Problem A. Master Zhu and Magic Numbers

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Master Zhu has n magic numbers. The i-th number a_i can be represented by a binary string of length i which can contain leading zeroes. When we reverse and concatenate these binary strings, we get a long string s of length $\frac{n(n+1)}{2}$. A substring from $s\left[\frac{i(i-1)}{2}\right]$ to $s\left[\frac{i(i+1)}{2}-1\right]$ inclusive is the binary representation of a_i, **from lowest to highest** digit. Here, the long string s is indexed starting from zero.

One day, Rin inputs Master Zhu's magic numbers into a program to create n new magic numbers. The i-th new number is b_i. Here is the code of the program in a C-like language.

```
for (int i = 1; i <= n; i++) {
  b[i] = 0,
  flag[i] = 0;
}

for (int i = 1; i <= n; i++) {
  for (int j = 1; j <= n; j++) {
    if (((1 << (j - 1)) & a[i]) > 0) {
      if (!flag[i]) {
        b[i] = b[j],
        flag[i] = 1;
      } else {
        b[i] = b[i] & b[j];
      }
    }
  }
  b[i] = b[i] ^ (1 << (i - 1));
}
```

In the code above, "`x << y`" is bitwise shift to the left, which is equivalent to multiplying `x` by $2^{\mathtt{y}}$, "`x & y`" is bitwise AND, and "`x ^ y`" is bitwise XOR. We assume that the numbers a_i and b_i can have arbitrarily many bits.

After that, Rin wants to ask q questions. The i-th question is a number c_i which can be represented in binary notation as a string of length len_i and can contain leading zeroes. When we reverse and concatenate these binary strings, we get a long string t of length L_q, where $L_k = \sum_{i=1}^{k} len_i$. A substring from $t\left[L_{i-1}\right]$ to $t\left[L_i - 1\right]$ inclusive is the binary representation of c_i, **from lowest to highest** digit. Here, the long string t is indexed starting from zero.

For each question, Rin requires Illya to calculate the number d_i:

```
for (int i = 1; i <= q; i++) {
  d[i] = 0;
  for (int j = 1; j <= min (n, len[i]); j++) {
    if (((1 << (j - 1)) & c[i]) > 0) {
      d[i] = d[i] | b[j];
    }
  }
}
```

Here, "`x | y`" is bitwise OR. We assume that the numbers c_i and d_i can have arbitrarily many bits.

The answer to the i-th question is the number of ones in the binary representation of d_i. Help Illya answer all the questions!

Input

The first line of the input contains two integers n and m denoting the number of magic numbers and the number of ones in the long string s ($1 \le n \le 5000$, $1 \le m \le 10^6$).

The second line contains m integers, the i-th integer p_i denotes that $s[p_i] = 1$, and all other digits of s are equal to 0 ($0 \le p_i < \frac{n(n+1)}{2}$, all p_i are distinct).

The third line contains two integers q and r denoting the number of questions and the number of ones in the long string t ($1 \le q \le 5000$, $1 \le r \le 10^6$).

The fourth line contains q integers, the i-th integer len_i denotes the length of binary representation of the i-th query c_i ($1 \le len_i \le 10^9$). It is guaranteed that $L_q = \sum_{i=1}^{q} len_i$ is at most 10^9.

The fifth line contains r integers, the i-th integer z_i denotes that $t[z_i] = 1$, and all other digits of t are equal to 0 ($0 \le z_i < L_q$, all z_i are distinct).

Output

For each question i, print a single line with a single integer: the number of ones in the binary representation of d_i.

Example

standard input	standard output
3 4 0 1 4 5 3 6 2 3 3 0 1 2 4 6 7	2 3 3

Problem B. Master Zhu and Chessboard

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Master Zhu has a rectangular board consisting of N rows and M columns. In the i-th row, the squares from the column L_i to the column R_i inclusive are colored black, and all other squares are colored white. Additionally, it is known that $L_i \le L_{i+1}$ and $R_i \le R_{i+1}$. Now Master Zhu is going to place some chess pieces on several black squares so that for each black square, there is at least one chess piece in its row or in its column.

Find the minimum number of chess pieces he should place.

