Problem A. Inversions of PQ and QP

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

You are given three integers N, A and B.

Consider two permutations $P = (P_1, P_2, \ldots, P_N)$ and $Q = (Q_1, Q_2, \ldots, Q_N)$ of $(1, 2, \ldots, N)$ with the following conditions:

- The inversion number of $(P_{Q_1}, P_{Q_2}, \ldots, P_{Q_N})$ equals A.
- The inversion number of $(Q_{P_1}, Q_{P_2}, \ldots, Q_{P_N})$ equals B.

Determine whether such permutations exist, and if they do, construct one example.

You have T test cases; solve each of them.

Definition of the inversion number

The inversion number of a sequence $R = (R_1, R_2, \ldots, R_M)$ of length M, is the number of pairs of integers (i, j) $(1 \leq i < j \leq M)$ such that $R_i > R_j$.

Input

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

T
case_1
$\vdots$
case_T

Each test case is given in the following format:

N A B

- $1 \leq T \leq 2 \times 10^5$
- $1 \leq N \leq 2 \times 10^5$
- $0 \leq A \leq \frac{N(N-1)}{2}$
- $0 \leq B \leq \frac{N(N-1)}{2}$
- The sum of N over all test cases does not exceed 2×10^5.
- All input values are integers.

Output

Print the answers for case_1, case_2, $\ldots$, case_T in the following format.

If there exist permutations $P = (P_1, P_2, \ldots, P_N)$ and $Q = (Q_1, Q_2, \ldots, Q_N)$ that satisfy the conditions, output any one of them in the following format:

Yes $P_1\ P_2\ \ldots\ P_N$ $Q_1\ Q_2\ \ldots\ Q_N$

If no such permutations exist, output `No`.

Example

standard input	standard output
3 3 3 1 3 1 2 8 13 11	Yes 1 3 2 2 3 1 No Yes 2 5 8 4 6 3 7 1 2 1 8 5 3 7 4 6

Note

For the output example of the first case, $(P_{Q_1}, P_{Q_2}, P_{Q_3}) = (3, 2, 1)$ and $(Q_{P_1}, Q_{P_2}, Q_{P_3}) = (2, 1, 3)$.

Problem B. Matching Query

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 megabytes

You are given an integer sequence $A = (A_1, A_2, \ldots, A_N)$ of length N, where each element is an integer between 0 and $M - 1$ (inclusive).

You need to process Q queries in order. The i-th query is described as follows:

- Given integers x_i and y_i, update the x_i-th element of A to y_i. After the update, solve the following problem:
  - Construct an undirected graph G with N vertices based on the sequence A. The vertices are numbered from 1 to N, and there is an edge between vertices u and v ($1 \le u < v \le N$, note the order of u and v) if and only if $A_u + 1 \equiv A_v \pmod{M}$. Find the size of a maximum matching of G.

Input

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

$$
\begin{array}{l}
N\ M\ Q \\
A_1\ A_2\ \ldots\ A_N \\
x_1\ y_1 \\
x_2\ y_2 \\
\vdots \\
x_Q\ y_Q
\end{array}
$$

- $2 \le N \le 3 \times 10^5$
- $1 \le Q \le 3 \times 10^5$
- $2 \le M \le 3 \times 10^5$
- $0 \le A_i < M$
- $1 \le x_i \le N$
- $0 \le y_i < M$
- All input values are integers.

Output

Print Q lines. The i-th line should contain the answer to the i-th query.

Example

standard input	standard output
6 3 5 1 1 0 2 0 2 6 0 4 1 5 2 1 2 6 2	1 1 2 3 3

Note

For the first query, A_6 is updated to 0, resulting in $A = (1, 1, 0, 2, 0, 0)$. The graph G has edges between the vertex pairs $(1, 4), (2, 4), (4, 5)$ and $(4, 6)$. The size of a maximum matching of G is 1.

For the second query, A_4 is updated to 1, resulting in $A = (1, 1, 0, 1, 0, 0)$. The graph G has an edge between the vertex pair $(3, 4)$. The size of a maximum matching of G is 1.

Problem C. 2-Power Rush

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	1024 megabytes

A multiset of positive integers is called a **good set** if all its elements are powers of 2.

