International Collegiate Programming Contest
The 2021 Ethiopian Collegiate Programming Contest
Ethiopia
September 2021

The International Collegiate Programming Contest
Sponsored by ICPC Foundation

The 2021 Ethiopian Collegiate Programming Contest

(Contest Problems)

Ethiopia
September 2021

Problem A. Easy problem

Input file:	`qual.in`
Output file:	`standard output`
Balloon Color:	`Pink`

There are n students who want to participate in a regional contest this year. Eslam wants to know how many teams could be formed from that number of students knowing that one team consists of 3 different students and any student can belong to at most 1 team.

Can you help Eslam calculate the maximum number of teams that can be formed from this number of students?

Input

The first line contains a single integer T, denoting the number of test cases.

Each test case contains one integer n ($1 \leq n \leq 10^3$), the number of students.

Output

Print a single line containing the maximum number of teams.

Example

qual.in	standard output
3 6 13 5	2 4 1

Problem B. Trains

Input file:	`trains.in`
Output file:	`standard output`
Balloon Color:	`Orange`

You are given the arrival and leaving times of n trains.

There is an infinite number of railways on which the train can stop.

The i^{th} train arrive at the moment l_i and leaves at r_i.

Every train will stop at the first empty railway.

So, if the first railway was empty, it will stop on it, if it was not empty, it will stop on the second if it was empty and so on.

You are going to find out the railway the i^{th} train will stop on when it arrives.

Input

The first line of the input contain one integer T ($1 \leq T \leq 100$) denoting the number of testcases.

Each testcase starts with number n ($1 \leq n \leq 10^5$).

Then n lines follows, each line consist of two integers l_i, r_i ($1 \leq l_i < r_i \leq 10^9$). the arrival and leaving times for the i^{th} train.

Output

Each test case should contain n positive integers, representing the railway that the i^{th} train will stop on.

Example

trains.in	standard output
1 3 1 4 2 3 5 6	1 2 1

Problem C. Polygon is shifted

Input file:	`polygon.in`
Output file:	`standard output`
Balloon Color:	`Yellow`

Abdulqader hates polygons so much. Whenever he sees a regular polygon, he tries to shift it into circle by covering all its edges till it becomes a circle. He is now old and got curious about the total area enclosed between the circle and the regular polygon.

Can you help him?

Input

The first line of input contains one integer T ($1 \leq T \leq 100$) denoting the number of testcases.

Each testcase consists of two integers n , l ($3 \leq n \leq 359$) ($1 \leq l \leq 10^9$) denoting the number of edges of the polygon and the length of every side in the regular polygon.

Output

Print the area enclosed between the circle and the regular polygon.

Example

polygon.in	standard output
2 3 6 4 4	6.522197 9.132741

Problem D. Math or Geometry?

Input file:	`matheo.in`
Output file:	`standard output`
Balloon Color:	`Light Green`

You are given the radius of a circle and two lines that intersect inside the circle cutting the circle into four parts.

The center of the circle is (0,0).

You should output the percentage of the biggest part over the total area of the circle.

Input

The first line of the input contain one integer T ($1 \leq T \leq 100$) denoting the number of testcases.

Each testcase consist of three lines:

- The first line contains one integer r denoting the radius of the circle.
- The second line contains four integers x_1, y_1, x_2, y_2 denoting the coordinates of the points of the first line.
- The third line contains four integers x_3, y_3, x_4, y_4 denoting the coordinates of the points of the second line.

All the integers are less than or equal to 10^5 in absolute terms.

