Problem A. The Very Hard One

Input file:	`veryhard.in`
Output file:	`veryhard.out`
Time limit:	2 seconds
Memory limit:	256 megabytes

Positive integer A is a divisor of an integer B if B can be divided by A without remainder. For example, 15 has 4 divisors: $1, 3, 5, 15$. Your task is to determine whether the number of divisors of a given integer is even or odd.

Input

The first line of the input file contains one integer number N ($1 \le N \le 100$). The second line contains N integers X_i ($1 \le X_i \le 10^{18}$). Numbers in the line are space-separated.

Output

The only line of the output file must contain N numbers separated by spaces. The i-th number should be 0, if the number of divisors of X_i is even, and 1 otherwise.

Example

veryhard.in	veryhard.out
2 4 5	1 0

Problem B. Robots and Oil Transportation System

Input file:	`robots.in`
Output file:	`robots.out`
Time limit:	2 seconds
Memory limit:	256 megabytes

In some country there is a huge oil transportation system. The system consists of control stations. Some pairs of stations are connected by pipes.

System is connected, so there is a path via pipes from each control station to each other control station. But there are some pipes such that if they will broke, the system will become not connected. These pipes are called magistral. It's guaranteed that there is at least one magistral pipe.

The company has bought two robots for maintaining magistral pipes. After receiving a command, some magistral pipe will be chosen and both robots start to move to different ends of this pipe. Arrival time is the time needed for robot that will arrive to its destination later.

Robots are moving along pipes. They pass one unit of length in one unit of time. At the moment of receiving the command robots are located on the given control stations (may be different).

Write a program that will determine a magistral pipe such that robots' arrival time is minimal possible.

Input

First line of the input file contains two integer numbers N ($2 \le N \le 100\,000$) and M ($2 \le M \le 100\,000$) — number of control stations and pipes.

Each of the next M lines contains three integers: numbers of two control stations and length of the pipe connecting them. Length of the pipe does not exceed 1000.

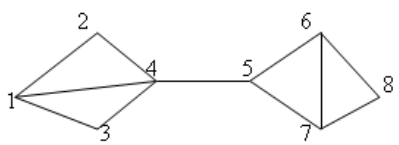
$(M+2)$-th line contains two integers: numbers of the control stations where robots are located at the moment of receiving command.

Output

Output the only one number — minimal time robots' arrival time to a magistral pipe.

Example

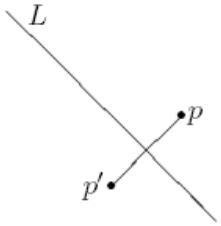
robots.in	robots.out
8 11 1 2 3 1 3 5 1 4 8 3 4 4 2 4 3 4 5 2 5 6 9 5 7 3 6 7 4 6 8 5 7 8 6 3 6	7



Problem C. Symmetry

Input file:	`symmetry.in`
Output file:	`symmetry.out`
Time limit:	2 seconds
Memory limit:	256 megabytes

Given a point p and a line L on the plane, the reflection of p against L is a point p' such that the segment pp' is perpendicular to L and its middle point is on L. If p is on L, then $p' = p$.



Given a set of points on the plane, the axis of symmetry is a line on the plane such that the reflection of any point of the set against that line gives a point from this set. In this problem you are given a set of points on the plane, and you must decide whether there exists at least one axis of symmetry or not.

Because of using real numbers, you may assume that the axis of symmetry is a line L such that for any point of the set p its reflection p' against the line L lies within 10^{-6} distance of some point of the set.

Input

The input contains several test cases. The first line of each input file contains an integer K ($1 \le K \le 10$) indicating the number of test cases. The first line of each test case contains an integer N indicating the number of points in the set ($1 \le N \le 100\,000$). Each of the next N lines describes a single point of the set using two real numbers X and Y separated by a single space ($-10\,000 \le X, Y \le 10\,000$), these numbers represent the coordinates of the point on the plane. You may assume that no two points of each test case have the same location. It is guaranteed that the total sum of values of N for all test cases does not exceed 200 000.

Output

For each test case output a single line `YES` if there exists at least one axis of symmetry for the provided set of points, or `NO` otherwise.

Example

symmetry.in	symmetry.out
3 4 -10 0 10 0 10 10 10 -10 4 -10 0 10 1 10 10.5 10 -10.5 6 -1000 30 -100 20 -10 10 1000 30 100 20 10 10	YES NO YES