For a non-negative integer N, define $f(N)$ as

$$f(N) = \sum_{T \in S_N} \prod_{i \in T} i$$

where S_N is the set of all good sets whose elements sum to N. Note that the sum of elements in an empty set is defined as 0, and the product of elements in an empty set is defined as 1. Thus, we define $f(0) = 1$.

You are given three non-negative integers T, a and b.

Define N_i as $N_i = (ai + b) \bmod 2^{30}$. Compute the following value

$$\sum_{i=0}^{T-1} (f(N_i) \bmod 998244353) \oplus i$$

where $\oplus$ represents the bitwise XOR operation.

Input

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

$T \ a \ b$

- $1 \le T \le 10^7$
- $0 \le a, b < 2^{30}$
- All input values are integers.

Output

Print the answer in a single line.

Examples

standard input	standard output
`5 1 0`	`17`
`3 1000000000 1000000000`	`1217611736`

Note

In the first example, $N_0 = 0, N_1 = 1, N_2 = 2, N_3 = 3$ and $N_4 = 4$.

- Good sets with a sum of 0 are $\{\}$. Thus, $f(0) = 1$.
- Good sets with a sum of 1 are $\{1\}$. Thus, $f(1) = 1$.
- Good sets with a sum of 2 are $\{1, 1\}$ and $\{2\}$. Thus, $f(2) = (1 \times 1) + (2) = 3$.

- Good sets with a sum of 3 are $\{1,1,1\}$ and $\{1,2\}$. Thus, $f(3) = (1 \times 1 \times 1) + (1 \times 2) = 3$.
- Good sets with a sum of 4 are $\{1,1,1,1\}, \{1,1,2\}, \{2,2\}$ and $\{4\}$.
  Thus, $f(4) = (1 \times 1 \times 1 \times 1) + (1 \times 1 \times 2) + (2 \times 2) + (4) = 11$.

Therefore, the answer is $(1 \oplus 0) + (1 \oplus 1) + (3 \oplus 2) + (3 \oplus 3) + (11 \oplus 4) = 17$.

In the second example, $N_0 = 1000000000, N_1 = 926258176$ and $N_2 = 852516352$.

Problem D. Swap Counter

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

For a permutation $(Q_1, Q_2, \ldots, Q_M)$ of $(1, 2, \ldots, M)$, we define a sequence $f(Q)$ of length $M-1$ as follows:

- Initialize a sequence X of length $M-1$ as $X = (0, 0, \ldots, 0)$.
- Perform the following operation $M-1$ times:
  - for $i = 1, 2, \ldots, M-1$, do the following:
    - if $Q_i > Q_{i+1}$, swap Q_i and Q_{i+1}, and add 1 to X_i.
    - if $Q_i < Q_{i+1}$, do nothing.
- The final sequence X is defined as $f(Q)$.

You are given a sequence $B = (B_1, B_2, \ldots, B_{N-1})$ of length $N-1$. Determine whether there exists a permutation P of $(1, 2, \ldots, N)$ such that $f(P) = B$. If such a permutation exists, find the lexicographically smallest one.

You have T test cases; solve each of them.

Input

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

```
T
case_1
case_2
⋮
case_T
```

Each test case is given in the following format:

```
N
B_1 B_2 ... B_{N-1}
```

- $1 \le T \le 1.5 \times 10^5$
- $2 \le N \le 3 \times 10^5$
- $0 \le B_i \le N-1$ $(i = 1, 2, \ldots, N-1)$
- The sum of N over all test cases does not exceed 3×10^5.
- All input values are integers.

Output

Output T lines. On the i-th line, print the answer for the i-th test case.

If there are no permutations satisfying the conditions, print `-1`. Otherwise, print the lexicographically smallest permutation satisfying the conditions.