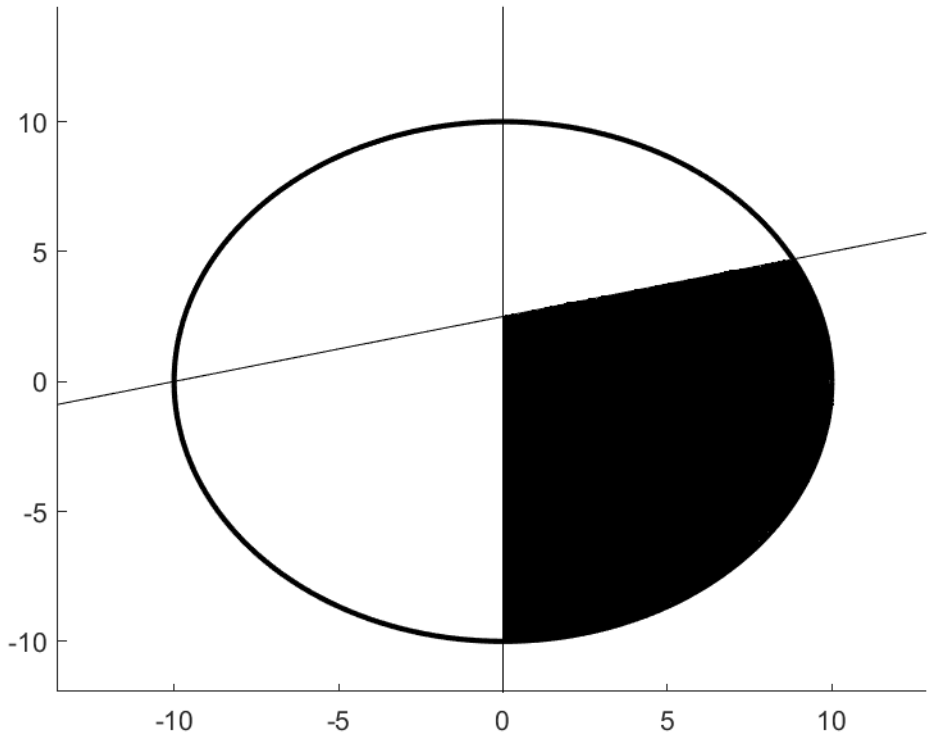
Output

Print the percentage of the biggest sector over the total area of the circle.

Example

matheo.in	standard output
2 10 0 10 0 0 -10 0 10 5 10 -10 0 10 0 0 -10 0 10	36.3087 25.0000

Note



The first testcase is described in the figure above.

In the second testcase the two lines cut the circle into four equal parts, so the answer is 25 percent.

It is guaranteed that the intersection point of the two given lines is inside the circle.

Problem E. Shop Questions

Input file:	questions.in
Output file:	standard output
Balloon Color:	Gold

Valentino is a manager in a new shop in the mall. In the shop, there are n items placed next to each other on a shelf, numbered from 1 to n, and each item costs c_i coins.

There will be q number of customers visiting the store. Each customer will ask a question in the form of: How many items whose number is between 1 and k have a cost that is between l and r coins?

Valentino wants to answers the customers' questions quickly, so he has asked you to write a program to help him in answering these questions.

Input

The first line contains two integers n and q $(1 \leq n \leq 10^5)$, $(1 \leq q \leq 10^5)$.

The following line contains n integers, $c_1, c_2, ..., c_n (1 \leq c_i \leq 10^9)$.

Then follows q lines, each line contains three integer k, l, r $(1 \leq k \leq n)$, $(1 \leq l \leq r \leq 10^9)$

Output

Print the answer of each question on a single line.

Example

questions.in	standard output
5 3 10 150 3 9 7 2 1 15 4 2 9 5 11 12	1 2 0

Problem F. Unsorted array

Input file:	`unsorted.in`
Output file:	`standard output`
Balloon Color:	`Blue`

We have an array A of N integer elements, and we want to make it sorted in a non-ascending order using the following operation any number (possible zero) of times.

We can take a subsequence of elements from the array and decrease each element of it by $(X > 0)$.

We want to know the minimum sum of X, if we choose the applied operations optimally.

A subsequence of a given sequence is a sequence that can be derived from the given sequence by deleting some or no elements without changing the order of the remaining elements.

A non-ascending array is an array where for each $(1 \leq i \leq N-1)$, $A[i] \leq A[i+1]$.

Input

The input file starts with T, the number of testcases.

Each testcase starts with $(1 \leq N \leq 10^5)$, donating the size of the array and followed by N separated integers $(-10^9 \leq A[i] \leq 10^9)$.

Output

For each testcase, output the sum of X needed.

Example

unsorted.in	standard output
1 6 5 3 1 2 10 5	5

Note

In the first testcase, we will do the following:

Choose the subsequence that contains the first, second and fifth elements, and decrease each one of them by 2 and return them to the original array to become the following:

3 1 1 2 8 5

Now, choose the first and the fifth elements and decrease them by 2:

1 1 1 2 6 5

Finally, decrease the fifth elements by 1:

1 1 1 2 5 5

So, the answer becomes: X = 2 + 2 + 1 = 5.