Input

The first line of the input contains two integers N and M: the number of rows and columns ($1 \le N, M \le 100$). Each of the next N lines contains two integers L_i and R_i ($1 \le L_i \le R_i \le M$). It is guaranteed that $L_i \le L_{i+1}$ and $R_i \le R_{i+1}$.

Output

Output the minimum number of chess pieces Master Zhu should place.

Examples

standard input	standard output
3 3 1 1 2 2 3 3	3
2 4 1 3 2 4	2
3 2 1 2 1 2 1 2	2

Problem C. Master Zhu and Candies

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Master Zhu puts n heaps of candies on the table. Two players are playing the following game: on their turn, each player can either pick any positive number of candies from the same heap, or split some heap into three smaller non-empty heaps. Player who picks the last candy wins.

Master Zhu wants you to find out which player will win the game if both play optimally.

Input

The first line of input contains an integer n indicating the number of heaps ($1 \le n \le 10^6$). The next line contains n integers $s_1, \ldots, s_n$ representing the number of candies in each heap ($1 \le s_i \le 10^9$).

Output

If the first player wins, print "`First`", otherwise print "`Second`".

Examples

standard input	standard output
2 4 4	Second
3 1 2 4	First

Problem D. Master Zhu and Rikka

Input file:	*standard input*
Output file:	*standard output*
Time limit:	4 seconds
Memory limit:	512 mebibytes

As we all know, Master Zhu is the most powerful man in the universe. He has infinite power to protect the world.

A magical rooted tree grows in Master Zhu's garden. The tree contains n vertices and $n-1$ edges. The root of the tree is in vertex 1. Each vertex i contains an amount of *Zhu power* equal to a_i.

Little Rikka is a curious girl: she has m questions to ask to Master Zhu. But now Master Zhu is busy protecting our world, so he wants you to help him answer Rikka's questions.

Each Rikka's question has the format "$t\ u\ v\ a\ b$".

- If t is 1, then u is equal to v, and Rikka looks at the subtree rooted in vertex u and wants to know the GCD (greatest common divisor) of S_a and S_b, where S_a is the sum of numbers that appear exactly a times in this subtree, and S_b is the sum of numbers that appear exactly b times in this subtree.
- If t is 2, Rikka looks at the simple path between vertices u and v and wants to know the GCD of T_a and T_b, where T_a is the sum of numbers that appear exactly a times on this path, and T_b is the sum of numbers that appear exactly b times on this path.

Here, for any x, we define $\mathrm{GCD}(x, 0) = \mathrm{GCD}(0, x) = x$.

Input

The first line of input contains an integer T, the number of test cases ($1 \le T \le 10$).

The first line of each test case contains two integers n and m: the number of vertices in the magical tree and the number of Rikka's questions ($1 \le n, m \le 10^5$).

The next line contains n integers a_1, a_2, ..., a_n: the amount of Zhu power in each vertex ($1 \le a_i \le 10^9$).

Each of the next $n-1$ lines contains two integers u and v and denotes an edge connecting vertices u and v ($1 \le u, v \le n$). It is guaranteed that together, these edges form a tree.

Each of the next m lines contains one of Rikka's questions in the format described above ($1 \le u, v \le n$, $1 \le a, b \le n$).

Output

For each Rikka's question, print the answer on a separate line.

Example

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

Problem E. Master Zhu and Palindromes

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

Master Zhu has a string $S[1, \ldots, n]$. This string can contain only the first five lowercase English letters. Another peculiar property of S is that the length of each palindrome substring in S is less than 20.

For a palindrome string $P[1, \ldots, k]$, its *tail* is the string $P[\lfloor k/2 \rfloor + 1, \ldots, k]$. For example, the tail of the string "`aba`" is "`ba`", and the tail of the string "`caac`" is "`ac`".

Given L, R, and a string T, Master Zhu wants you to find the number of different palindrome substrings in $S[L, \ldots, R]$ such that T is a prefix of their tails. Here, two substrings are considered different if their starting or ending positions in S differ.

Input

The first line of input contains one integer C, the number of test cases ($1 \le C \le 50$).

The first line of each test case contains a string S consisting only of the first five lowercase English letters ($1 \le |S| \le 10^5$, the length of each palindrome substring in S is less than 20).