Example

standard input	standard output
3 4 2 1 1 5 2 0 2 4 6 2 3 2 1 1	3 2 4 1 -1 3 5 4 2 6 1

Note

In the first test case, when $P = (3, 2, 4, 1)$, $f(P)$ is calculated as follows:

- At first, $X = (0, 0, 0)$.
- The first operation is performed as follows:
  - Since $P_1 > P_2$, swap P_1 and P_2, and add 1 to X_1.
    - As a result, $X = (1, 0, 0)$ and $P = (2, 3, 4, 1)$.
  - Since $P_2 < P_3$, do nothing.
  - Since $P_3 > P_4$, swap P_3 and P_4, and add 1 to X_3.
    - As a result, $X = (1, 0, 1)$ and $P = (2, 3, 1, 4)$.
- The second operation is performed as follows:
  - Since $P_1 < P_2$, do nothing.
  - Since $P_2 > P_3$, swap P_2 and P_3, and add 1 to X_2.
    - As a result, $X = (1, 1, 1)$ and $P = (2, 1, 3, 4)$.
  - Since $P_3 < P_4$, do nothing.
- The third operation is performed as follows:
  - Since $P_1 > P_2$, swap P_1 and P_2, and add 1 to X_1.
    - As a result, $X = (2, 1, 1)$ and $P = (1, 2, 3, 4)$.
  - Since $P_2 < P_3$, do nothing.
  - Since $P_3 < P_4$, do nothing.

Thus, $f(P) = (2, 1, 1)$. In particular, this P is the lexicographically smallest P that satisfies $f(P) = B$.

In the second test case, there are no permutations P such that $f(P) = (2, 0, 2, 4)$.

Problem E. 010-11 Shorten

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

You are given a string S of length N consisting of `0` and `1`.

You can perform the following two operations on S:

- **Operation 1**: choose a contiguous substring `010` and replace it with `1`.
- **Operation 2**: choose a contiguous substring `11` and replace it with `1`.

Find the maximum number of operations you can perform.

You have T test cases; solve each of them.

Input

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

T
case_1
$\vdots$
case_T

Each test case is given in the following format:

N
S

- $1 \leq T \leq 10^5$
- $1 \leq N \leq 10^6$
- S is a string of length N consisting of `0` and `1`.
- The sum of N over all test cases does not exceed 10^6.
- T and N are integers.

Output

Print T lines. On the i-th line, output the answer to the i-th test case.

Example

standard input	standard output
5 6 010100 4 0110 3 100 2 00 20 01001100000001101001	3 2 0 0 11

Note

In the first test case, you can perform the operations as follows:

- Replace the underlined `010` of `010100` with `1`. S becomes `0110`.
- Replace the underlined `11` of `0110` with `1`. S becomes `010`.
- Replace the underlined `010` of `010` with `1`. S becomes `1`.

You can not perform the operations on S more than three times, so the answer is 3.

Problem F. Another Long Sequence Inversion

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

You are given three non-negative integers L, R and X in binary. Find the inversion number of the integer sequence $(L \oplus X, (L+1) \oplus X, \ldots, R \oplus X)$ of length $R - L + 1$, and output it in binary.

Here, $\oplus$ is the bitwise XOR operation.

You have T test cases; solve each of them.

Definition of the inversion number

The inversion number of a sequence $B = (B_1, B_2, \ldots, B_M)$ of length M, is the number of pairs of integers $(i, j)(1 \le i < j \le M)$ such that $B_i > B_j$.

Input

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

T
case_1
case_2
$\vdots$
case_T

Each test case is given in the following format:

L R X

- $1 \le T \le 2 \times 10^5$
- $0 \le L \le R < 2^{2 \times 10^5}$
- $0 \le X < 2^{2 \times 10^5}$
- L, R and X are given in binary representation without leading zeros (except that 0 is considered a one-digit integer).
- The total number of digits in the binary representation of R over all test cases does not exceed 2×10^5.
- The total number of digits in the binary representation of X over all test cases does not exceed 2×10^5.
- All input values are integers.

Output

Output T lines. On the i-th line, print the answer for the i-th test case in binary.

Example

standard input	standard output
3 101 1000 10 1101 10111010 0 1000110 1110011011101 100011110010	10 0 11100100001101111010100

Note

In the first test case, L, R and X in decimal are $L = 5, R = 8$ and $X = 2$, respectively. Since $(5 \oplus 2, 6 \oplus 2, 7 \oplus 2, 8 \oplus 2) = (7, 4, 5, 10)$, the inversion number is 2.

