Problem A. Mode

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

You are given an array a of length n. Define cnt_x as the number of occurrences of x in a.

Now you can do the following operation **at most once**: select a non-empty subarray $a_l, a_{l+1}, a_{l+2}, \dots, a_r$ and an integer $k \in [-10^9, 10^9]$, and add k to all the elements in the subarray.

Your first task is to find the maximum possible value of $W = \max\{cnt_x \mid x \in \mathbb{Z}\}$ after one operation. Your second task is to find all v such that $cnt_v = W$ can be achieved after one operation.

Input

The first line contains an integer T ($1 \le T \le 20$), the number of test cases.

Each test case consists of two lines. The first line contains a single integer n ($2 \le n \le 2 \cdot 10^5$), and the second line contains n integers denoting the array ($1 \le a_i \le 10^9$).

It is guaranteed that $\sum n \le 5 \cdot 10^5$, and a_i are not all the same.

Output

For each test case, output one integer on the first line, denoting the maximum value W. Then for all integers v satisfying the condition, output them in ascending order.

Example

`standard input`	`standard output`

4
5
1 2 3 2 1
5
1 1 3 1 1
6
2 4 2 4 8 8
5
1 2 3 4 5
	4
	1
	5
	1
	4
	2
	4
	8
	2
	1
	2
	3
	4
	5

Note

The values of W for the test cases are 4, 5, 4, 2.

Problem B. Tree

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

We generate two rooted trees with n vertices in the following way.

The first tree is generated as follows:

1. Vertex 1 is the root of the tree.
2. For all $i \in [2, n]$, we select one vertex from $[1, i-1]$ as the father of i.

The second tree is generated as follows:

1. Vertex n is the root of the tree.
2. For all $i \in [1, n-1]$, we select one vertex from $[i+1, n]$ as the father of i.

A way to generate the trees is *good* if and only if every vertex i which is a leaf in tree 1 is not a leaf in tree 2, and every vertex i which is not a leaf in tree 1 is a leaf in tree 2. The root of every tree is not a leaf, regardless of the number of adjacent edges.

Now for all $n \in [2, N]$, calculate the number of good ways to generate trees. Two ways are considered different if and only if there exists a vertex i such that the parent of i in at least one tree is different in these two ways. You should output the answer modulo M.

Input

The first line of input contains two integers N and M ($2 \le N \le 500$, $10 \le M \le 2^{30}$).

Output

Output $N-1$ lines: the answers for $n = 2, 3, \ldots, N$.

Example

standard input	*standard output*

`5 998244353`	`1`
	`2`
	`12`
	`120`

Problem C. Ramen

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	1024 mebibytes

Grammy likes to eat noodles. She divided a very long strip of noodle into N parts of unit length. Each part i has deliciousness a_i. She would like to fold the noodle into one piece of unit length before eating by repeating the following operation several (possibly, zero) times.

Let n be the current length of the noodle. In each operation, Grammy can choose a length ℓ such that $2\ell \le n$ and $a_i > 0$ for all $i \le \ell$, and fold the noodle $a_1, a_2, \ldots, a_\ell, a_{\ell+1}, \ldots, a_{2\ell}, a_{2\ell+1}, \ldots, a_n$ into $a_{\ell+1} + a_\ell, a_{\ell+2} + a_{\ell-1}, \ldots, a_{2\ell} + a_1, a_{2\ell+1}, \ldots, a_n$, where n is the length of the noodle before the operation. After the operation, the length will become $n - \ell$.

Grammy wants to know whether she can fold the noodle to length 1, can you tell her?

Input

The first line of input contains a single integer N ($1 \le N \le 100\,000$).

The second line contains N integers a_i ($-20\,000 \le a_i \le 20\,000$), representing the deliciousness of each part of the noodle.

Output

If Grammy can fold the noodle to length 1, output a single line with the word "YES". Otherwise, output a single line with the word "NO".

Examples

standard input	*standard output*
3 1 2 -5	YES
5 2 -5 2 3 1	NO

Problem D. Rotate Sum 2

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	1024 mebibytes

Grammy loves geometry. Today, she takes out her precious convex polygon and plays with it on a piece of paper. The polygon has n vertices numbered from 1 to n in counterclockwise order. For vertex i, the next vertex in this order is $i^+ = i \bmod n + 1$, and the previous one is $i^- = (i + n - 2) \bmod n + 1$.

