The 2nd Universal Cup

Stage 20: Ōokayama

January 26-27, 2024

This problem set should contain 16 problems on 34 numbered pages.

Based on

Tokyo Tech Programming Contest

Problem A. Numerous Elimination

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

A tournament is held with N players numbered $1, 2, \ldots, N$.

There are N queues labeled $0, 1, \ldots, N-1$ at the venue, and a player standing in queue i $(0 \le i \le N-1)$ indicates that they have won i consecutive matches at that time.

At the start of the tournament, players are lined up in queue 0 in order of players $1, 2, \ldots, N$.

The tournament determines the rank of each player according to the following procedure:

1. When there is exactly one player standing in each queue, the rank of the player in queue i is $N-i$. In this case, the procedure ends.
2. Among the queues with two or more players, select the queue with the smallest number as queue l.
3. The top two players in queue l leave the queue and play a match. The winner of the match joins the back of queue $l+1$, and the loser joins the back of queue 0.
4. Return to step 1.

Find the number of matches played in this tournament, modulo 998244353.

Assume that there are no ties in the matches, and it can be proven that the answer is unique regardless of the match results.

Input

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

N

- N is an integer.
- $1 \le N \le 10^5$

Output

Print the answer.

Examples

standard input	standard output
3	4
5	26
100000	538161387

Note

In the first example, assuming that the player with the smaller number wins the match, the tournament progresses as follows:

Queue 0	Queue 1	Queue 2	Explanation
$\underline{1,2},3$			Players 1 and 2 have a match. Player 1 joins Queue 1, and player 2 joins Queue 0.
$\underline{3,2}$	1		Players 3 and 2 have a match. Player 2 joins Queue 1, and player 3 joins Queue 0.
3	$\underline{1,2}$		Players 1 and 2 have a match. Player 1 joins Queue 2, and player 2 joins Queue 0.
$\underline{3,2}$		1	Players 3 and 2 have a match. Player 2 joins Queue 1, and player 3 joins Queue 0.
3	2	1	Since there is exactly one player in each queue, the tournament ends.

The tournament consists of 4 matches, so the output is 4.

Problem B. Almost Large

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

You are given a set of non-negative integers of size N, denoted as $S = \{S_1, S_2, \ldots, S_N\}$.

There is a variable x, initially set to S_1. You can perform the following operation any number of times:

- Choose one element from S and denote it as y. Replace x with y if the following **condition** is satisfied:
  - **Condition**: Let X_j and Y_j be the digits at the 3^j place in the ternary representations of x and y, respectively. The number of indices j such that $X_j > Y_j$ must be at most 1.

Determine whether it is possible to make $x = S_N$ after performing some operations.

Input

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

```
N
S_1 S_2 ··· S_N
```

- All values in the input are integers.
- $2 \leq N \leq 2 \times 10^5$
- $0 \leq S_i < 3^{12}$ $(1 \leq i \leq N)$
- $S_i \neq S_j$ $(1 \leq i < j \leq N)$

Output

Output `Yes` if it is possible to make $x = S_N$, otherwise output `No`.

Examples

standard input	standard output
2 21 14	Yes
2 12 1	No
5 5 15 45 135 405	Yes

Note

In the first example, you can transform $x = 21$ to $x = 14$ as follows:

- Initially, $x = 21$. Choose $y = 14$ and perform the operation.
  - In ternary representation, $(X_2, X_1, X_0) = (2, 1, 0)$ for x, and $(Y_2, Y_1, Y_0) = (1, 1, 2)$ for y.
  - There is only one index $j = 2$ where $X_j > Y_j$, so replace x with 14.

Problem C. Yet Another Simple Math Problem

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

You are given an integer N. Find the number of pairs of positive integers (a, b) that satisfy both of the following conditions:

- $1 \le a, b \le N$
- There exist positive integers (x, y) such that $x + y^2 = a$ and $x^2 + y = b$

Given T test cases, solve each of them.

Input

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

T
case_1
case_2
$\vdots$
case_T

Each test case case_i $(1 \le i \le T)$ is given in the following format:

N

- All values in the input are integers.
- $1 \le T \le 10^5$
- $1 \le N \le 10^{18}$

Output

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

Example

standard input	standard output
3 6 1 101	4 0 83

Note

In the first test case, there are four pairs (a, b) that satisfy the conditions: $(a, b) = (2, 2), (3, 5), (5, 3), (6, 6)$. For example, for $(a, b) = (3, 5)$, choosing $(x, y) = (2, 1)$ satisfies $x + y^2 = 3 = a$ and $x^2 + y = 5 = b$, fulfilling the given conditions.

Problem D. Spacecraft

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

In a 3-dimensional space, there are N distinct stars shining at different coordinates. The i-th star is located at point $P_i(x_i, y_i, z_i)$. Additionally, a spherical spacecraft with radius R is floating with the origin as its center.