The second line contains one integer q, the number of queries ($1 \le q \le 10^5$). Each of the next q lines contains two integers L and R and a string T consisting only of the first five lowercase English letters ($1 \le L \le R \le |S|$, $1 \le |T| \le 10$).

Output

For each query, print a single line with a single integer: the number of different palindrome substrings in $S[L, \ldots, R]$ such that T is a prefix of their tails.

Example

standard input	standard output
1 bceaeeddee 5 5 8 e 3 5 e 1 2 a 5 9 d 5 9 de	3 2 0 4 1

Problem F. Master Zhu and Video

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

Master Zhu is watching a video. He wants to adjust the sound volume from p dB to q dB. Master Zhu is powerful enough to control the sound volume mentally. He needs one second to issue one order. So, each second, he may either issue an order "Up!", issue an order "Down!", or do nothing.

If he issues an order "Up!", the volume increases by 1 dB. An order "Down!" works in a more compilcated way: if on the previous second, there was an order "Down!" and the volume decreased by x dB, it decreases by $2 \cdot x$ dB, otherwise the volume decreases by 1 dB.

If after some order the volume becomes negative, the system will break, so Master Zhu must be careful.

Find the minimum time needed for Master Zhu to adjust the volume from p dB to q dB.

Input

The first line of input contains one integer T, the number of test cases ($1 \le T \le 3 \cdot 10^5$).

Each of the next T lines contains two integers p and q ($0 \le p, q \le 10^9$).

Output

For each test case, print a single line with a single integer: the minimum time in seconds Master Zhu needs to adjust the volume.

Example

standard input	standard output
2 1 5 7 3	4 4

Problem G. Master Zhu and Polygons

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Consider a regular polygon with N vertices, where N is odd. The vertices are numbered from 1 to N in circular order. We can select M of these vertices to form a convex polygon on them. Master Zhu wants you to find how many of such possible convex polygons have exactly K acute angles. As their number can be very large, find the answer modulo $10^9 + 7$. Two polygons are considered different if the sets of numbers in their vertices differ.

Input

The first line of input contains one integer T, the number of test cases ($1 \le T \le 5 \cdot 10^4$).

Each test case is described by a single line containing three integers N, M, and K ($3 \le N \le 10^6$, $3 \le M \le N$, $0 \le K \le M$, and N is odd).

Output

For each test case, print the answer modulo $10^9 + 7$ on a separate line.

Example

standard input	standard output
1 5 4 2	5

Problem H. Master Zhu and Root

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	512 mebibytes

Master Zhu has a number n. He asks you q queries, the i-th query is a pair of integers (x_i, y_i). For i-th query, Master Zhu would like you to find the smallest non-negative integer k_i such that, for some p which is a prime divisor of n, the equivalence $x_i^{k_i} \equiv y_i$ modulo p holds, or to determine that no such k_i exists.

In this problem, we consider that $0^0 = 1$.

Input

The first line of input contains two integers n and q ($1 \le n \le 10^8$, $1 \le q \le 10^5$). Each of the next q lines contains two integers x_i and y_i ($0 \le x_i, y_i \le 10^9$).

Output

For each query, print the answer on a separate line. If you found k_i, print it, otherwise print the number -1.

Example

standard input	standard output
175 2 2 1 2 3	0 3

Problem I. Master Zhu and Binary Trees

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

One day, Master Zhu invented an interesting game which he named Tree Maker. In this game, all trees are binary trees.

Initially, there is a tree with only one vertex and a cursor on it. In order to build the tree, the player can control the cursor to perform five operations described below:

- "0": Jump to the parent of the current vertex (it must exist).
- "1": Jump to the left child of the current vertex (it must exist).
- "2": Jump to the right child of the current vertex (it must exist).
- "3 x": Generate an arbitrary binary tree with x vertices and make it the left subtree of the current vertex (before such operation, the current vertex must have no left child).
- "4 x": Generate an arbitrary binary tree with x vertices and make it the right subtree of the current vertex (before such operation, the current vertex must have no right child).

When an operation is performed, the log system writes down a record of it.