Problem G. Convex Hull of Intersections

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

There are N lines on the xy-plane. The i-th line ℓ_i is represented by the equation $a_i x + b_i y + c_i = 0$. Let P be the set of all intersection points among these lines, defined as follows:

$$P = \{p \in \mathbb{R}^2 \mid \exists i, j \in \{1, 2, \ldots, N\} \text{ s.t. } p \in \ell_i, p \in \ell_j, i \neq j\}.$$

Find the area of the convex hull of P. If the convex hull is empty, a single point, or a line segment, the area is considered to be 0.

You have T test cases; solve each of them.

Definition of the convex hull

The convex hull $\mathrm{conv}(S)$ of a finite set $S = \{x_1, \ldots, x_{|S|}\}$ is defined as follows:

$$\mathrm{conv}(S) = \left\{ \sum_{i=1}^{|S|} \alpha_i x_i \,\middle|\, \sum_{i=1}^{|S|} \alpha_i = 1,\ 0 \leq \alpha_i \leq 1 \right\}.$$

Input

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

T
case_1
case_2
$\vdots$
case_T

Each test case is given in the following format:

N
$a_1\ b_1\ c_1$
$a_2\ b_2\ c_2$
$\vdots$
$a_N\ b_N\ c_N$

- $1 \leq T$
- $2 \leq N \leq 10^4$
- $|a_i|, |b_i|, |c_i| \leq 10^3$
- $a_i \neq 0$ or $b_i \neq 0$.
- Any two lines ℓ_i and ℓ_j are distinct ($i \neq j$).
- The total sum of N over all test cases does not exceed 2×10^5.
- All inputs values are integers.

Output

Output T lines. The i-th line should contain the answer for the i-th test case.

Your output will be considered correct when its absolute or relative error from actual answer is at most 10^{-5}.

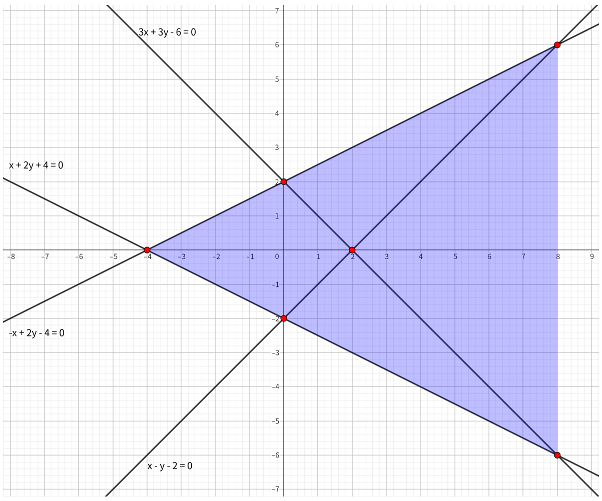
Example

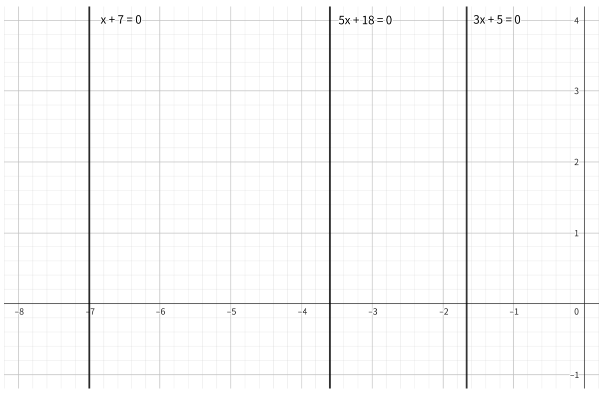
standard input	standard output
3 4 1 -1 -2 3 3 -6 -1 2 -4 1 2 4 3 3 0 5 5 0 18 1 0 7 3 314 159 -1 313 158 -1000 315 160 999	72.0 0 0.0016129032

Note

In the first example, the convex hull of P forms a triangle connecting the points $(8, 6), (-4, 0)$ and $(8, -6)$ in this order, with an area of 72.

In the second example, since all three lines are parallel, $P = \emptyset$, meaning the convex hull has an area of 0.





Problem H. 12 Grid

Input file: `standard input`
Output file: `standard output`
Time limit: 2 seconds
Memory limit: 1024 megabytes

There is an $N \times N$ grid. The cell in the i-th row and j-th column is denoted as cell (i, j).