A point p in space is considered a *lovely point* if the following conditions simultaneously hold for $i = 1, 2, \ldots, N$:

- Star i is observable from point p. In other words, the line segment with endpoints p and P_i does not pass through or touch the spacecraft's surface.

Find the number of connected components in the region where lovely points exist. Specifically, for the set L of all lovely points and the relation $\sim$ defined below, determine the size of the quotient set $L/\sim$.

- For $p_1, p_2 \in L$, $p_1 \sim p_2$ if and only if there exists a curve on L with endpoints p_1 and p_2.

Note that this value can be proven to be an non-negative integer.

Given T test cases, solve each of them.

Input

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

T
case_1
case_2
$\vdots$
case_T

Each case_i $(1 \le i \le T)$ is given in the following format:

N R
x_1 y_1 z_1
$\vdots$
x_N y_N z_N

- All values in the input are integers.
- $1 \le T \le 10$
- $1 \le N \le 500$
- $1 \le R < \sqrt{x_i^2 + y_i^2 + z_i^2} \le 10^3$ $(1 \le i \le N)$
- $(x_i, y_i, z_i) \ne (x_j, y_j, z_j)$ $(1 \le i < j \le N)$
- The answer remains unchanged by the following operations:

– For $i = 1, 2, \ldots, N$, independently choose a line l_i passing through the origin and a real number θ_i ($|\theta_i| \leq 10^{-6}$). Move the position of star i to the position obtained by rotating it around the axis l_i by an angle of θ_i.

Output

Output the answer.

Example

standard input	standard output
3 4 12 13 0 0 0 15 0 0 -15 0 0 0 15 6 100 0 0 101 0 0 -101 0 101 0 0 -101 0 101 0 0 -101 0 0 20 333 328 -160 -572 -165 417 -847 -319 -45 271 359 -467 -625 -355 -451 658 -280 -424 687 -65 -224 573 475 -371 373 -246 -54 -903 595 -196 -305 622 -570 -250 386 -541 -566 647 455 -424 734 117 -405 830 -10 -393 -334 137 154 74 459 -92 -651 -93 -131 879 148 45 -48 126 -660	1 0 3

Note

In the first test case, there exist lovely points.

- For example, $(0, 0, 100)$ is a lovely point. Any line segment connecting this point with any of the given 4 points does not pass through or touch the spacecraft.
- Additionally, $(21, 0, 0)$ is also a lovely point. These two points belong to the same connected component.

In the second test case, there are no lovely points.

Problem E. R-Connected Components

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

For a positive integer R, define the number of connected components in the following infinite undirected graph as $f(R)$.

- The set of vertices is $\mathbb{Z}^2$. In other words, for any pair of integers x, y, there exists a vertex (x, y).
- There exists an edge between vertices (x_1, y_1) and (x_2, y_2) if and only if $|x_1 - x_2|^2 + |y_1 - y_2|^2 = R$.

Given a positive integer R, output $f(R)$. If $f(R)$ is infinite, output `inf`.

Given T test cases, solve each of them.

Input

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

T
case_1
case_2
$\vdots$
case_T

Each case_i $(1 \le i \le T)$ is given in the following format:

R

- All values in the input are integers.
- $1 \le T \le 100$
- $1 \le R \le 10^9$

Output

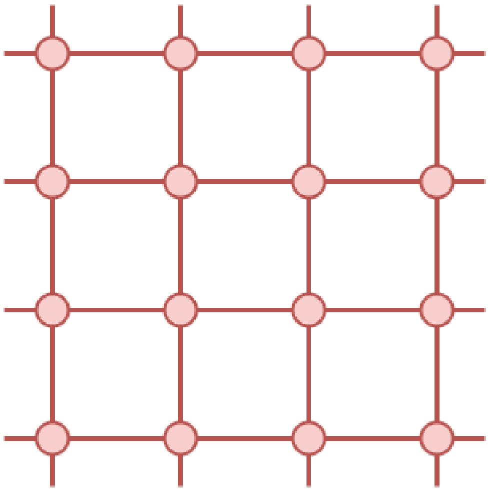
For each test case, output $f(R)$ if it is finite, otherwise output `inf`.

Example

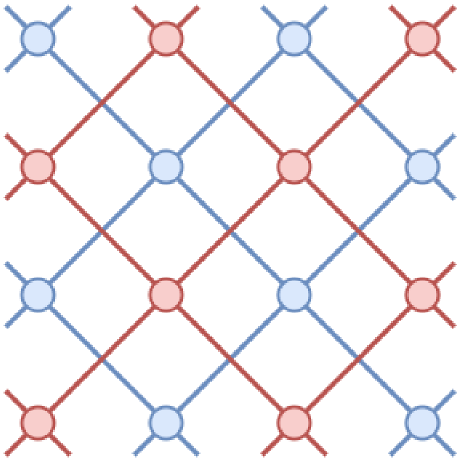
standard input	standard output
3 1 2 3	1 2 inf

Note

In the first test case, $R = 1$. The edges are formed as shown below, resulting in a single connected component.



