Criterion 2020 Round 7

Furqan Software

https://toph.co/c/criterion-2020-round-7

Schedule

The contest will run for **2h30m0s**.

Authors

The authors of this contest are Arcturus, curly_braces, fire_tornado, Hasinur_, msabeer, pranonraian, shajia, TarifEzaz, and Tashdid_trdb.

Rules

This contest is formatted as per the official rules of ICPC Regional Programming Contests.

You can use Bash 5.0, Brainf*ck, C# Mono 6.0, C++11 GCC 7.4, C++14 GCC 8.3, C++17 GCC 9.2, C11 GCC 9.2, Erlang 22.3, Free Pascal 3.0, Go 1.13, Haskell 8.6, Java 1.8, Kotlin 1.1, Node.js 10.16, Perl 5.30, PHP 7.2, PyPy 7.1 (2.7), PyPy 7.1 (3.6), Python 2.7, Python 3.7, and Ruby 2.6 in this contest.

Be fair, be honest. Plagiarism will result in disqualification. Judges' decisions will be final.

Notes

There are 6 challenges in this contest.

Please make sure this booklet contains all of the pages.

If you find any discrepencies between the printed copy and the problem statements in Toph Arena, please rely on the later.

A. Assignment Assignment Assignment

After the season of online classes, now it's time for evaluation. A teacher wants to evaluate the students by giving them assignments. The teacher thinks that the best way to evaluate the students is to make them present on some topics in front of the whole class. So, he has asked you to choose the topics on which each student will present.

For convenience, the teacher has given you a list of probable topics from each chapter. But, he also gave you some conditions while choosing these topics:

- There can be more than 1 topic from each chapter.
- Each student should take at least 1 topic.
- More than one student can choose topics from a single chapter.
- One student can choose one or more topics from only one chapter. But, one topic can be covered by at most one student.
- Some topics may not be chosen. But the number of topics that are not chosen should not be very large.

As Fafa is the class representative, the teacher has asked him to provide a list of topics that each student chooses.

Now, the problem is no student wants to choose more than one topic but in that case, a lot of topics remain unassigned. Then the teacher will become very angry. Fafa and others don't want to face this situation. So they have decided that each student will choose an equal number of topics. Also, to avoid any clash, the students have agreed to assign the topics randomly satisfying all the rules stated by the teacher.

Now, Fafa is busy with writing a program that will randomly assign exact K topics to each student. He has asked your help to find the optimal number K that denotes the number of topics that will be assigned to each student. In other words, he asked you to find the maximum number of topic each student will choose so that the number of topics that are not chosen is as minimum as possible. *It is guaranteed that each student will get at least one topic.*

Input

The first line of the input will contain a single integer $T(1 \leq T \leq 15)$ that denotes the number of test cases.

For each test case, the first line will contain 2 integers N and $S (1 \leq N, S \leq 100,000)$ where N denotes the number of chapters and S denotes the number of students in your class.

Next line will contain N space-separated integers $a_1, a_2, \ldots, a_n$ where $a_i (1 \leq a_i \leq 1,000,000,000)$ denotes the number of topics from i^{th} chapter. Here, $\sum_{i=1}^{n} A_i$ will always be greater than or equal to the number of students.

Output

For each case, print the case number and the maximum number of topics that can be assigned to each student satisfying all conditions.

Samples

Input	Output
3 3 8 2 5 1 4 2 4 3 2 1 1 4 8	Case 1: 1 Case 2: 3 Case 3: 2
For first test case, each student can take maximum 1 topics and there won't be any topic left For second test case, each student can take maximum 3 topics, one will take 3 topics from chapter 1 and another student will take 3 topics from chapter 2. Total 4 topics will be left. For third test case, 4 students each takes 2 topics and no topics will be left.	

B. Taofiq the House Prefect

Tomorrow is a very special day for Taofiq. He is participating in an intra school quiz contest. As he is one of the house prefects of his school, he has a lot of responsibilities. Furthermore, he needs to do well in the contest. Before any special event of his life, Taofiq takes some time for himself to contemplate. It helps him to think about himself and generate new ideas. Today is no exception. He has found a magic trick that will help his house team to do well in the contest. Now he is feeling sleepy and can not reach every member of his team. So he comes up with a different approach to propagate the magic trick to everyone.

Since everyone on his team has a mobile phone, he can propagate his message to each of them. But he decides to call only his friends from the team. Then every one of his friends will call their friends. In this way, he will communicate with the team members and propagate the news. There is another rule, everyone including him will call their friends with the ascending order of their roll numbers.

Suppose, the roll number of Taofiq is 8 and he has 3 friends to call whose roll numbers are 5, 2 and 9; then:

- Taofiq will call the friend with roll number 2,
- then he will call the friend roll number 5 and finally,
- he will call the friend with roll number 9.