You are given M tuples of integers (t_k, i_k, j_k, d_k) $(k = 1, 2, \ldots, M)$. Each tuple (t, i, j, d) satisfies the following:

- t is either 0 or 1.
- If $t = 0$, then $1 \leq i \leq N - 1$ and $1 \leq j \leq N$.
- If $t = 1$, then $1 \leq i \leq N$ and $1 \leq j \leq N - 1$.
- d is either 1 or 2.

You are asked to determine if there is a way to fill each cell of the grid with an integer such that the following conditions are satisfied, and if so, construct one valid configuration:

- The integer written in each cell is between 0 and 10^9 inclusive.
- The absolute difference between the integers written in two adjacent cells (up, down, left, or right) is either 1 or 2.
- For each $k = 1, 2, \ldots, M$, the following holds:
  - If $t_k = 0$, then the absolute difference between the integers written in the cells (i_k, j_k) and $(i_k + 1, j_k)$ is d_k.
  - If $t_k = 1$, then the absolute difference between the integers written in the cells (i_k, j_k) and $(i_k, j_k + 1)$ is d_k.

Input

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

N M
t_1 i_1 j_1 d_1
t_2 i_2 j_2 d_2
$\vdots$
t_M i_M j_M d_M

- $1 \leq N \leq 1000$
- $0 \leq M \leq \min\{2 \times 10^5, 2N(N-1)\}$
- Each (t_k, i_k, j_k, d_k) satisfies the conditions in the problem statement.
- If $k \neq \ell$, $(t_k, i_k, j_k) \neq (t_\ell, i_\ell, j_\ell)$.
- All input values are integers.

Output

If it is impossible to fill the grid with satisfying the conditions, print `No`.

If it is possible, output $N+1$ lines. On the first line, print `Yes`. On the $i+1$-th line $(i = 1, 2, \ldots, N)$, print the integers to be written on the cells $(i,1),(i,2),\ldots,(i,N)$ in this order, separated by spaces.

If there are multiple answers, print any.

Examples

standard input	standard output
2 3 0 1 1 2 1 1 1 1 0 1 2 2	Yes 0 1 2 3
2 3 0 1 1 2 1 1 1 2 1 2 1 1	Yes 0 2 2 3
2 4 0 1 1 2 1 1 1 2 1 2 1 1 0 1 2 2	No

Note

For the first example,

```
Yes
5 4
3 2
```

can also be a solution.

For the second example,

```
Yes
0 2
2 1
```

can also be a solution.

For the third example, there are no valid configurations.

Problem I. Small Steps

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

For a tree t with n vertices numbered from 1 to n, let $f(t)$ be the number of permutations $p = (p_1, \ldots, p_n)$ such that the following condition is satisfied:

- For each $i = 1, \ldots, n$, there are at most 2 edges on a simple path connecting vertices p_i and p_{i+1}, where $p_{n+1} = p_1$.

You are given positive integers N, K, and a tree T_0 with N vertices numbered from 1 to N. The i-th edge of T_0 connects vertices A_i and B_i.

A tree obtained by joining K disjoint copies of T_0 is called a **good tree**. More formally, we call a tree with NK vertices numbered from 1 to NK a **good tree** if and only if the following condition is satisfied:

- For all integers $1 \le i \le N-1$ and $0 \le k \le K-1$, there is an edge between vertices $(A_i + N \times k)$ and $(B_i + N \times k)$.

Find the sum of $f(T)$ over all good trees T, and output this value modulo 998244353.

You have Q test cases; solve each of them.

Input

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

Q
case_1
$\vdots$
case_Q

Each test case is given in the following format:

N K
A_1 B_1
$\vdots$
A_{N-1} B_{N-1}

- $1 \le Q \le 2 \times 10^5$
- $1 \le N \le 2 \times 10^5$
- $1 \le K \le 2 \times 10^5$
- $1 \le A_i < B_i \le N$
- The sum of N over all test cases does not exceed 2×10^5.
- T_0 is a tree.
- All input values are integers.

Output

Print Q lines. On the i-th line, output the answer to the i-th test case.