Firstly, Grammy draws a horizontal line on the paper. Secondly, she chooses two vertices i and j of the polygon independently and equiprobably. Thirdly, she places the edge between vertex i and vertex i^- on the line, landing all other vertices above the line, and draws a vertical line through vertex j. Next, she rotates the polygon clockwise, taking vertex i as the rotation center, until vertex i^+ hits the line. When vertex i^+ hits the line, she changes the rotation center to vertex i^+ and rotates again until vertex i^{++} (the next after i^+) hits the line. She repeats this operation until vertex i hits the line again. Finally, she draws another vertical line through the vertex j and calculates the area between the trajectory of vertex j and the three lines.

Since you do not know which points Grammy will choose, you want to calculate the expected value of the area.

Input

The first line contains a single integer n ($3 \le n \le 100\,000$), denoting the number of vertices in the polygon.

Each of the following n lines contains two integers x_i and y_i ($-10^9 \le x_i, y_i \le 10^9$), denoting the coordinates of a vertex of the polygon. The vertices are given in counterclockwise order. It is guaranteed that the polygon is strictly convex.

Output

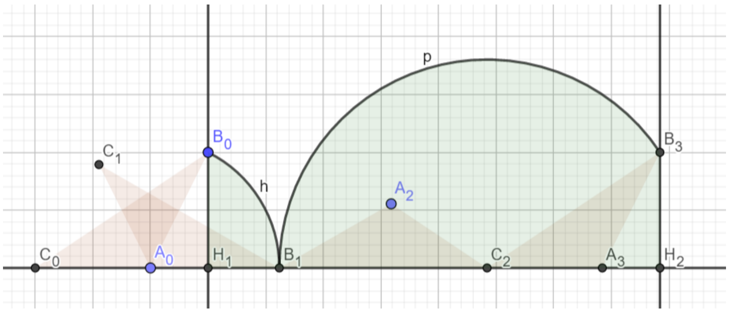
Output a single real number denoting the expected area. The answer is considered correct if its absolute of relative error does not exceed 10^{-4}.

Example

standard input	*standard output*

3
1 -1
1 1
-1 2
	18.763234503173919

Note



For the first example, if the i-th vertex is marked as A_0, and the j-th vertex is marked as B_0, then the polygon will be $A_3B_3C_2$ after 3 rotations, and the trajectory of vertex j is arc h and arc p. The area of the green part is the answer in this case.

Problem E. Smaller LCA

Input file:	*standard input*
Output file:	*standard output*
Time limit:	4 seconds
Memory limit:	1024 mebibytes

Grammy has a tree with vertices numbered from 1 to n. For each vertex as the root, she wants to know how many unordered pairs of points (x, y) have their lowest common ancestor z satisfy the inequality $z \leq x \cdot y$. Please count it for her.

Input

The first line contains a single integer n ($1 \leq n \leq 300\,000$), denoting the number of vertices of the tree.

Each of the next $n-1$ lines contains two integers u_i and v_i ($1 \leq u_i, v_i \leq n$), indicating that there is an edge between vertex u_i and vertex v_i. It is guaranteed that the given graph is a tree.

Output

Output n lines. The i-th line must contain a single integer: the number of pairs satisfying the condition when vertex i is the root.

Example

standard input	*standard output*
`5`	`15`
`1 2`	`15`
`4 2`	`15`
`2 5`	`15`
`3 5`	`14`

Problem F. Noodle

Input file:	*standard input*
Output file:	*standard output*
Time limit:	7 seconds
Memory limit:	1024 mebibytes

Putata is a boy who loves eating noodles. Now he's waiting for the great chef Budada to cook the most delicious noodle ever for him.

The noodle which Budada is cooking for him can be described as an array a of length n, where n is even. The amount of sauce initially at position i is a_i.

In one operation, Budada will do the following process.

1. Budada will fold the noodle, the length of the noodle will become $\frac{n}{2}$, the amount of sauce at position i will become the sum of the amounts of sauce at positions i and $n-i+1$. Formally, the amount of sauce at position i of the new noodle b_i satisfies $b_i = a_i + a_{n-i+1}$.

2. Then Budada will stretch the noodle to the original length, and the amount of sauce will be evenly divided. Formally, the amount of sauce at position i of the new noodle a'_i satisfies $a'_i = \frac{1}{2} \cdot b_{\lceil \frac{i}{2} \rceil}$.