Everyone on his team will follow the same rule. For simplicity, a person will try to call all of his friends sequentially. Also, a person will call his friends only after those who have received the call before him are finished calling. If someone has already got the messages from a friend, then he will only call his friends and will not pick up any other phone calls. So, if someone calls his friend but finds him unreachable, he will safely assume that the friend has already got the message.

Being a good house prefect, Taofiq already knows which team member is supposed to call which other members. Before the start of the contest, he will make some queries about the conversations to ensure that everything is done perfectly. In each query, he will mention two team members x and y and will ask whether they had a phone conversation or not.

As Taofiq is busy with other work, he needs your help. Write a program which will help Taofiq to make the queries easily.

Input

The first line contains an integer T ($1 \leq T \leq 10$), denoting the number of test cases. The first line of each case contains two integers n ($2 \leq n \leq 100000$) and m ($0 \leq m \leq 1000000$). Here, n represents the number of members in Taofiq's team and m represents the number of direct friendships among the team members. Then there will be m lines. Each of these m lines contain two integers x () and (), that represents friendship between team member x and team member y. After the m lines, there will be another integer Q ($1 \leq Q \leq 10$) that represents queries. Then there will be lines where each of the line represent a query in the format a ().

Taofiq will always have the roll number 0.

Output

In the output, for each case, print the case number on a line by itself. For each query, print **Yes** on a line, if there was a conversation between a and b, or print **No** otherwise.

Samples

Input	Output
2 9 10 0 3 0 8 1 0 1 3 1 5 2 4 3 2 4 7 5 2 2 6 5 1 3 0 6 4 5 4 7 6 7 6 8 0 1 1 2 1 3 1 5 2 4 5 4 5 3 4 3 1 1 0	Case 1: No No No Yes No Case 2: Yes

C. EP-Palindromes

You will be given an array of integers A. You have to count the number of $\textbf{subarrays}$ (a non-empty sequence of consecutive elements of an array) of A that are **EP-Palindrome**.

Definition of **EP-Palindrome**

> An array V is EP-Palindrome if $|V| \mod 2 = 0$ and rearranging the elements in V can form a palindrome. Here, $|V|$ denotes the number of elements in V, also known as the length of V. For example, let $V = [1, 2, 1, 2]$ where $|V| \mod 2 = 0$. We can get $V = [1, 2, 2, 1]$ or $V = [2, 1, 1, 2]$ by rearranging the elements of V. So, $V = [1, 2, 1, 2]$ is an $EP - Palindrome$.

Input

The first line of the input file will contain an integer $T (1 \leq T \leq 30)$ which denotes the number of test cases. Then there will be T test cases.

The first line of each test case will contain a single integer $N (1 \leq N \leq 10^5)$, denoting the size of the array A.

The second line of each test case will contain N integers $A[1], A[2], \ldots, A[N]$ which are the elements of the array. Here, the i^{th} integer denotes the element $A[i] (1 \leq A[i] \leq 2 \times 10^6)$.

Output

For each test case, print "$\textbf{Case X: Y}$" without the quotation symbol, where X is the case number and Y is the number of $EP - Palindromic$ subarrays of A.

Samples

Input	Output
2 3 1 1 1 4 1 2 1 2	Case 1: 2 Case 2: 1

In the sample input, the first test case contains an array $A[] = [1, 1, 1]$. Subarrays of A are $[1], [1], [1], [1, 1], [1, 1], [1, 1, 1]$

