Problem A. The Walking Adam

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Dark yellow`

Adam has just started learning how to walk (with some help from his brother Omar), and he falls down a lot. In order to balance himself, he raises his hands up in the air (that's a true story), and once he puts his hands down, he falls.

You are given a string, each character represents a step he walks, if that character is 'U' that means his hands are up in this step, if this character is 'D' that means his hands are down and he fell down in this step. Your task is to count how many steps he will walk before falling down for the first time.

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer $\boldsymbol{T}$ representing the number of test cases.

Each test case will consist of a single line, containing a non-empty string of at most 100 characters, and each character is either 'U' or 'D'. The characters from left to right represent Adam's steps in the order he walks them.

Output

For each test case print a single line containing the number of steps that Adam will walk before falling down, or the length of the string if he won't fall down.

Example

standard input	standard output
3 UUUDU DDD UU	3 0 2

Note

In the first test case, he falls down after 3 steps.

In the second test case, he falls down before making any steps.

In the third test case, he doesn't fall down at all.

Problem B. Endure Invests in Apples

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Light green`

Endure Capital is an early stage investment fund headed by entrepreneurs. Unlike many venture capital firms, Endure prides itself on being founder-focused. It believes in the people behind the idea, and works with them through challenges, hurdles and objectives.

This year, Endure decided to invest in a new technology used in Apple factories. This technology processes apples, and outputs some data in numbers. One of the very first problems they faced, is bucketing apples into 3 buckets based on their size.

Because everyone's taste is different, some people prefer small apples, others prefer medium ones and the rest prefer large apples. Your task is to contribute to this technology by counting the number of small, medium and large apples produced by the apple factory.

You will be given N apples, each with a size S_i. Apples are considered small if their size is less than or equal to X. They are considered medium if their size is less than or equal to Y and they're not small. Otherwise, they are considered large apples.

In other words:

- Small apple sizes $\subset [1, X]$
- Medium apple sizes $\subset]X, Y]$
- Large apple sizes $\subset]Y, \infty[$

Given N, their sizes, X and Y, calculate A, B, C; the number of small, medium and large apples, respectively.

Input

The first line of input has a single integer N, $1 \leq N \leq 10^5$ – the number of apples.

The second line has 2 space-separated integers X and Y, $1 \leq X < Y \leq 10^5$.

The third line has N space-separated integers, each representing the size S_i of the i_{th} apple. $1 \leq S_i \leq 10^5$.

Output

Output 3 integers; A, B and C, each representing the number of small, medium and large apples, respectively.

Examples

standard input	standard output
4 17 21 2 3 25 24	2 0 2
5 80 98 100 99 100 69 100	1 0 4

Problem C. Inside the circle

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Turquoise`

You are given a circle C, a set of "obstacle" circles $O_1, ..., O_n$ and two points S and E.

Count the minimum number of obstacles you will have to go through by taking a valid path from S to E.

A path is valid if all the points on the path lie inside C.

The following properties are guaranteed:

- The points S and E are inside C.
- All obstacles are either inside C or intersect with C.
- Each obstacle will be intersecting with at most two other circles (can be C or any of the obstacles).
- No obstacle will contain another obstacle.

Input

First line will be the number of test cases T, in each test case you will be given the following:

First line contains four integers describing the coordinates of the points S and E. ($-10^9 \leq S_x, S_y, E_x, E_y \leq 10^9$)

Second line contains an integer n, the total number of circles (the big circle + (n-1) obstacles). ($1 \leq n \leq 10^4$)

Followed by n lines, each describing a circle (first C, followed by the n-1 obstacles).

Each line contains three integers r, x, and y, describing the radius of the circle and coordinates of the center.

All integers on the same line are separated by spaces.

Output

For each test case output one line containing the number of obstacles you have to go through to pass from S to E.