Putata has a favorite position on the noodle, which is a certain position x. Now you are asked to answer q queries. In the i-th query, you should output the amount of sauce at position x after k operations. The x is the same for all queries, but k is given separately for each query.

It can be shown that the answer can be expressed as an irreducible fraction $\frac{x}{y}$, where x and y are integers and $y \not\equiv 0 \pmod{998\,244\,353}$. Output the integer equal to $x \cdot y^{-1} \pmod{998\,244\,353}$. In other words, output such an integer a that $0 \le a < 998\,244\,353$ and $a \cdot y \equiv x \pmod{998\,244\,353}$.

Since the input is quite large, you will have to use a generator to generate the queries, and you only have to output $\oplus_{i=1}^{q}(ans_i \cdot i)$. Please notice that this number is **not** taken modulo 998 244 353. Here, $\oplus$ means bitwise exclusive-or operation.

Input

The first line contains three integers *test*, T, and *seed*, which are an **unrelated variable**, the number of test cases, and the seed for generating test data. Please note that *test* **will not** be used to solve the problem, you can just ignore it. The generator code is given further below.

For each test case, the input will contain two lines.

The first line contains four integers n, q, x, and $k_{\max}$ ($1 \le n \le 2 \cdot 10^6$, $1 \le q \le 5 \cdot 10^7$, $1 \le x \le n$, $1 \le k_{\max} \le 10^{18}$).

The second line contains n integers, the i-th integer is a_i ($0 \le a_i < 998\,244\,353$).

It is guaranteed that $\sum n \le 2 \cdot 10^6$, $\sum q \le 5 \cdot 10^7$, and n is even.

Output

Output T lines. The i-th line must contain the answer to the i-th test case.

Example

standard input	*standard output*

0 2 13
4 2 1 3
1 4 2 3
6 2 3 3
6 2 5 3 1 4
	5
	499122191

Note

In the first test case of the sample, $\{a_i\}$ are $\{1, 4, 2, 3\}$ initially.

- After one operation, it becomes $\{2, 2, 3, 3\}$.
- After two operations, it becomes $\{\frac{5}{2}, \frac{5}{2}, \frac{5}{2}, \frac{5}{2}\}$.
- The generated queries are:
- The position is $x = 1$;
- The first query: $k = 0$, $a_x = 1$;
- The second query: $k = 1$, $a_x = 2$;
- The answer is $(1 \cdot 1) \oplus (2 \cdot 2) = 5$.

In the second test case, $\{a_i\}$ is $\{6, 2, 5, 3, 1, 4\}$ initially.

- After one operation, it becomes $\{5, 5, \frac{3}{2}, \frac{3}{2}, 4, 4\}$.
- After two operations, it becomes $\{\frac{9}{2}, \frac{9}{2}, \frac{9}{2}, \frac{9}{2}, \frac{3}{2}, \frac{3}{2}\}$.
- The generated queries are:
- The position is $x = 3$;
- The first query: $k = 2$, $a_x = \frac{9}{2}$, and $\frac{9}{2} \equiv 499\,122\,181 \pmod{998\,244\,353}$;
- The second query: $k = 0$, $a_x = 5$.
- The answer is $(499\,122\,181 \cdot 1) \oplus (5 \cdot 2) = 499\,122\,181 \oplus 10 = 499\,122\,191$.

The generator will be given below:

```
#include <bits/stdc++.h>
using namespace std;

unsigned long long rd (unsigned long long &x) {
  x ^= (x << 13);
  x ^= (x >> 7);
  x ^= (x << 17);
  return x;
}

int main () {
  int test, T;
  unsigned long long seed;
  scanf("%d%d%llu", &test, &T, &seed);
  for (int Case = 1; Case <= T; Case ++) {
    int n, q, x;
    long long k_max;
    scanf("%d%d%d%lld", &n, &q, &x, &k_max);
    vector<int> a(n + 1);
    for (int i = 1; i <= n; i ++) {
      scanf("%d", &a[i]);
    }
    for (int i = 1; i <= q; i ++) {
      long long k = rd(seed) % k_max;
      /*
      Code your solution here.
      */
    }
  }
}
```

Problem G. Geometry

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	1024 mebibytes

Grammy has a special two-dimensional coordinate system: the angle between the positive half-axis of the X-axis and the positive half-axis of the Y-axis is 60 degrees.