In the second test case, $R = 2$. The edges are formed as shown below, resulting in two connected components.



In the third test case, $R = 3$. There are no edges in this graph, and the number of connected components is infinite.

Problem F. N^a (log N)^b

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

You are given a function $F(N)$ for a positive integer N, represented as a string F following the BNF notation for the ⟨expr⟩ symbol as follows:

⟨expr⟩ ::= ⟨term⟩ | ⟨expr⟩ '`+`' ⟨term⟩
⟨term⟩ ::= ⟨factor⟩ | ⟨term⟩ '`*`' ⟨factor⟩
⟨factor⟩ ::= '`N`' | '`N^`' ⟨number⟩ | '`log(`' ⟨expr⟩ '`)`' | '`log(`' ⟨expr⟩ '`)^`' ⟨number⟩ | '`(`' ⟨expr⟩ '`)`'
⟨number⟩ ::= ⟨nonzero_digit⟩ | ⟨nonzero_digit⟩⟨digit_string⟩
⟨digit_string⟩ ::= ⟨digit⟩ | ⟨digit⟩⟨digit_string⟩
⟨nonzero_digit⟩ ::= '`1`' | '`2`' | '`3`' | '`4`' | '`5`' | '`6`' | '`7`' | '`8`' | '`9`'
⟨digit⟩ ::= '`0`' | ⟨nonzero_digit⟩

The symbols represent the following:

- `N`: N
- `+`: Addition $+$
- `*`: Multiplication $\times$
- `log`: Natural logarithm $\log$
- `(`, `)`: Parentheses, with higher precedence than addition `+` or multiplication `*`
- `^`: Exponentiation, with higher precedence than addition `+` or multiplication `*`

⟨number⟩ represents an integer in decimal notation, guaranteed to be between 1 and 10^9. Also, '`log(`' ⟨expr⟩ '`)^`' ⟨number⟩ represents $(\log(\langle\text{expr}\rangle))^{\langle\text{number}\rangle}$.

For example, the following strings can be ⟨expr⟩ symbols:

- `N+log(N)*N`: Represents $N + \log(N) \times N$.
- `N^1+N^2+log(N)+log(N)^1000000000`: Represents $N^1 + N^2 + \log(N) + (\log(N))^{1000000000}$.
- `N*(N+(log(N+N)^2*N))+(((N)))`: Represents $N \times (N + (\log(N+N))^2 \times N) + (((N)))$.
- `(log((N)))`: Represents $(\log((N)))$.

The following strings cannot be ⟨expr⟩ symbols:

- `(log(N)+N)^2`: The form '`(`' ⟨expr⟩ '`)^`' ⟨number⟩ is not used in ⟨factor⟩.
- `(log(N))^2`
- `(N`
- `)N(`

- `N^1000000001`
- `N^02`
- `N^0`
- `N^N`
- `2`
- `log(3)`
- `N-log(N)`
- `log(N)/N`

While $F(N)$ may not be defined for all positive integers N, for any input, there exists a positive integer N_0 such that $F(N)$ is defined for all positive integers $N \geq N_0$.

Therefore, define the set S of all non-negative integer pairs (a, b) such that the limit

$$\lim_{N \to \infty} \frac{F(N)}{N^a (\log N)^b}$$

converges to a finite value (including 0). Output the **lexicographically smallest pair** (a, b) in S.

Here, a non-negative integer pair (a, b) is the lexicographically smallest in S if it belongs to S, and for any other pair (a', b') in S, either:

- $a < a'$
- $a = a'$ and $b \leq b'$

It is proven that S is not an empty set, and furthermore, there exists the lexicographically smallest pair in S.

Input

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

```
F
```

- The function $F(N)$ is given as a string F following the $\langle \text{expr} \rangle$ symbol defined in the problem statement.
- $1 \leq |F| \leq 10^5$

Output

Output the lexicographically smallest pair (a, b) of S separated by a space.

Examples

standard input	standard output
N*log(N^2)*log(N)+N+log(N^1+N)^2*N	1 2
N*log(log(N))	1 1
(((N))*N^234567890+N^2)	234567891 0

Note

In the first example, $F(N) = N \times \log(N^2) \times \log(N) + N + (\log(N^1 + N))^2 \times N$.

For this case, non-negative integer pairs (a, b) such that the limit in the problem statement converges to a finite value include $(a, b) = (1, 2), (1, 3), (2, 0)$, etc. For these pairs, the limits are as follows:

$$\lim_{N \to \infty} \frac{F(N)}{N^1 (\log N)^2} = 3$$

$$\lim_{N \to \infty} \frac{F(N)}{N^1 (\log N)^3} = 0$$

$$\lim_{N \to \infty} \frac{F(N)}{N^2 (\log N)^0} = 0$$

Note that 0 is considered a finite value. On the other hand, for example, $(a, b) = (1, 1)$ leads to:

$$\lim_{N \to \infty} \frac{F(N)}{N^1 (\log N)^1} = \infty$$

and does not converge to a finite value.