Problem G. Long Travel

Input file:	`long.in`
Output file:	`standard output`
Balloon Color:	`Violet`

You are given the coordinates of N points, you are at the point $(0, 0)$, and you want to go to the point $(V, 0)$.

You want to pass by all those points in an ascending order according to their X-coordinates.

You can remove at most K points from the set of points.

We want you to calculate the length of the shortest path to your destination if you deleted at most K points optimally.

Input

The first line of input contains one integer T $(1 \leq T \leq 100)$ denoting the number of testcases.

Each testcase starts with three integers n, k and v $(1 \leq K \leq N \leq 1000)$ $(1 \leq V \leq 10^9)$ denoting the number of points, the maximum number of points that you can remove and the X-coordinate of the end point.

Then N lines follows, each line consists of two integers X, Y denoting the X-coordinate and Y-coordinate of the given point $(1 \leq X \leq 10^9)$ $(-10^9 \leq Y \leq 10^9)$.

Output

You should output the minimum distance from $(0, 0)$ to $(V, 0)$ if you can remove at most K points.

Example

long.in	standard output
2 2 0 1002 0 1000 1000 0 2 1 1002 0 1000 1000 0	2416.213562 1002.000000

Note

It is guaranteed that V is greater than the most-right point.

Problem H. Nihad hates setprecision

Input file:	`precesion.in`
Output file:	`standard output`
Balloon Color:	`Navy`

Nihad is a very smart student, he studied for the programming exam real hard.

The Exam was really easy for Nihad, so when anyone asks him about any question he says the correct answer right away.

Ali is a friend of Nihad, he finds pleasure when he spots a wrong answer in the exam paper.

So he came to Nihad and asked how did he print the average of the marks as the question in the exam said.

Nihad peacefully showed him his solution which is:

std::cout « average « endl;

Ali laughed out loud and Nihad didn't understand, so Ali told Nihad that he should print the average with only 2 decimal places, and he taught him about the setprecision function in C++ .

Nihad got very Angry because he doesn't like to lose marks over silly things, and he wanted to make this function look useless.

So, he went to the doctor and suggested a new problem:

You are given an integer $x > 0$ and an integer d.

You will count how many integer numbers y , $y > 0$ such that x/y have d decimal places at most.

For example, if $x = 2$ and $d = 1$ the answer will be 6.

$2/2 = 1$, there aren't any decimal places so it's acceptable.

$2/3 = 0.\bar{6}$ there is an infinite number of decimal places so it's rejected.

and so on.

Examples:

x = 7, d = 1

7 / 1 = 7

7 / 2 = 3.5

7 / 5 = 1.4

7 / 7 = 1

7 / 10 = 0.7

7 / 14 = 0.5

7 / 35 = 0.2

7 / 70 = 0.1

All of those are acceptable because none of them have more than one decimal place.

Input

The first line contains a single integer $T \leq 10000$ denoting the number of test cases

Each test case has 2 integers x and d $1 \leq x \leq 10^9$, $0 \leq d \leq 9$.

Output

Output how many integer numbers y , $y > 0$ such that x/y have at most d decimal places.

Example

precesion.in	standard output
3 65 0 6 2 23 2	4 24 18

Problem I. Desk

Input file:	`desk.in`
Output file:	`standard output`
Balloon Color:	`White`

Valentino works as a teacher in a strange school. In each class, there are n students and they all share one big desk. The student sitting in position i on the desk has a unique student ID a_i.

Initially, the students are seated randomly but there are some naughty students making noises, so Valentino wants to expel them from the class. He has a list of the IDs of the naughty students that he wants to expel. Valentino will move down the list and expel these bad students one by one.

To expel a naughty student, Valentino does the following steps:

First, he removes each of the $i - 1$ students that are sitting before the naughty student. Then, he expels the naughty student and returns all the students that he removed. **Each operation takes a second** (removing a student, expelling a student, returning a student).

But Valentino has a powerful option that can help him save time: when returning the students that he removed, he can return them **in any order**.

Now Valentino wants to know the minimum time needed to expel all the bad students from his class in order of their appearance on the list.

Input

The first line contains one integer t ($1 \le t \le 100$) — the number of test cases.

The first line contains two integers n and l ($1 \le l \le n \le 10^5$) − the number of students sitting at a desk and the number of students to be expelled.

The second line contains n integers a_1, a_2, ..., a_n ($1 \le a_i \le n$) − The unique ID of the students in order of seating.