Rin played this game for a whole day yesterday. As a forgetful man, although Rin knew the shape of the tree while playing, after a sleep he forgot it. All he has now is the log of operations. Rin wants to know: according to the log, how many possible shapes the tree could have had yesterday after all operations?

Can you answer this question? As the answer may be very large, it is sufficient to find it modulo $10^9 + 7$.

Input

The first line of input contains an integer n denoting the number of lines in the log ($1 \le n \le 500$).

Then follow n lines of the log. The format of the log is as described above.

It is guaranteed that, for every operation of the types 3 and 4, the integer x is positive, and the total number of vertices in the tree will never exceed 500. You can also assume that there exists at least one tree such that the given log is valid for that tree.

Output

Print a single line with a single integer: the answer to Rin's question modulo $10^9 + 7$.

Examples

standard input	standard output
2 3 3 4 3	25
2 3 3 1	5

Problem J. Master Zhu and Instability

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Consider an array A of length N. The *instability* of A is defined as

$$\sum_{i=1}^{N-1} (|A[i+1] - A[i]|).$$

Master Zhu wants to stabilize an array. In order to do that, he wants to select an integer X and change every element $A[i]$ to $(A[i] \oplus X)$. Here, $u \oplus v$ is the bitwise XOR of u and v.

Find the smallest non-negative integer X Master Zhu must choose to minimize the instability of a given array, and calculate the resulting instability.

Input

The first line of input contains an integer N ($1 \le N \le 10^5$). Next line contains N integers $A[i]$, indicating the elements of the array ($0 \le A[i] < 2^{20}$).

Output

Print a single line containing two integers: the smallest non-negative integer X which must be used in order to reach the minimum possible instability and the resulting instability itself.

Examples

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

Problem K. Master Zhu and Math Problem

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	512 mebibytes

Master Zhu once came up with the following mathematical problem:

Given four integers, A, B, C, and D, how many different quadruples of integers (a, b, c, d) are there which satisfy all the following conditions:

$$\begin{gathered} a + c > b + d \\ a + d \geq b + c \\ 0 \leq a \leq A \\ 0 \leq b \leq B \\ 0 \leq c \leq C \\ 0 \leq d \leq D \end{gathered}$$

Find the number of such quadruples. As the answer may be very large, it is sufficient to calculate it modulo $10^9 + 7$.

Input

The first line of input contains an integer T, the number of test cases ($1 \leq T \leq 1000$).

Each test case is given on a single line containing four integers A, B, C, and D ($0 \leq A, B, C, D \leq 10^{18}$).

Output

For each test case, output a single line with a single integer: the answer modulo $10^9 + 7$.

Example

standard input	standard output
1 2 1 1 1	10

Problem L. Master Zhu and the Leaper

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Consider an $n \times m$ rectangular board consisting of square cells. Master Zhu put a *leaper* at position $(1, 1)$, which is the upper left cell.

The leaper is able to jump from position (x_1, y_1) to position (x_2, y_2) if and only if the positive integers x_1, y_1, x_2, and y_2 satisfy the following conditions:

$$(x_2 - x_1)^2 + (y_2 - y_1)^2 = 5,$$
$$x_2 > x_1,$$
$$y_2 > y_1.$$

Unfortunately, there are some obstacles on the board. The leaper can never enter a square with an obstacle.

Master Zhu wants to move the leaper to position (n, m), which is the lower right cell of the board, by making zero or more jumps. Help him find the number of ways the leaper can achieve its goal. As the answer may be very large, calculate it modulo 110 119.

Input

The first line of input contains one integer T, the number of test cases ($1 \le T \le 540$).

The first line of each test case contains three integers n, m, and r: the height of the board, the width of the board, and the number of obstacles on the board, respectively ($1 \le n, m \le 10^{18}$, $0 \le r \le 100$).

Then follow r lines. Each of them contains two integers x and y: coordinates of an obstacle ($1 \le x \le n$, $1 \le y \le m$). It is guaranteed that all given obstacles are distinct, and the position $(1, 1)$ contains no obstacle.

Output

For each test case, print the answer modulo 110 119.

Example

standard input	standard output
5 1 1 0 3 3 0 4 4 1 2 1 4 4 1 3 2 7 10 2 1 2 7 1	1 0 2 1 5