Consider the following graph. The vertices are all integer coordinates (x, y) such that at least one of x, y is odd and $-2a+1 \le x \le 2a-1$, $-2b+1 \le y \le 2b-1$, $-2c+1 \le x+y \le 2c-1$. The edges from (x, y) go to $(x, y+1)$, $(x, y-1)$, $(x+1, y)$, $(x-1, y)$, $(x+1, y-1)$, and $(x-1, y+1)$.

Find the size of the maximum independent set of vertices in this graph. Additionally, find the number of such sets modulo 998 244 353.

Input

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

Each of the following T lines contains three integers a, b, c $(1 \le a, b, c \le 10^6)$.

Output

Output T lines. Each line must contain two integers: the size of the maximum independent set and the number of such sets. **Please note that the size should not be taken modulo** 998 244 353.

Example

standard input	*standard output*
6	7 4
2 1 2	4 1
1 1 137	1124 31585548
3 94 95	23951 33873190
3 1998 1996	1289433675488 748596399
998244 353999 999999	23600 480090154
50 120 150	

Note

The following picture shows the situation for the first and second test case of the sample.

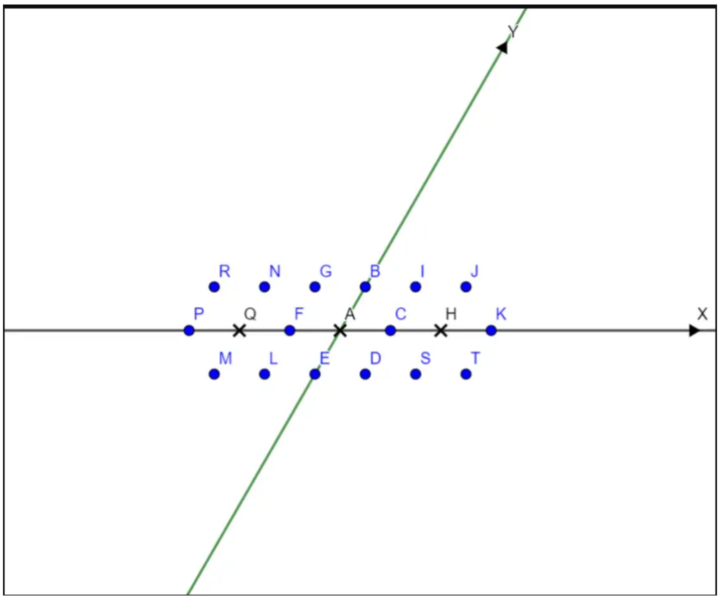
Point J has coordinates $(2, 1)$, point F has coordinates $(-1, 0)$, and point H has coordinates $(2, 0)$. Among these three points, only H has even X-coordinate and even Y-coordinate. The neighbours of point A are $BCDEFG$.

In the first test case, the points that satisfy the conditions are $NGBIJPFCKMLEDST$.

The size of the maximum independent set is 7, and there are 4 ways: $PNLBDJT$, $RMFBDJT$, $RMGECJT$, $RMGEISK$.

In the second test case, the points that satisfy the conditions are $GBIFCLED$.

The size of the maximum independent set is 4, and there is one way: $LGID$.



Picture for test case 1 and 2.

Problem H. Rectangle Placement

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	1024 mebibytes

Grammy has a rectangular grid with W vertical lines and H horizontal lines. She wants to draw two non-intersecting rectangles along the grid lines. One rectangle is allowed to be completely contained inside another, but the two rectangles cannot intersect at any point, including edges and corners.

Please count the number of different rectangle drawings, modulo 998 244 353. Two drawings are considered different if and only if a grid edge is colored in one of the drawings but not in the other.

Input

The only line contains two integers W and H ($4 \le W, H \le 10^9$).

Output

Output a single integer, denoting the number of different drawings modulo 998 244 353.

Examples

standard input	*standard output*
`4 5`	`275`
`723435135 239873451`	`832099301`

Problem I. Infectious Disease

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	1024 mebibytes

In the year of 2202, a strange disease begins to spread in a city of n people.

To prevent the disease from spreading, experts invented a strong vaccine called Mysterious Oscar. On day 0, one citizen is infected by the disease, and another citizen is vaccinated. If a person becomes vaccinated, he/she will be cured immediately and will not catch or spread the disease anymore.