Example

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

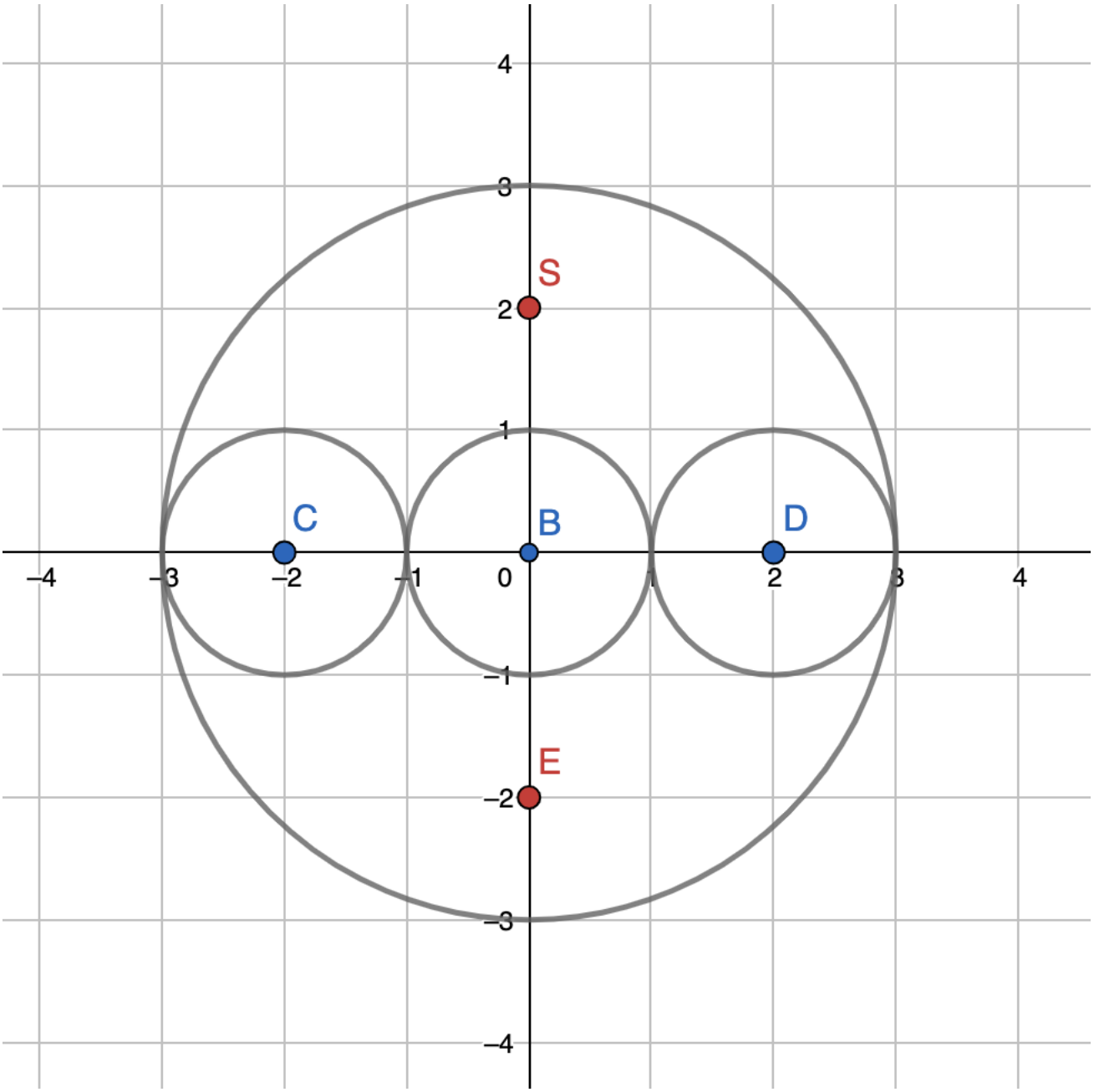


Рис. 1: The valid path from S to E can pass through at least one obstacle.

Problem D. Holidays for the Volunteers

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Light blue`

The ACPC volunteers have been working a lot to make the ACPC happen. They have not taken a single vacation in a long time. As the ACPC time approached, they started daydreaming about their vacations. One of the dreams was the following:

You work for a company in the outer space which gives you a salary of A dollars per day, and if you work for K days in a row, you will get a bonus of B dollars.

As years in the outer space are not the same on Earth, the year at this company consists of N days (numbered from 1 to N). This year, there are M holidays, which initially you can't work on.

However, you can choose at most Q holidays and work on them.

You want to buy a fancy car that costs S dollars, so you want to choose Q holidays in an optimal way to earn S or more dollars as soon as possible.

In other words, you want to know the smallest index of the day on which you would have earned S or more dollars. If that day does not exist, print -1.

Note that different blocks of K days do not overlap, which means that if you work for $2 \times K$ days, you will get $2 \times B$ dollars and not $(K + 1) \times B$ dollars as bonus.

Input

You will be given T test cases.