It can be shown that within the set S of all pairs satisfying the conditions, $(a, b) = (1, 2)$ is lexicographically the smallest.

In the second example, $F(N) = N \times \log(\log(N))$. For $(a, b) = (1, 1)$:

$$\lim_{N \to \infty} \frac{F(N)}{N^1 (\log N)^1} = 0$$

and converges to a finite value.

It can be shown that within the set S of all pairs satisfying the conditions, $(a, b) = (1, 1)$ is lexicographically the smallest.

In the third example, $F(N) = \big(((N)) \times N^{234567890} + N^2\big)$.

Problem G. Cola

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

Alice has a favorite permutation $P = (P_1, P_2, \ldots, P_N)$ of $(1, 2, \ldots, N)$. Bob found out that if he guesses P, he will receive a cola from Alice. So, Bob decides to ask Alice questions to guess P.

Bob can ask the following question up to M times:

- Choose a permutation $Q = (Q_1, Q_2, \ldots, Q_N)$ of $(1, 2, \ldots, N)$ and ask Alice if her favorite permutation is Q.

Here, $M \le N$ holds.

Alice will respond to Bob's questions with the following actions:

- If $P = Q$, Alice will give a cola to Bob.
- If $P \ne Q$, Alice will tell Bob the smallest index i such that $P_i \ne Q_i$.

For example, if $P = (4, 3, 2, 1)$ and Bob asks the question with $Q = (4, 3, 1, 2)$, Alice informs Bob that there exists an index i such that $P_i \ne Q_i$, and the smallest such i is 3.

Note that even if Bob identifies P after the M-th question, he won't receive a cola.

Initially, Bob has no information about P. Please calculate the maximum probability that Bob receives a cola from Alice, and output this probability modulo 998244353.

Definition of probability modulo 998244353

It can be proven that the probability sought in this problem will always be a rational number. Also, in the constraints of this problem, it is guaranteed that when the sought probability 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 $0 \le z < 998244353$ satisfying $y \equiv xz \pmod{998244353}$, so output z.

Input

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

N M

- All values in the input are integers.
- $1 \le M \le N \le 10^7$

Output

Output the answer.

Examples

standard input	standard output
2 1	499122177
1 1	1
167 91	469117530

Note

In the first example, since there is only one question allowed, and there are two possible permutations for P, Bob can receive a cola with a probability of $\frac{1}{2}$.

Note that even if Bob misses on the first question, he can still identify P, but he won't receive a cola.

In the second sample, Bob will always receive a cola with the first question.

Problem H. 404 Chotto Found

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

404 Only a Bit Found

[Back to Top]

You are given N strings $S_1, S_2, \ldots, S_N$. Find the number of non-empty strings T that satisfy the following condition:

- Among the N strings $S_1, S_2, \ldots, S_N$, there is exactly one string that contains T as a (consecutive) substring.

Input

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

$$N$$
$$S_1$$
$$S_2$$
$$\vdots$$
$$S_N$$

- $1 \leq N \leq 10^5$
- $1 \leq |S_i| \leq 10^6$ $(1 \leq i \leq N)$
- $\left(\sum_{i=1}^{N} |S_i|\right) \leq 10^6$
- S_i $(1 \leq i \leq N)$ consists of lowercase English letters.

Output

Output the answer.

Examples

standard input	standard output
2 abc ca	5
2 aab aab	0
1 aba	5
3 tokyoinstituteoftechnology tokyomedicalanddentaluniversity instituteofsciencetokyo	905

Note

Example 1

Considering the case of $T =$ '`a`', both $S_1 =$ '`abc`' and $S_2 =$ '`ca`' contain '`a`' as a substring, so the condition is not satisfied.

For $T =$ '`ab`', only $S_1 =$ '`abc`' contains '`ab`' as a substring, so the condition is satisfied.

For $T =$ '`d`', neither $S_1 =$ '`abc`' nor $S_2 =$ '`ca`' contains '`d`' as a substring, so the condition is not satisfied.

The strings satisfying the condition are $T =$ '`b`', '`ab`', '`bc`', '`ca`', '`abc`', totaling 5.

Example 2

Considering the case of $T =$ '`ab`', both $S_1 =$ '`aab`' and $S_2 =$ '`aab`' contain '`ab`' as a substring, so the condition is not satisfied.

There are no strings satisfying the condition.

Example 3

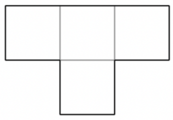
The strings satisfying the condition are $T =$ '`a`', '`b`', '`ab`', '`ba`', '`aba`', totaling 5.