On each subsequent day d ($d > 0$), the infected citizens will infect others one by one. Each of the citizens who were infected strictly before day d will choose one uninfected and unvaccinated citizen to infect equiprobably. If at some point, one infected citizen has no unvaccinated and uninfected citizens to choose from, then he/she will do nothing.

After infection, the vaccinated citizens will persuade others to take the vaccine one by one. Each of the citizens who were vaccinated strictly before day d will choose 2 different unvaccinated citizens equiprobably, and persuade them so that they become vaccinated. If at some point, one vaccinated citizen has less than 2 unvaccinated citizens to choose, then he/she will persuade all the remaining unvaccinated citizens to take the vaccine.

Grammy wants to know how many days will pass before the disease will be fully extinguished. Please tell her the expected number of days before all patients become cured.

It can be shown that the answer can be expressed as an irreducible fraction $\frac{x}{y}$, where x and y are integers and $y \not\equiv 0 \pmod{10^9+7}$. Output the integer equal to $x \cdot y^{-1} \pmod{10^9+7}$. In other words, output such an integer a that $0 \le a < 10^9+7$ and $a \cdot y \equiv x \pmod{10^9+7}$.

Input

The only line contains an integer n ($2 \le n \le 1.4 \cdot 10^7$), denoting the population of the city.

Output

Output a single integer, denoting the expected number of days before all patients become cured modulo 10^9+7.

Examples

standard input	*standard output*
2	1
114	505208013

Note

In the first sample, one citizen took the vaccine on day 0, and he/she persuaded the other citizen, the only patient, to take the vaccine on day 1, so the disease must be completely cured on day 1.

Problem J. Positive String

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

Grammy had a unique insight about strings. She thinks that a string is *positive* if and only if it is lexicographically larger than its reversal.

Now you are given a string, please find out how many contiguous substrings of it are positive according to Grammy's insight.

Input

The single line contains a string S $(1 \le |S| \le 200\,000)$, consisting of lowercase English letters only.

Output

Output a single integer denoting the number of positive substrings of S.

Examples

standard input	*standard output*
jjikkollp	4
pbpbppb	7

Problem K. DFS

Input file:	`standard input`
Output file:	`standard output`
Time limit:	8 seconds
Memory limit:	1024 mebibytes

You are given a rooted tree of n vertices, and r is the root of the tree. Each vertex x has value a_x.

Let us define the DFS procedure starting from x to find y:

1. Push x on the stack.
2. Check w, the top element of the stack. If $w = y$, the procedure ends. Otherwise, if there is at least one son of w which is not visited, choose one such son with equal probability and push it on the stack.
3. Repeat step 2 until there is no unvisited son.
4. Pop the top element from the stack.
5. Repeat step 2 until the stack is empty.

The procedure is legal if and only if y is in the subtree of x.

Define $f(x, y)$ as the expectation of the minimum value of all vertices which were pushed on the stack during the DFS procedure starting from x to find y.

Now we want to calculate $\sum f(x, y)$ for all legal pairs (x, y). It can be shown that the answer can be expressed as an irreducible fraction $\frac{x}{y}$, where x and y are integers and $y \not\equiv 0 \pmod{998\,244\,353}$. Output the integer equal to $x \cdot y^{-1} \pmod{998\,244\,353}$. In other words, output an integer a such that $0 \le a < 998\,244\,353$ and $a \cdot y \equiv x \pmod{998\,244\,353}$.

Input

The first line contains an integer T ($1 \le T \le 100$), denoting the number of test cases.

For each test case, the first line contains two integers n and r ($1 \le n \le 4 \cdot 10^5$, $1 \le r \le n$), denoting the number of vertices in the tree and the root.

The following line contains n integers, the i-th integer of them is $a_i (1 \le a_i \le 10^9)$ denoting the value of vertex i.

Each of the next $n - 1$ lines contains two integers u and v ($1 \le u, v \le n$), denoting an edge of the tree.

It is guaranteed that $\sum n \le 8 \cdot 10^5$. It is also guaranteed that the given graph is indeed a tree.

Output

Output T lines. Each line must contain one integer: the answer to the respective test case.

Example

standard input	*standard output*

4
1 1
1
3 3
3 3 4
3 1
3 2
6 1
5 2 4 1 3 6
1 2
1 6
2 3
2 4
4 5
5 1
5 4 3 2 1
1 2
1 3
3 4
3 5
	1
	16
	34
	499122202