Now,

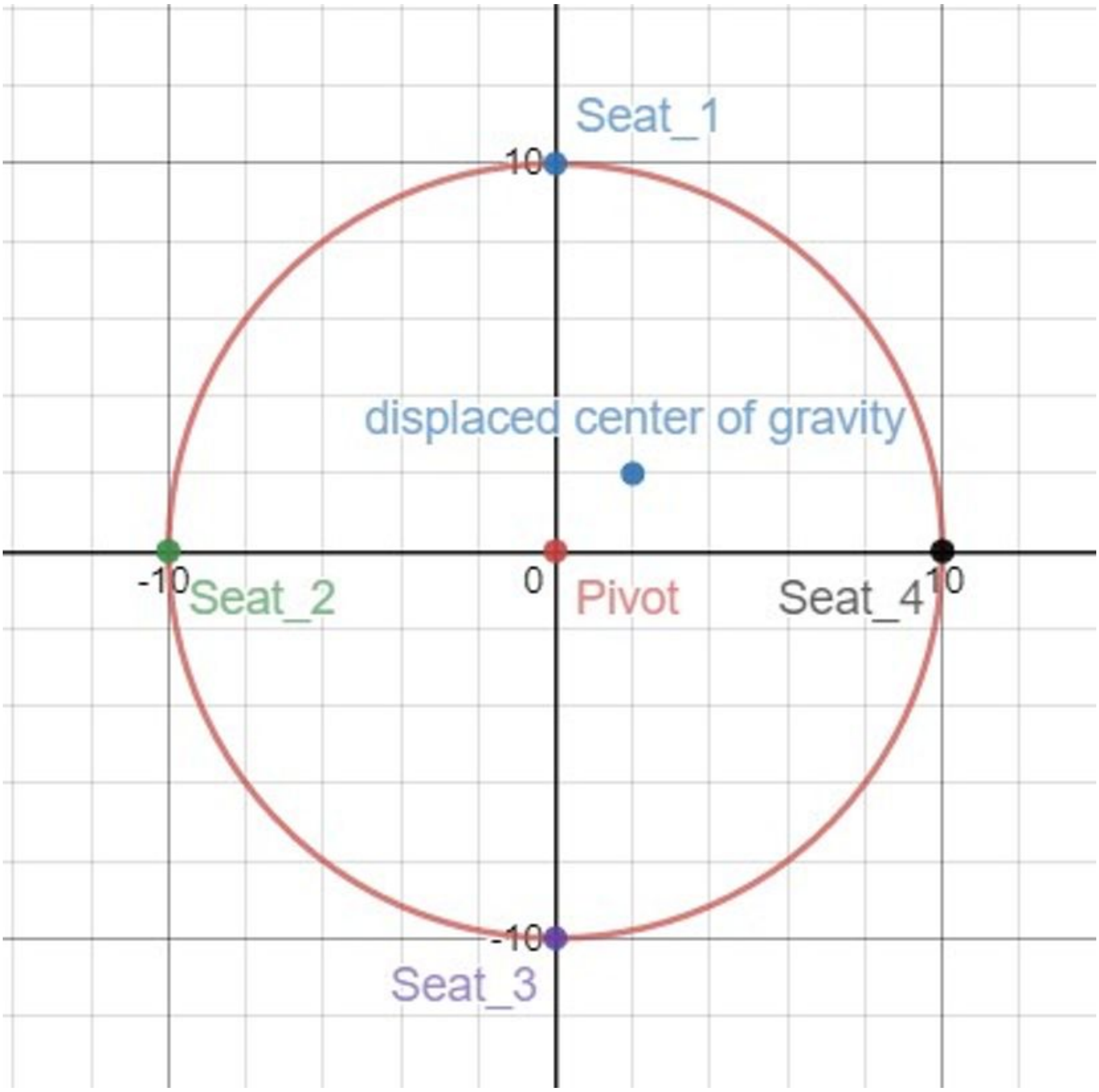
- $[1]$ has length $= 1$, $1 \mod 2 \neq 0$. So, it's not EP-Palindrome.
- $[1]$ has length $= 1$, $1 \mod 2 \neq 0$. So, it's not EP-Palindrome.
- $[1]$ has length $= 1$, $1 \mod 2 \neq 0$. So, it's not EP-Palindrome.
- $[1, 1]$ has length $= 2$, $2 \mod 2 = 0$. The subarray is a palindrome already. So, $\textbf{it\&\#39;s EP-Palindrome}$.
- $[1, 1]$ has length $= 2$, $2 \mod 2 = 0$. The subarray is a palindrome already. So, $\textbf{it\&\#39;s EP-Palindrome}$.
- $[1, 1, 1]$ has length $= 3$, $3 \mod 2 \neq 0$. So, it's not EP-Palindrome.

The array A has 2 subarrays in total which are $\textbf{EP-Palindromes}$

D. Merry-Go-Round

The Merry-Go-Round is an amusement ride, popular in rural Bangladesh. Today is your birthday, and to treat your N friends, you have decided to take them to an amusement park to enjoy the Merry-Go-Round ride. But this ride is special. Why so? Because it has infinitely many seats and one seat can be shared among infinitely many people.

A Merry-Go-Round cam be considered as a circle, balanced on a pivot at the center, that looks like this:



Consider the seats as individual points on the circle boundary/perimeter. When no seat is occupied, the center of the gravity remains at the pivot. So the system is perfectly horizontally balanced at the pivot. But, if some seats are occupied by one or more people, the center of gravity may be displaced from the center of the circle.

Another specialty of this Merry-Go-Round is its cost of riding. This cost is defined as the distance from the center of the circle to the displaced center of gravity. This distance is calculated as Pythagorean distance between two points i.e. $\sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2}$.

You are a genius and found out that if you can make your friends take seats in some particular ways, the cost could be reduced. Now you are given the weights of your N friends, $w_1, w_2, \dots, w_N$ (not necessarily all equal), and the radius of the circle R. Find out the **minimum** cost you need to pay for the ride.