Problem I. T Tile Placement Counting

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

A whale is elegantly swimming with its tail fin emerging from the water surface. This is a tiling problem.

Determine the number of ways to tile an $H \times W$ grid with tiles in the shape of the letter T, as shown in the figure below. Calculate the result modulo 998244353.



When tiling the grid with the T-shaped tiles, the following conditions must be satisfied:

- Tiles must be placed along the grid cells.
- Tiles must not protrude beyond the grid.
- Different tiles must not cover the same grid cell.
- There should be no grid cell left uncovered by any tile.

Additionally, tiles can be used with rotation, without distinguishing between the front and back, and without distinguishing between different tiles. Furthermore, arrangements of tiles that match only after rotation or reflection are considered distinct.

Input

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

H W

- All values in the input are integers.
- $1 \le H \le 30$
- $1 \le W \le 10^{18}$

Output

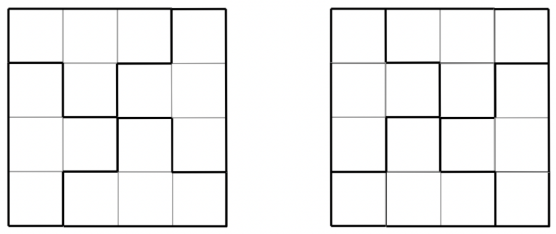
Output the answer.

Examples

standard input	standard output
4 4	2
2 8	0
12 3456	491051233

Note

In the first example, there are two possible ways to tile the grid with tiles as follows:



In the second example, there is no way to tile the grid with tiles.

Problem J. Set Construction

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

You are given an integer $N \geq 2$ and an integer M such that $2 \leq M \leq \frac{N(N+1)}{2}$. Construct a set A of non-negative integers satisfying the following conditions:

- If $x \in A$, then $0 \leq x \leq 2^N - 1$.
- $0 \in A$.
- $2^N - 1 \in A$.
- If $x, y \in A$, then $(x \text{ AND } y) \in A$.
- If $x, y \in A$, then $(x \text{ OR } y) \in A$.
- The number of elements in A is equal to M.

Here, AND denotes the bitwise AND operation, and OR denotes the bitwise OR operation.

Given 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 case_i $(1 \leq i \leq T)$ is given in the following format:

$$
N \ M
$$

- All values in the input are integers.
- $1 \leq T \leq 30$
- $2 \leq N \leq 60$
- $2 \leq M \leq \frac{N(N+1)}{2}$

Output

For each test case, output M distinct non-negative integers forming a set A that satisfies all the conditions given in the problem statement. You can output the elements in any order.

Note that it can be proven that a valid answer always exists under these constraints.

Example

standard input	standard output
3 3 5 4 8 60 2	0 1 3 5 7 0 1 3 7 8 9 11 15 0 1152921504606846975

Note

For the first test case, choosing $A = \{0, 1, 3, 5, 7\}$ satisfies all the conditions in the problem statement. For example, $(3 \text{ AND } 5) = 1 \in A$, and $(3 \text{ OR } 5) = 7 \in A$.

Any A that satisfies the conditions is acceptable; for instance, the output '`7 1 4 0 5`' is also valid. The elements in the output do not need to be in ascending order.

The output '`1 2 3 5 7`' is not valid because $0 \notin A$.

The output '`0 3 4 5 7`' is not valid because $3, 5 \in A$, but $(3 \text{ AND } 5) = 1 \notin A$.

The output '`7 7 7 0 0`' is not valid. Note that the set should not be a multiset.

Problem K. Dense Planting

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

You are given an integer K. Construct an undirected graph that satisfies the following conditions:

- The number of vertices N is between 1 and 100 (inclusive).
- The number of edges M is at most 1000.
- Assuming all edges are distinguishable, there are exactly K spanning trees in the graph. In other words, among the 2^M ways of choosing some edges from the M edges and removing the rest, there are exactly K ways such that the remaining edges form a tree.

Input

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

K

- K is an integer.
- $1 \le K \le 10^9$

Output

Output an undirected graph that satisfies the conditions in the following format. If there are multiple graphs that satisfy the conditions, you may output any of them.

N M
U_1 V_1
$\vdots$
U_M V_M

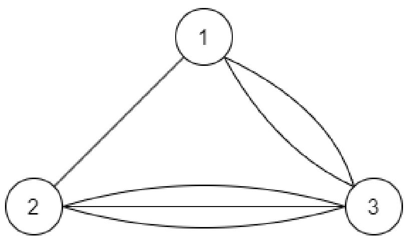
U_i, V_i $(1 \le i \le M)$ represent that the i-th edge connects vertices U_i and V_i.