The third line contains l integers b_1, b_2, ..., b_l ($1 \le b_i \le n$) − The IDs of the students to be expelled.

Output

For each test case, print one integer − the minimum number of seconds which Valentino has to spend to expel the students in the list in order of their appearance on the list.

Example

desk.in	standard output
2 3 3 2 3 1 3 1 2 3 1 2 1 3 2	7 1

Problem J. Eisneria's Canal

Input file:	`tunnel.in`
Output file:	`standard output`
Balloon Color:	`Silver`

A few months ago, Eisneria's Canal was blocked by a huge container ship. The engineers of the Canal decided to expand it.

The Canal is represented as $n \times m$ grid.

if $grid_{i,j}s = 1$ this part is water, otherwise it is a land.

Unfortunately, they can only dig one cell to become water.

They want the total adjacent water cells to be as large as possible, Can you help them?

Input

the first line of the input contains one integer $(1 \leq T \leq 100)$ denoting the number of testcases.

Each testcase start with two integers $n, m(1 \leq n, m \leq 10^3)$. denoting the number of rows and columns respectively.

Then, n lines follow, each line consists of m integers, if $grid_{i,j} = 1$ this part is water, otherwise it is a land.

Output

Print the maximum number of adjacent cells after doing the operation.

Example

tunnel.in	standard output
2 3 3 1 1 0 0 0 0 1 1 1 3 6 1 1 0 0 0 0 1 1 0 1 1 1 1 1 0 1 1 1	6 13

Problem K. The Beautiful gcd

Input file:	gcdisk.in
Output file:	standard output
Balloon Color:	Black

There is an array of n integers $a_1, a_2, \ldots, a_n$ and there is an integer k. We call a subarray of this array **beautiful** if and only if the greatest common divisor of all the numbers in this subarray is equal to **exactly** k.

Given n, k and the array a, you need to find the number of different beautiful subarrays.

Input

The first line of input contains a single integer T ($1 \leq T \leq 50$) – the number of test cases.

Each test case consists of two lines. The first line contains two integers n and k ($1 \leq n \leq 10^5$) ($1 \leq k \leq 10^9$).

The second line contains n space separated integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 10^{18}$).

It's guaranteed that the sum of n over all test cases doesn't exceed 10^6.

Output

For each test case, print a line with a single integer that represents the answer to the problem.

Example

gcdisk.in	standard output
2 4 2 1 3 4 2 6 7 8 2 4 1 3 9	2 0

Problem L. The jumping man

Input file:	`jump.in`
Output file:	`standard output`
Balloon Color:	`Dark Green`

Bob lives between the mountains. These mountains form a 2D-array A with dimensions $n * m$. Wach cell in this array $A_{i,j}$ represents the height of the mountain (i, j) in meters. Bob is initially at mountain $(Xstart, Ystart)$ and he wants to go to the mountain $(Xgoal, Ygoal)$. Bob has an ability of jumping equal to k meters.

If Bob is currently at cell (x, y) he can move to one of the following cells: $(x + 1, y), (x, y + 1), (x - 1, y), (x, y - 1)$ and he can do a move if and only if: (the height of the mountain he is going to) is less than (the height of the mountain he is standing on right now + k).

Can you find the minimum k (Bob's ability to jump) possible which allows Bob to go from $(Xstart, Ystart)$ to $(Xgoal, Ygoal)$.

Input

The first line contains a single integer T $(1 \leq T \leq 50)$ − the number of test cases. The description of the test cases follows.

The first line of each testcase contains two integers $n, m(1 \leq n, m \leq 300)$ denotes the number of rows and the number of columns in the array A, respectively.

The second line of each testcase contains four integers: $Xstart$, $Ystart$, $Xgoal$, $Ygoal$ $(1 \leq Xstart, Xgoal \leq n)$ $(1 \leq Ystart, Ygoal \leq m)$.

Each line of the following n lines in each testcase contains m integers, where the j_{th} integer in the i_{th} line $A_{ij}(-10^9 \leq A_{ij} \leq 10^9)$ denotes the height of the mountain (i, j).

Output

Print an integer that denotes the minimum k (Bob's ability to jump).

Example

jump.in	standard output
1 4 4 1 1 4 3 -1 2 -3 5 1 0 -1 -2 1 3 1 3 2 4 8 5	3