Problem A. Rise of Shadows

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Rumor has it that shadows rise in a prime leap year. A prime leap year is a leap year, and the year number is also a prime number.

Toilet-Ares has recently learned the definitions of leap year and prime number. Given a specific year number, he wants to know if it is a prime leap year.

Recall that

- every year that is exactly divisible by four is a leap year, except for years that are exactly divisible by 100, but these centurial years are leap years if they are exactly divisible by 400;
- a prime number (or a prime) is a positive integer greater than one that is not a product of two smaller positive integers.

Input

The first line contains only one integer T $(1 \le T \le 1000)$, denoting the number of test cases.

Each case consists of only an integer a $(2 \le a \le 10^9)$ in one line, representing the number of the year Toilet-Ares wants to know about.

Output

For each case, if the year is a prime leap year, print `yes` in one line, otherwise print `no` in one line.

Example

standard input	standard output
1 2020	no

Problem B. Kobolds and Catacombs

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

Kobolds are rat-like, candle-loving cave folk, digging deep beneath the surface for millennia. Today, they gather together in a queue to explore yet another tunnel in their catacombs!

But just before the glorious movement initiates, they have to arrange themselves in non-descending heights. The shortest is always the leader digging small holes, and the followers swelling it.

The kobolds are hyperactive; they like to move here and there. To make the arrangement easier, they decide to group themselves into consecutive groups first, then reorder in each group.

What's the maximum number of consecutive groups they can be partitioned into, such that after reordering the kobolds in each group in non-descending order, the entire queue is non-descending?

For example, given a queue of kobolds of heights `[1, 3, 2, 7, 4]`, we can group them into three consecutive groups (`[1] [3, 2] [7, 4]`), such that after reordering each group, the entire queue can be non-descending.

Input

The first line of the input contains a single integer n ($1 \le n \le 10^6$), denoting the number of kobolds.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$), representing the heights of the kobolds in the queue.

Output

Print a single integer, denoting the maximum number of groups.

Example

standard input	standard output
5 1 3 2 7 4	3

Problem C. The Witchwood

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

Shenyang's night fair culture is developed very well. Every time Bob comes to Shenyang, he will definitely go to a night fair called *The Witchwood*. There are n snack stalls in *The Witchwood*, the ith of which gives him a_i pleasure.

Bob's stomach allows him to eat k snack stalls at most. So Bob wants to know the maximum pleasure he can get after visiting the night market.

Input

The first line of input contains two integers n ($1 \le n \le 1000$) and k ($1 \le k \le n$), indicating the number of snack stalls and the capacity of Bob's stomach.

The second line of input contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$), the ith of which indicates the pleasure of the ith snack stall.

Output

Print one integer denoting the maximum pleasure Bob can get.

Example

standard input	standard output
5 2 9 8 10 2 4	19

Problem D. Closest Pair of Segments

Input file:	*standard input*
Output file:	*standard output*
Time limit:	12 seconds
Memory limit:	512 mebibytes

The closest pair of points problem is a well-known problem in computational geometry. In this problem, you are given n points on the Euclidean plane, and you need to find a pair of points with the smallest distance between them.

Now, Claris, the brilliant one who has participated in programming contests for several years, is trying to solve a harder problem named the closest pair of segments problem, which also has a quite simple description as above.

However, the problem seems too hard, even for Claris, and she is asking you for help.

Now n segments are lying on the Euclidean plane. You have to pick two different segments, and then pick a point on each of them. Do it in such a way that the distance between these two points is the minimum possible.

For simplicity, no two given segments share a common point. Also, you don't need to show her the two points: just find the minimum possible distance between them instead.

Input

The input contains several test cases, and the first line contains a single integer T ($1 \le T \le 100$): the number of test cases.

For each test case, the first line contains one integer n ($2 \le n \le 100\,000$), which is the number of segments on the Euclidean plane.

The following n lines describe all the segments lying on the Euclidean plane. The i-th of these lines contains four integers, x_1, y_1, x_2, and y_2, describing a segment that connects (x_1, y_1) and (x_2, y_2), where $-10^9 \le x_1, y_1, x_2, y_2 \le 10^9$.

It is guaranteed that, in each test case, the two endpoints of each segment do not coincide, and no two segments share a common point. It is also guaranteed that the sum of n in all test cases does not exceed $100\,000$.

Output

For each test case, output a line containing a single real number: the answer to the closest pair of segments problem with an absolute or relative error of at most 10^{-6}.

Precisely speaking, assume that your answer is a and and the jury's answer is b. Your answer will be considered correct if and only if $\frac{|a-b|}{\max\{1,|b|\}} \le 10^{-6}$.

Example

standard input	*standard output*
2 2 0 1 1 2 1 1 2 0 2 0 1 1 2 2 2 3 1	0.707106781185 1.000000000001