Examples

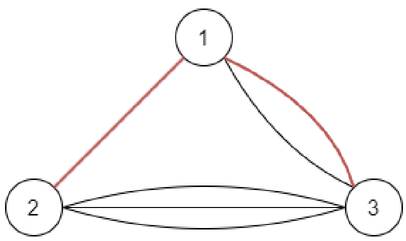
standard input	standard output
11	3 6 1 2 1 3 1 3 2 3 2 3 2 3
54	4 10 1 2 2 3 2 3 2 3 3 4 3 4 3 4 4 1 4 1 4 1

Note

In the first example, the output graph is represented by the following diagram:



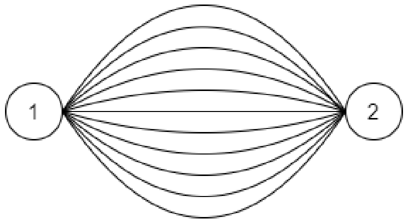
For example, choosing the following 2 edges forms a spanning tree of this graph:



- A spanning tree consisting of edges $1-2$ and $1-3$ can be selected in 2 ways.
- A spanning tree consisting of edges $1-2$ and $2-3$ can be selected in 3 ways.
- A spanning tree consisting of edges $1-3$ and $2-3$ can be selected in 6 ways.

Therefore, there are a total of 11 spanning trees.

Additionally, the following graph also has 11 spanning trees, so the following output is also considered correct:



standard output
2 11 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2

Problem L. Next TTPC 3

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

The TokyoTech Programming Contest is a programming competition held once a year. There are four uppercase English strings, S_1, S_2, S_3, S_4. Starting from next year, the abbreviation T_x for the TokyoTech Programming Contest held x years later is determined as follows:

- T_x is a string consisting of four uppercase English characters.
- The i-th character of T_x $(1 \leq i \leq 4)$ is equal to the $(((x-1) \bmod |S_i|)+1)$-th character of S_i. (Here, $|S_i|$ represents the length of the string S_i).

You are given a positive integer N. After how many years will the abbreviation be `TTPC` for the N-th time?

Input

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

N
S_1
S_2
S_3
S_4

- N is an integer.
- $1 \leq N \leq 10^6$
- $1 \leq |S_i| \leq 10^3$ $(1 \leq i \leq 4)$
- S_i $(1 \leq i \leq 4)$ consists of uppercase English characters.

Output

Output a positive integer x that satisfies the following two conditions. If there is no such x, output `-1` instead.

- $T_x =$ `TTPC`
- The string `TTPC` appears N times in $T_1, T_2, \ldots, T_x$

Examples

standard input	standard output
3 TTPC TLE P AC	34
670055 TF OITFKONTO GFPPNPWTZP CCZFB	-1
910359 TOKYO TECH PROGRAMMING CONTEST	1401951321

Note

In the first example, the abbreviation becomes `TTPC` for the first time after 10 years, for the second time after 22 years, and for the third time after 34 years. Therefore, the answer is 34.

In the second example, the abbreviation `TTPC` never occurs.

Problem M. Sum is Integer

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

You are given $2N$ positive integers $(p_1, q_1, p_2, q_2, \dots, p_N, q_N)$.

Find the number of pairs of integers (l, r) that satisfy the following conditions:

- $1 \le l \le r \le N$
- $\displaystyle\sum_{i=l}^{r} \frac{p_i}{q_i}$ is an integer.

Input

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

N
$p_1\ q_1$
$p_2\ q_2$
$\vdots$
$p_N\ q_N$

- All values in the input are integers.
- $1 \le N \le 2 \times 10^5$
- $1 \le p_i \le q_i \le 10^5$ $(1 \le i \le N)$

Output

Output the answer.

Examples

standard input	standard output
4 1 6 1 3 1 2 1 2	2
5 1 1 2 2 3 3 4 4 5 5	15
2 1 99999 99999 100000	0

Note

In the first example, there are two pairs (l, r) that satisfy the conditions: $(l, r) = (1, 3), (3, 4)$. In fact,

- $\sum_{i=1}^{3} \frac{p_i}{q_i} = \frac{1}{6} + \frac{1}{3} + \frac{1}{2} = 1$
- $\sum_{i=3}^{4} \frac{p_i}{q_i} = \frac{1}{2} + \frac{1}{2} = 1$

In the second example, all pairs of integers (l, r) with $1 \leq l \leq r \leq 5$ satisfy the condition.

In the third example, $\sum_{i=1}^{2} \frac{p_i}{q_i} = \frac{1}{99999} + \frac{99999}{100000} = \frac{9999900001}{9999900000} = 1.00000000010000100001...$ is not an integer.

Problem N. Bracket Sequestion

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3.5 seconds
Memory limit:	1024 megabytes

You are given a positive integer N and a prime number M.

A string consisting of `(`, `?`, `)` is called **good** if it satisfies the following conditions:

- By replacing each `?` in the string with either `(` or `)`, it can be transformed into a **balanced brackets sequence**.