Example

standard input	standard output
5 4 1 1 2 1 3 1 4 4 1 1 2 2 3 3 4 1 4 4 2 1 2 1 3 1 4 6 200000 1 3 2 3 3 4 4 5 4 6	24 8 192 2304 210217795

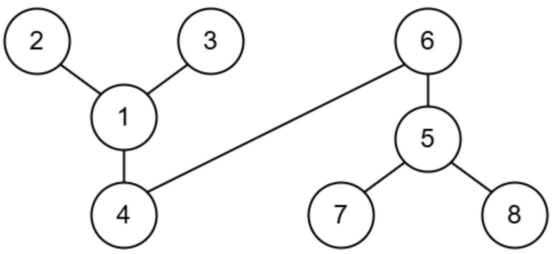
Note

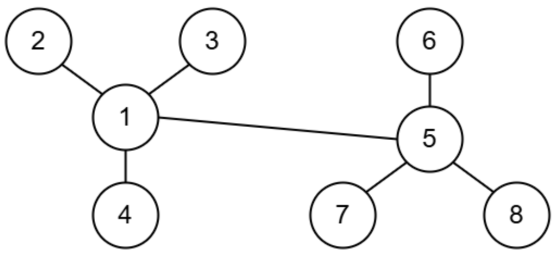
In the first test case, all possible permutations are counted since every simple path has at most 2 edges.

In the second test case, you need to count 8 permutations: $(1, 2, 4, 3)$, $(1, 3, 4, 2)$, $(2, 1, 3, 4)$, $(2, 4, 3, 1)$, $(3, 1, 2, 4)$, $(3, 4, 2, 1)$, $(4, 2, 1, 3)$ and $(4, 3, 1, 2)$.

In the third test case, you need to calculate the sum of $f(T)$ over all trees T with 4 vertices. Note that T_0 may have no edges.

In the fourth test case, the following trees are examples of good trees:





Problem J. Median Operations

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	1024 megabytes

You are given a positive **odd** integer N and a permutation $P = (P_1, P_2, \ldots, P_N)$ of $(1, 2, \ldots, N)$.

You have a sequence A, which is initially equal to P. You can repeatedly perform the following operation on sequence A:

- Choose a contiguous subsequence of odd length from A. Let m be the median of this subsequence. Remove the selected subsequence from A and insert m at its position.
  - More precisely, choose integers l and r such that $1 \le l \le r \le |A|$ ($|A|$ is the length of A) and $r - l + 1$ is odd. Replace A with $(A_1, \ldots, A_{l-1}, m, A_{r+1}, \ldots, A_{|A|})$, where m is the median of $(A_l, A_{l+1}, \ldots, A_r)$.

For each $k = 1, 2, \ldots, N$, determine if you can transform A into a sequence (k) of length 1 through these operations.

You have T test cases; solve each of them.

Input

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

```
T
case_1
case_2
⋮
case_T
```

Each test case is given in the following format:

```
N
P_1 P_2 ... P_N
```

- $1 \le T \le 10^4$
- $3 \le N \le 2 \times 10^5$
- N is odd.
- $(P_1, P_2, \ldots, P_N)$ is a permutation of $(1, 2, \ldots, N)$.
- The total sum of N over all test cases does not exceed 2×10^5.
- All input values are integers.

Output

Output T lines.

The i-th line should contain a string of length N representing the answer for the i-th test case. The k-th character of this string should be `1` if it is possible to transform the sequence into (k) through the given operations, and `0` otherwise.

Example

standard input	standard output
2 5 2 3 1 5 4 7 7 6 3 4 5 2 1	00110 0101010

Note

In the first test case,

- For $k = 3$, by choosing $(l, r) = (1, 5)$, we can transform A into (3).
- For $k = 4$, by first choosing $(l, r) = (1, 3)$ to transform A into $(2, 5, 4)$, then choosing $(l, r) = (1, 3)$ again, we obtain $A = (4)$.
- For $k = 1, 2, 5$, no such sequence of operations exists.

Problem K. Shuffle and Max Bracket Score

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 megabytes

Aoba-san came up with the following problem.

> **Max Bracket Score**
>
> You are given an integer sequence $A = (A_1, A_2, \ldots, A_{2N})$ of length $2N$.
> For a correct parentheses sequence s of length $2N$, its score is defined as follows:
>
> - The sum of A_i for all indices i where s_i is `(`.
>
> Find the maximum possible score among all correct parentheses sequences of length $2N$.

Hirose-san thought it was too easy, and came up with the following problem.

