vertex 4 and the subtree consisting only of vertex 6.
The subtree with vertex 4 has weight 18; the subtree with vertex 6 has weight 1. Output the difference, 17 (figure below, right).



Thus, output the answers to the queries of second type in order: 2, 1, and 17, separated by newlines.

G - Travelling Salesman Problem

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 625 points

Problem Statement

You and N merchants stand on a number line. The merchants are numbered $1, 2, \ldots, N$.

Initially, you are at coordinate 0, and merchant i is at coordinate X_i. Each merchant holds one item; the item held by merchant i is called item i.

Your goal is to receive items $1, 2, \ldots, N$ in this order.

You may repeat any number of times, in any order, the following three operations:

- Move yourself by 1. The cost of this operation is C.
- Choose one merchant and move that merchant by 1. The cost of this operation is D.
- Choose one merchant, say merchant i. If you and merchant i are at the same coordinate, and you have not yet received item i, then receive item i from merchant i. Otherwise, do nothing. The cost of this operation is 0.

Find the minimum total cost required to achieve the goal.

Also, output one possible combination of coordinates at which you receive each item when the total cost is minimized.

Constraints

- $1 \le N \le 2 \times 10^5$
- $1 \le C, D \le 10^5$
- $-10^5 \le X_i \le 10^5$
- All input values are integers.

Input

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

$$
\begin{array}{l}
N \quad C \quad D \\
X_1 \quad X_2 \quad \ldots \quad X_N
\end{array}
$$

Output

Output two lines.

The first line should contain the minimal total cost required to achieve the goal.

The second line should contain N integers $A_1, A_2, \ldots, A_N$ separated by spaces. Here, there must exist a sequence of operations that satisfies both of the following conditions:

- The goal is achieved, with the minimum possible total cost.
- For every integer i such that $1 \le i \le N$, you receive item i at coordinate A_i.

Sample Input 1

```
3 2 3
1 -1 2
```

Sample Output 1

```
10
0 0 2
```

For example, the following sequence of operations achieves the goal with total cost 10:

- Move merchant 1 from coordinate 1 to 0. The cost of this operation is 3.
- Move merchant 2 from coordinate -1 to 0. The cost of this operation is 3.
- Receive item 1 from merchant 1. The cost of this operation is 0.
- Receive item 2 from merchant 2. The cost of this operation is 0.
- Move yourself from coordinate 0 to 1. The cost of this operation is 2.
- Move yourself from coordinate 1 to 2. The cost of this operation is 2.
- Receive item 3 from merchant 3. The cost of this operation is 0.

It is impossible to achieve the goal with total cost less than 10.

Sample Input 2

```
2 100000 60000
100000 -100000
```

Sample Output 2

```
12000000000
0 0
```

Sample Input 3

```
6 4 4
2 -1 5 -2 -2 2
```

Sample Output 3

```
56
0 -1 -1 -2 -2 2
```