N.B.: You should ignore the weight of the circle.

Input

The first line contains a single integer $T(1 \leq T \leq 10^4)$ - the number of test cases.

The first line of each test case contains 2 space-separated integer $N(1 \leq N \leq 10^5)$ and $R(0 \leq R \leq 10^9)$ - the number of friends and the radius of the circle.

The second line of each test case contains N space-separated integers $w_i(1 \leq w_i \leq 10^9)$.

It is guaranteed that the sum of N over all test cases does not exceed 10^6.

Output

For each test case, print the **minimum** cost you need to pay in separate lines. An absolute precision error of less than 10^{-6} will be ignored.

Samples

Input	Output
2 2 6 5 10 4 20 12 12 12 12	2.0000000000 0.0000000000

E. Infinite Shuffler

HeRock made an infinite array shuffler, which he thinks nobody can hack. The machine takes an infinite array, the operator gives it a key then it shuffles the array using that key. Now, he'll tell you the mechanism and the secret key. Can you hack the machine?

Shuffler Mechanism:

The shuffler has a secret key P of n distinct integers. **It is guaranteed that P has 0 as one of its elements and the elements are sorted in ascending order**. The shuffler has an array G of infinite length. G initially contains all integers from 0 to infinity. The 0^{th} element is 0, 1^{st} element is 1, 2^{nd} element is 2, and so on. It has another array S, which stores the shuffled array. Initially, S is empty.

The machine does the following operations an infinite number of times:

1. Take all the elements at index P_i (for each i in range $[0, n-1]$) from array G, put them in array S sequentially. All the elements taken away creates some empty spaces in array G.
2. Shrink the array G. That is if there is any empty position, fill it up with the element of the next non-empty position. Note that it'll create an empty position from where you have taken the element.

After repeating the operations an infinite number of times, the array S is the infinite shuffled array.

Examples

If the key was 0, 1 then the shuffled array would look like 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, …

If the key was 0, 2 then the shuffled array would look like 0, 2, 1, 4, 3, 6, 5, 8, 7, 10, …

If the key was 0, 3, 4 then the shuffled array would look like 0, 3, 4, 1, 6, 7, 2, 9, 10, 5, …

To test whether you have successfully hacked the machine or not, HeRock will ask you Q questions. On each question he will give you an integer x, You have to tell the element at index in array S. That means you need to print .

Input

Input starts with two integers n and .

Next line contains integers, the elements of $P(0 \le P_i \le 10^{12})$.

The next Q lines contains one integer each, the index $x(0 \le x \le 10^{12})$ for that query.

Output

For each query, print a line containing the answer to that query.

Samples

Input	Output
2 10 0 2 0 1 2 3 4 5 6 7 8 9	0 2 1 4 3 6 5 8 7 10

Input	Output
3 8 0 3 4 0 1 2 3 4 5 6 7	0 3 4 1 6 7 2 9

F. Back to Back

You will be given an array A of size N and Q queries.

The queries will be given in the following form:

- \textbf{1 L R}: You have to print the maximum sub-array sum from L^{th} index to R^{th} index (inclusive) of the array. That means, you need to print $MAX(\sum_{k=i}^{j} A[k])$ where, $L \leq i \leq j \leq R$. You can't select empty sub-array.
- \textbf{2 L R X}: You have to make X left circular shift of the values in the range $[L, R]$. For this problem, a left circular shift means removing the first element from the sub-array and inserting it at the end of the sub-array after changing its sign. For example, one left circular shift of a sub-array $[1, 2, 3, 4, 5]$ will be $[2, 3, 4, 5, -1]$.

Input

The first line will contain an integer T, the number of test cases. Then, the first line of each test case will contain an integer N, the size of the array A and the next line will contain the array. After that, there will be an integer Q, the number of queries followed by Q queries as explained above.

Constraints:

- $1 \leq T \leq 10^3$
- $1 \leq N \leq 10^5$
- $1 \leq Q \leq 10^5$
- $1 \leq L \leq R \leq N$
- $0 \leq X \leq R - L + 1$
- $|A_i| \leq 10^5$
- The sum of N will be $\leq 10^5$ over all the test cases
- The sum of Q will be $\leq 10^5$ over all the test cases

Output

For each test case, print "**Case #Y:**" (without quotes) in the first line where Y is the case number. Then for each query type 1, print the answer in a new line.

Samples

Input	Output
2 5 10 4 8 7 9 3 2 1 5 2 2 5 5 1 1 1 5 5 11 6 3 10 6 3 2 1 3 1 1 1 3 1 3 4	Case #1: 24 Case #2: 9 10