> **Shuffle and Max Bracket Score**
>
> You are given an integer sequence $A = (A_1, A_2, \ldots, A_{2N})$ of length $2N$.
> After shuffling A uniformly at random, compute the expected value of the answer to Max Bracket Score problem modulo 998244353.

Solve **Shuffle and Max Bracket Score**.

Definition of a correct parentheses sequence

A string is said to be a correct parentheses sequence if one of the following conditions is satisfied.

- It is an empty string.
- It is a concatenation of `(`, s, `)`, for some correct parentheses sequence s.
- It is a concatenation of s and t, for some correct parentheses sequences s and t.

Definition of the expected value modulo 998244353

It can be proven that the sought expected value is allways a rational number. Also, in the constraints of this problem, it is guaranteed that when the sought expected value is expressed in the form of an irreducible fraction $\frac{y}{x}$, x is not divisible by 998244353. In this case, there exists a unique integer $0 \le z < 998244353$ satisfying $y \equiv xz \pmod{998244353}$, so output z.

Input

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

> N
> A_1 A_2 $\ldots$ A_{2N}

- $1 \le N \le 10^5$
- $1 \le A_i \le 10^9$
- All input values are integers.

Output

Print the answer in a single line.

Examples

standard input	standard output
1 1 2	499122178
2 1 2 3 4	831870300
4 31415 92653 58979 32384 62643 38327 95028 84197	420993474

Note

For the first example,

- When $A = (1, 2)$, the answer to the Max Bracket Score problem is 1.
  - When s is `()`, the score is 1.
- When $A = (2, 1)$, the answer to the Max Bracket Score problem is 2.
  - When s is `()`, the score is 2.

The expected value is $\frac{1}{2}(1 + 2) = \frac{3}{2}$.

For the second example,

- For example, when $A = (1, 2, 3, 4)$, the answer to the Max Bracket Score problem is 4.
  - When s is `()()`, the score is $1 + 3 = 4$.
  - When s is `(())`, the score is $1 + 2 = 3$.
- Similarly, when $A = (4, 3, 2, 1)$, the answer to the Max Bracket Score problem is 7.
  - When s is `()()`, the score is $4 + 2 = 6$.
  - When s is `(())`, the score is $4 + 3 = 7$.

The expected value is $\frac{35}{6}$.

Problem L. Square Connection

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

You are given positive integers, s and t. You can perform the following operation any number of times (including zero):

- Choose an integer u such that $1 \leq u \leq 4 \times 10^{18}$ and $s + u$ is a square number, then replace s with u.

Find the minimum number of operations required to make s equal to t, and provide one sequence of operations.

Formally, find a sequence of integers $(u_1, u_2, \ldots, u_K)$ satisfying the following conditions:

- K is the minimum number of operations required to make s equal to t.
- For each $i = 1, 2, \ldots, K$, the condition $1 \leq u_i \leq 4 \times 10^{18}$ holds.
- Let $u_0 = s$, for each $i = 1, 2, \ldots, K$, the sum $u_{i-1} + u_i$ is a square number.
- $u_K = t$.

It is guaranteed that under the constraints of this problem, there always exists a way to transform s into t using at most 10^6 operations.

You have T test cases; solve each of them

Input

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

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

Each test case is given in the following format:

$$s \ t$$

- $1 \leq T \leq 3 \times 10^5$
- $1 \leq s, t \leq 10^9$
- $s \neq t$
- The sum of the numbers of operations ($= K$) over all test cases does not exceed 10^6.
- All input values are integers.

Output

Output T lines. On the i-th line, print the answer for the i-th test case in the following format:

$$K \ u_1 \ u_2 \ \ldots \ u_K$$

If there are multiple answers, print any.

Example

standard input	standard output
3 8 3 20 24 998236771 998244353	2 1 3 3 5 76 24 1 998244353

Note

For the first test case, $8 + 1 = 9$ and $1 + 3 = 4$ are both square numbers. Additionally, it is not possible to replace 8 into 3 in a single operation.

Problem M. Divide Digit String

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

You are given a digit string S of length N consisting of digits from 1 to 9, and integers M and K $(1 \le K \le M \le N)$.

You need to split S into M non-empty contiguous substrings and interpret each substring as a decimal integer. Find the smallest possible value that can be the K-th largest among these M integers.