Each test case has:

- The first line has 7 space-separated integers:
  - N – the number of days in a year $1 \leq N \leq 10^9$.
  - M – number of holidays, $1 \leq M \leq 5\ 000$.
  - K – number of consecutive days in a row. $1 \leq K \leq 10^8$.
  - Q – number of holidays you can choose to work on. $1 \leq Q \leq M \leq 5\ 000$
  - A – your salary per day. $1 \leq A \leq 10^9$.
  - B – the bonus, $1 \leq B \leq 10^9$.
  - S – the price of the fancy car, $1 \leq S \leq 10^{18}$.
- The second line has M space separated integers $H_1, H_2, H_3 ... H_M$, where H_i is the index of the i^{th} holiday in the year. It's guaranteed that these indices are distinct. $(1 \leq h_i \leq N)$.

It is guaranteed that the sum of each of N, M, K, Q, A, B, S over all test cases will not exceed its maximum value.

Output

Output the smallest index of the day at which you can earn S or more dollars.

Print -1 if you can't earn S dollars in any way.

Example

standard input	standard output
1 10 2 3 1 1 2 6 2 4	5

Problem E. Arc Measure

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Orange`

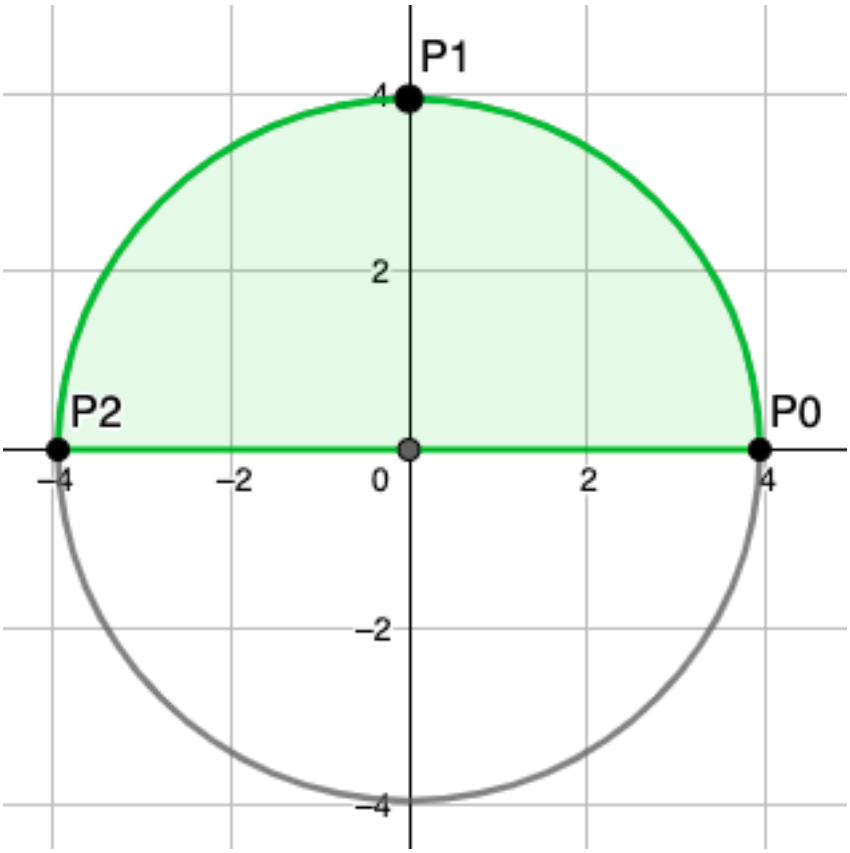
Coach Academy teaches students to think, create and code. One day the instructor gave the students this interesting problem.

If you were one of these students, would you be able to solve it?

Given a set of N points on the circumference of a $2D$ circle centered at the origin, where the i^{th} point is located at a_i degrees from the $+ve$ x-axis.

You need to find the minimum integer arc measure M, such that you can covered all the points using no more than K arcs of measure M. The arc covering must be contiguous with no gaps.

It is ok for two or more arcs to overlap.



Input

First line will be the number of test cases T, in each test case you will be given the following:

First line will have 2 integers N and K separated by a space. $(1 \leq N, K \leq 360)$

Followed by a line containing N integers separated by a space, each integer $(0 \leq A_i \leq 359)$

Output

For each test case, output one number in a line by itself, M the minimum arc measure.

Example

standard input	standard output
1 3 1 0 90 180	180

Note

An arc measure is the number of degrees it can cover from the circumference. For example, an arc measure of 90 covers quarter of the circle.