Find the number of good strings of length $2N$, modulo M.

Here, a **balanced brackets sequence** is defined as one of the following:

- An empty string.
- There exists a balanced brackets sequence A, and the string obtained by concatenating `(`, A, `)` in this order.
- There exist non-empty balanced brackets sequences A and B, and the string obtained by concatenating A, B in this order.

Input

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

```
N M
```

- All values in the input are integers.
- $1 \leq N \leq 9 \times 10^8$
- $9 \times 10^8 \leq M \leq 10^9$
- M is a prime number.

Output

Output the answer.

Examples

standard input	standard output
1 998244353	4
2 900000011	28
999937 999999937	170733195
167167924 924924167	596516682

Note

In the first example, there are 4 good strings of length $2N(= 2)$: `()`, `(?`, `?)`, `??`.

Problem O. 2D Parentheses

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

On a two-dimensional plane, there are N open parentheses and N closing parentheses. The coordinates of the i-th open parenthesis are $(x_{1,i}, y_{1,i})$, and the coordinates of the i-th closing parenthesis are $(x_{2,i}, y_{2,i})$.

You perform the following operation N times to arrange rectangles on the plane:

- Select one open parenthesis and one closing parenthesis on the plane. Here, if the coordinates of the selected open parenthesis are (x_1, y_1) and the coordinates of the selected closing parenthesis are (x_2, y_2), then it must be the case that $x_1 < x_2$ and $y_1 < y_2$.
- Remove the selected open and closing parentheses from the plane and instead arrange a rectangle with vertices at the points $(x_1, y_1), (x_1, y_2), (x_2, y_2), (x_2, y_1)$ on the plane.

Determine if it is possible to arrange N rectangles on the plane in a way that satisfies the following condition. If possible, provide one such arrangement.

- For any two different rectangles, the common area is either 0, or one is completely contained within the other.

Input

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

$$
\begin{array}{l}
N \\
x_{1,1}\ y_{1,1} \\
x_{1,2}\ y_{1,2} \\
\vdots \\
x_{1,N}\ y_{1,N} \\
x_{2,1}\ y_{2,1} \\
x_{2,2}\ y_{2,2} \\
\vdots \\
x_{2,N}\ y_{2,N}
\end{array}
$$

- All values in the input are integers.
- $1 \le N \le 2 \times 10^5$
- $-10^9 \le x_{p,i}, y_{p,i} \le 10^9$
- If $(p, i) \ne (q, j)$, then $(x_{p,i}, y_{p,i}) \ne (x_{q,j}, y_{q,j})$.

Output

If there is no arrangement that satisfies the conditions, output `No` on a single line.

If there is an arrangement that satisfies the conditions, first output `Yes` on a single line. After that, for each $i = 1, 2, \ldots, N$, output the index c_i of the closing parenthesis that corresponds to the i-th open parenthesis.

If there are multiple arrangements that satisfy the conditions, you can output any of them.

