A	A Piece of Cake Score: 1 CPU: 1s Memory: 1200MB	ACM ICPC 2016 Dhaka Regional Preliminary Mock Contest (14/010/2016)

Okay, nothing to do with cakes really, whether you love them or not. This is to ascertain that this is a very very easy problem. So here is the deal, given an integer **n**, print all integers from 0 to **n** in increasing order. The limit of **n**? Who gives a damn when the problem is so easy? :D

Input

Input begins with a positive integer **T**, number of test cases. **T** lines follow, each containing a single integer, **n**.

Output

For each test case produce a line containing single space separated values from 0 to **n** in ascending order.

Sample Input	Sample Output
2 1 5	0 1 0 1 2 3 4 5

B	**Bus Stopage** Score: 1 CPU: 1s Memory: 1200MB	**ACM ICPC 2016 Dhaka Regional Preliminary Mock Contest (14/010/2016)**

Bus service of Dhaka is life saving for many people. It's the cheapest way of travel among the city, and it's even cheaper for the students due to half fare. So many students ride the bus to go to their institutions (schools, colleges or universities).

Government of Bangladesh wants this to be even cheaper for the students. That's why they are starting a special bus service for school students. This bus service will start from the start of the city and will go to the end (I know it seems weird, but try to imaging Dhaka as a linear city). There are N stoppages among this route and in every stoppage few gets on the bus and few people gets off. This number is always fixed because students don't change home or school that often. Now the authority has one problem and that is determining the size of the bus. The bus should be as big as that it can accommodate the students at every stoppage. Let's see an example. Suppose there are 4 stoppages. In first stoppage 5 people gets on the bus. In the second stoppage 4 people gets on and 2 gets off. So when leaving the second stoppage there are in total 7 people in the bus. In the third stoppage 3 people gets on and 4 gets off. So while leaving the third stoppage there are 6 people in the bus and the bus reaches to fourth and final stoppage with these 6 people. So the bus size must be 7 or higher because otherwise it won't be able to accommodate all the students from the second stoppage.

Given the information about the stoppages, you need to find the minimum size of the bus that's needed to accommodate all the students.

Input

Input will start with an integer, T, the number test cases. Each case will start with N (2<=N<=20000). Next there will be N-1 lines each with two integers A and B (0<=A,B<=20000), where in the ith line A is the number of people gets on bus and B is the number of people gets off at ith stoppage.

Output

Output a line for each case, "Case T: Size" where T is the case number and Size is the minimum required size of the bus.

Sample Input	**Sample Output**
2 4 5 0 4 2 3 4 2 4 0	Case 1: 7 Case 2: 4

C	Stick to Triangle Score: 1 CPU: 1s Memory: 1200MB	ACM ICPC 2016 Dhaka Regional Preliminary Mock Contest (14/010/2016)

You are given a stick of length N. You want to break it in three pieces such that it can form a triangle. In how many distinct triangles can you make? Two triangles are equal if all the side length is same when sorted in ascending order of length. So (1, 3, 2) is same to (3, 1, 2) because their side lengths are same is we sort them which is (1, 2, 3). But (1, 3, 4) is not same with (1, 2, 3). Suppose the lengths of three pieces are X, Y, Z respectively. Following constraints should be maintained:

1. X, Y, Z > 0.
2. X, Y, Z is an integer.
3. X + Y + Z = N

A triangle with zero area is considered a valid triangle. For example if N = 14, then there are 7 triangles: (1, 6, 7), (2, 5, 7), (2, 6, 6), (3, 4, 7), (3, 5, 6), (4, 4, 6), (4, 5, 5).

Input

First line will give you the number of test cases, T (T<=100). Then each line will have an integer N (0< N <= 300000).

Output

For each case print one line with the number of distinct triangles possible.

Sample Input	Sample Output
3 3 6 14	1 2 7

D	**Box Sorting** Score: 1 CPU: 1s Memory: 1200MB	**ACM ICPC 2016 Dhaka Regional Preliminary Mock Contest (14/010/2016)**

You are constantly getting threats from your mom to tidy up your desk. On top of your desk, there are lots and lots of cubic boxes of various sizes (You are a collector of boxes?!). Each box can contain one other box, which has size strictly smaller than itself. You have to put the boxes one inside of another in such a configuration that there are minimum number of boxes on the table.

Input

Input will start with the number of test cases, T ($T \leq 100$). Following that will be the number of boxes, N ($0 < N \leq 100,000$). The following N integers will denote the length of one side for each box, X ($0 < X \leq 2^{30}$).

Output

For each test case, print one line of output, "Case Y: Z". Where Y is the number of test case and Z is the minimum number of boxes on the table.

Sample Input	**Sample Output**
2 5 1 5 4 2 3 2 5 5	Case 1: 1 Case 2: 2

E	Large vs Small Score: 1 CPU: 1s Memory: 1200MB	ACM ICPC 2016 Dhaka Regional Preliminary Mock Contest (14/010/2016)

You will be given a list of numbers. After reading each number, you have to print the largest number you have still read divided by smallest number you have still read.

Output

For each X, print one line of output, “Case Y: Z”. Where Y is the position of X in input and Z is the result of the division rounded to two digit after decimal.

Sample Input	Sample Output
5 4 3 2 6 1	Case 1: 1.00 Case 2: 1.33 Case 3: 2.00 Case 4: 3.00 Case 5: 6.00

F	Process and Resources Score: 1 CPU: 1s Memory: 1200MB	ACM ICPC 2016 Dhaka Regional Preliminary Mock Contest (14/010/2016)

In operating systems, one of the major concerns is about process and resource management. Now you are to find whether a given process can be assigned to a given resource. In this problem, there will be two types of queries,

- Query type 1:
  - You will be given a process id (pid) and a resource id (rid). If the resource is already free (initially all resources are free), then output 'Y' and the resource will be hold by this process. Otherwise output 'N' (even if the resource is already held by this process).
- Query type 2:
  - You will be given a resource id. If the resource is already free output 'F', otherwise output the pid of the process which holds the resource and free the resource.

Input

The first line of the input file contains the number of test case T (1 <= T <= 25). Each test case starts with the number of query Q (1 <= Q <= 100000). Each of Q lines starts with the query type (either 1 or 2), then pid and rid (for type 1) or rid (for type 2) separated by spaces, where (0 < pid <= 2^63-1) and (0 < rid <= 2*10^7)

Output

For each query, output expected output described above. For clarification, see sample input/output.

Sample Input	Sample Output
1 4 1 100 1 1 101 1 2 1 1 101 1	Y N 100 Y