You have T test cases; solve each of them.

Input

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

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

Each test case is given in the following format:

$$\begin{array}{l} N \ M \ K \\ S \end{array}$$

- $1 \le T \le 10^5$
- $1 \le K \le M \le N \le 10^6$
- T, N, M and K are integers.
- S is a digit string of length N consisting of digits from 1 to 9.
- The sum of N over all test cases does not exceed 10^6.

Output

Print T lines. On the i-th line, print the answer for the i-th test case.

Example

standard input	standard output
3 5 2 1 12345 5 3 3 12345 10 7 1 3141592653	123 1 26

Note

In the first test case, by splitting S into `123` and `45`, the largest integer is 123, which is the smallest possible value for the largest integer.

In the second test case, by splitting S into `1`, `23` and `45`, the third largest integer is 1, which is the smallest possible value for the third largest integer.

Problem N. Palindromic Path

Input file: `standard input`
Output file: `standard output`
Time limit: 2 seconds
Memory limit: 1024 megabytes

You are given a simple, undirected graph G with $2N$ vertices and M edges. Each vertex i $(i = 1, 2, \ldots, 2N)$ has an integer label assigned to it, given by $\lfloor (i+1)/2 \rfloor$.

For each j $(j = 1, 2, \ldots, M)$, there exists a edge connecting vertexs u_j and v_j.

A sequence of vertices $P = (v_1, v_2, \ldots, v_K)$ in G is called a **palindromic path** if it satisfies the following three conditions:

1. $K \geq 2$.
2. P is a **simple path**, i.e.,
   - For each $k = 1, 2, \ldots, K-1$, there exists an edge between vertices v_k and v_{k+1}.
   - The sequence does not revisit any vertex, i.e., for all $1 \leq k < \ell \leq K$, $v_k \neq v_\ell$.
3. The sequence of integers labeled on the vertices of P forms a **palindrome**, meaning:
   - For each $k = 1, 2, \ldots, \lfloor K/2 \rfloor$, the condition $\lfloor (v_k + 1)/2 \rfloor = \lfloor (v_{K-k+1} + 1)/2 \rfloor$ holds.

For each integer $x = 1, 2, \ldots, N$, determine whether there exists a palindromic path in G that starts from a vertex labeled with x.

Input

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

$$
\begin{array}{l}
N\ M \\
u_1\ v_1 \\
u_2\ v_2 \\
\vdots \\
u_M\ v_M
\end{array}
$$

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq M \leq 4 \times 10^5$
- $1 \leq u_j < v_j \leq 2N$
- $(u_i, v_i) \neq (u_j, v_j)$ $(i \neq j)$
- All input values are integers.

Output

Output N lines. On the x-th line, if there exists a palindromic path starting from a vertex labeled with x, print `Yes`. Otherwise, print `No`.

Examples

standard input	standard output
4 9 1 3 2 5 2 7 3 5 4 6 4 8 5 6 6 7 6 8	No Yes Yes Yes
3 6 1 3 3 5 2 4 4 6 1 5 1 6	No Yes Yes

Note

In the first example, there are the examples of palindromic paths starting from $x = 2, 3$ and 4,

- For $x = 2$, $(3, 5, 6, 4)$.
- For $x = 3$, $(5, 6)$.
- For $x = 4$, $(7, 6, 8)$.

Problem O. Twin Contests

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

You are gevin a positive integer N. For $n = 1, 2, \ldots, N$, solve the following problem:

> Find the number, modulo 998244353, of permutations $P = (P_1, P_2, \ldots, P_N)$ obtained by rearranging $(1, 2, \ldots, N)$ such that for all $m = 1, 2, \ldots, N$,
>
> - $n \neq m \implies nP_n < mP_m$.

Input

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

> N

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

Output

Print N lines. On the i-th line, print the answer when $n = i$.

Example

standard input	standard output
3	3 1 0

Note

- When $n = 1$, the permutations satisfying the condition are $P = (1, 2, 3), (1, 3, 2)$ and $(2, 3, 1)$, for a total of 3.
- When $n = 2$, the permutation satisfying the condition is $P = (3, 1, 2)$, for a total of 1.
- When $n = 3$, there are no permutations satisfying the condition.