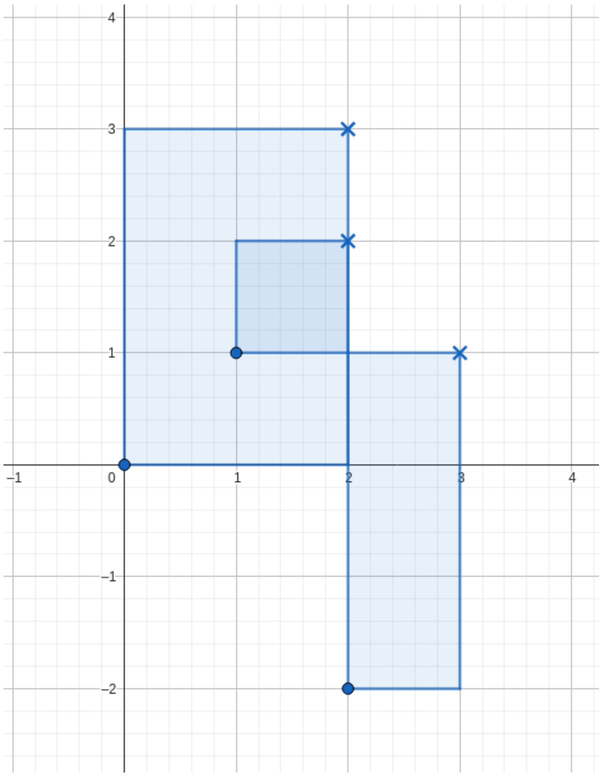
Examples

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

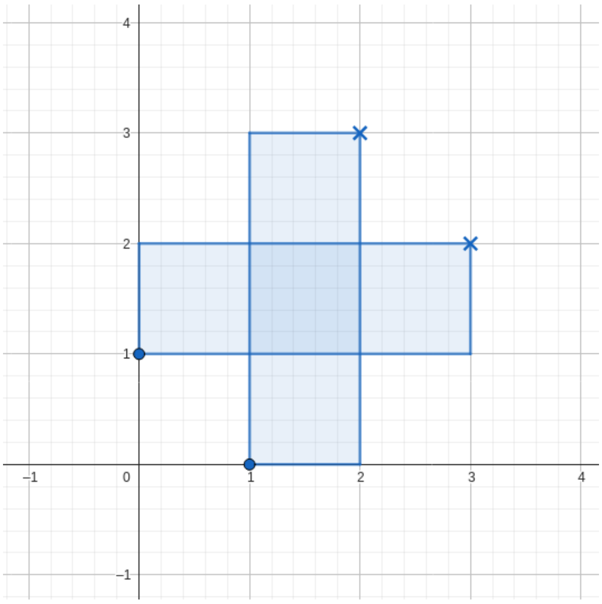
Note

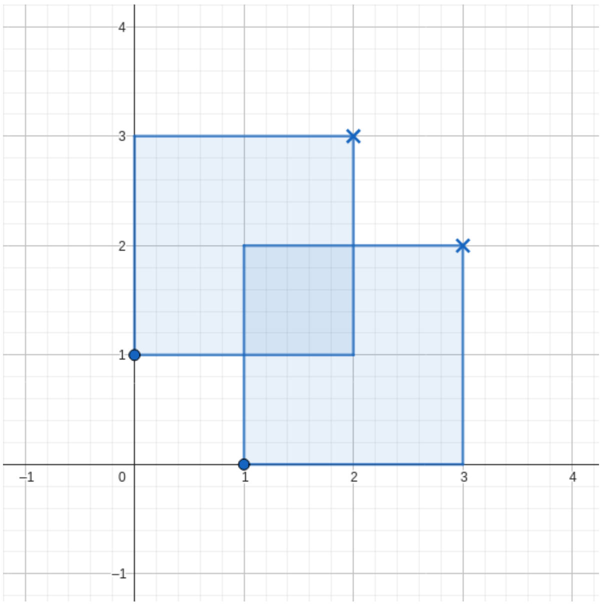
In the first example, arranging rectangles as shown in the figure satisfies the conditions.

Here, in the figure, circles represent open parentheses, and crosses represent closing parentheses.

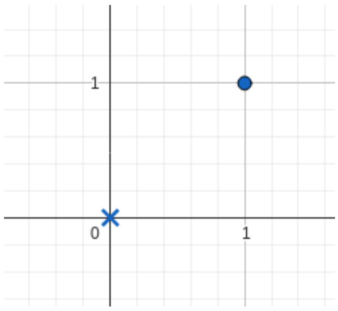


In the second example, as shown in the figure, it is impossible to satisfy the conditions regardless of how rectangles are arranged.





In the third example, unfortunately, it is not possible to arrange a rectangle.



Problem P. Bridge Elimination

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

There is an undirected graph with N vertices. The vertices of this graph are numbered from 1 to N, and each vertex i $(1 \le i \le N)$ has an integer A_i written on it. Although there are no edges in this graph, you are allowed to freely add edges.

There are $2^{\frac{N(N-1)}{2}}$ ways to add edges to make the graph a simple graph. Calculate the following **score** for each of them and find the sum of the scores modulo 998244353.

- When the graph is not connected, the **score** is 0.
- When the graph is connected, let G be the graph obtained by removing bridges from the original graph. Consider the sum of integers written on the vertices for each connected component of G, and define the product of these sums as the **score**.

Input

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

N
$A_1\ A_2\ \ldots\ A_N$

- All values in the input are integers.
- $1 \le N \le 400$
- $0 \le A_i < 998244353$ $(1 \le i \le N)$

Output

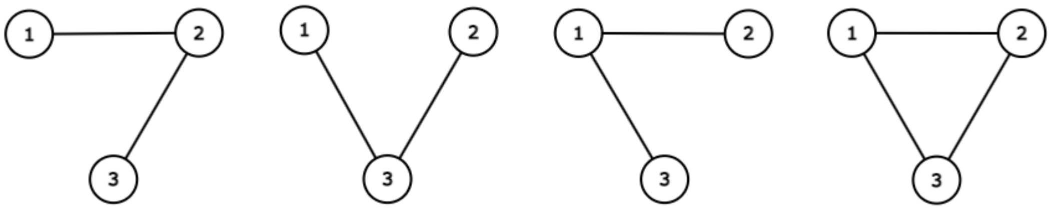
Output the answer.

Examples

standard input	standard output
3 8 5 9	1102
5 4 2 1 3 10	63860
7 229520041 118275986 281963154 784360383 478705114 655222915 970715006	35376232

Note

In the first example, the simple connected undirected graphs with 3 vertices are the following 4 patterns:



The scores are $360, 360, 360, 22$ respectively, so the answer is 1102.