LightOJ

A Childhood Game

1 seconds 64 MB Medium LOJ-1020 Debug

Statement Submissions Statistics Tutorial

English

Alice and Bob are playing a game with marbles; you may have played this game in childhood. The game is playing by alternating turns. In each turn a player can take exactly one or two marbles.

Both Alice and Bob know the number of marbles initially. Now the game can be started by any one. But the winning condition depends on the player who starts it. If Alice starts first, then the player who takes the last marble loses the game. If Bob starts first, then the player who takes the last marble wins the game.

Now you are given the initial number of marbles and the name of the player who starts first. Then you have to find the winner of the game if both of them play optimally.

Input

Input starts with an integer T (≤ 10000), denoting the number of test cases.

Each case contains an integer n ($1 \le n < 2^{31}$) and the name of the player who starts first.

Output

For each case, print the case number and the name of the winning player.

Sample

Input	Output
3 1 Alice 2 Alice 3 Bob	Case 1: Bob Case 2: Alice Case 3: Alice

C++ 17 Editor Upload code

```
1  # Add some code
```

LightOJ

A New Function

2 seconds 64 MB Medium LOJ-1098 Debug

Statement Submissions Statistics Tutorial

English

We all know that any integer number n is divisible by 1 and n. That is why these two numbers are not the actual divisors of any numbers. The function SOD(n) (sum of divisors) is defined as the summation of all the actual divisors of an integer number n. For example, SOD(24) = 2+3+4+6+8+12 = 35.

The function CSOD(n) (cumulative SOD) of aninteger n, is defined as below:

$$CSOD(n) = \sum_{i=1}^{n} SOD(i)$$

Given the value of n, your job is to find the value of CSOD(n).

Input

Input starts with an integer T (≤ 1000), denoting the number of test cases.

Each case contains an integer n ($0 \le n \le 2 * 10^9$).

Output

For each case, print the case number and the result. You may assume that each output will fit into a 64 bit signed integer.

Sample

Input	Output
3 2 100 200000000	Case 1: 0 Case 2: 3150 Case 3: 12898681201837053

C++ 17 Editor Upload code

```
1  # Add some code
```

LightOJ

Arrange the Numbers

1 seconds 64 MB Medium LOJ-1095 Debug

Statement Submissions Statistics Tutorial

English

Consider this sequence {1, 2, 3 ... N}, as an initial sequence of first N natural numbers. You can rearrange this sequence in many ways. There will be a total of N! arrangements. You have to calculate the number of arrangements of first N natural numbers, where in first M positions; exactly K numbers are in their initial position.

For Example, N = 5, M = 3, K = 2. You should count this arrangement {1, 4, 3, 2, 5}, here in first 3 positions 1 is in 1^{st} position and 3 in 3^{rd} position. So exactly 2 of its first 3 are in their initial position. But you should not count {1, 2, 3, 4, 5}.

Input

Input starts with an integer T (≤ 1000), denoting the number of test cases.

Each case contains three integers N (1 ≤ N ≤ 1000), M (M ≤ N), K (0 < K ≤ M).

Output

For each case, print the case number and the total number of possible arrangements modulo 1000000007 ($10^9 + 7$).

Sample

Input	Output
2 5 3 2 10 6 3	Case 1: 12 Case 2: 64320

C++ 17 Editor Upload code

```
1   # Add some code
```

LightOJ

Bank Robbery

1 seconds 64 MB Medium LOJ-1163 Debug

Statement Submissions Statistics Tutorial

English

On one very cold morning, Poltu decides to rob a bank. But while trying to hack into the security system, he found that it is locked by some random value. He also found a pattern on the random number, that is if he chops off the last digit of a number A, he gets a new number B. Then he calculates (A-B). He checked the first few numbers of the security system which exactly equals (A-B). Being very excited to have found the pattern, he learns that there are like 500 levels on the security system. He calculated all those numbers by hand but took a lot of time. As a sign of his accomplishment, he left a note on the vault declaring the pattern. You were the first officer on the crime scene and you've obtained the note. So if you can figure out A from (A-B), you can help solve the case!

By the way, Poltu succeeded in robbing the bank and he is running free at this point until you catch him!

Input

Input starts with an integer T (≤ 500), denoting the number of test cases.

Each line contains a single positive integer between 10 and 10^{18} (inclusive), giving the value of A-B.

Output

For each case, print the case number and the possible values of A in ascending order. Separate consecutive numbers with a single space.

Sample

Input	Output
4 31 18 12 17	Case 1: 34 Case 2: 19 20 Case 3: 13 Case 4: 18

LightOJ

Birthday Paradox

1 seconds 64 MB Medium LOJ-1104 Debug

Statement Submissions Statistics Tutorial

English

Sometimes some mathematical results are hard to believe. One of the common problems is the birthday paradox. Suppose you are at a party where there are 23 people including you. What is the probability that at least two people in the party have the same birthday?Surprisingly, the result is more than 0.5.

Here you have to do the opposite. You have given the number of days in a year. Remember that you can be on a different planet, for example, in Mars, a year is 669 days long. You have to find the minimum number of people you have to invite to a party such that the probability of at least two people in the party having the same birthday is at least 0.5.

Input

Input starts with an integer T (≤ 20000), denoting the number of test cases.

Each case contains an integer n ($1 \le n \le 10^5$) in a single line, denoting the number of days in a year on the planet.

Output

For each case, print the case number and the desired result.

Sample

Input	Output
2 365 669	Case 1: 22 Case 2: 30

C++ 17 Editor Upload code

```
1  # Add some code
```

LightOJ

Bi-shoe and Phi-shoe

1 seconds 64 MB Easy LOJ-1370 Debug

Statement Submissions Statistics Tutorial

English

Bamboo Pole-vault is a massively popular sport in Xzhiland. And Master Phi-shoe is a very popular coach for his success. He needs some bamboos for his students, so he asked his assistant Bi-Shoe to go to the market and buy them. Plenty of Bamboos of all possible integer lengths (yes!) are available in the market. According to Xzhila tradition,

Score of a bamboo = Φ (bamboo's length)

(Xzhilans are really fond of number theory). For your information, Φ (n) = numbers less than n which are relatively prime (having no common divisor other than 1) to n. So, score of a bamboo of length 9 is 6 as 1, 2, 4, 5, 7, 8 are relatively prime to 9.

The assistant Bi-shoe has to buy one bamboo for each student. As a twist, each pole-vault student of Phi-shoe has a lucky number. Bi-shoe wants to buy bamboos such that each of them gets a bamboo with a score greater than or equal to his/her lucky number. Bi-shoe wants to minimize the total amount of money spent for buying the bamboos. One unit of bamboo costs 1 Xukha . Help him.

Input

Input starts with an integer T (≤ 100), denoting the number of test cases.

Each case starts with a line containing an integer n ($1 \le n \le 10000$) denoting the number of students of Phi-shoe. The next line contains n space separated integers denoting the lucky numbers for the students. Each lucky number will lie in the range $[1, 10^6]$.

Output

For each case, print the case number and the minimum possible money spent for buying the bamboos. See the samples for details.

Sample

Input	Output

LightOJ

Dice (III)

1 seconds 64 MB Medium LOJ-1248 Debug

Statement Submissions Statistics Tutorial

English

Given a dice with n sides, you have to find the expected number of times you have to throw that dice to see all its faces at least once. Assume that the dice is fair, that means when you throw the dice, the probability of occurring any face is equal.

For example, for a fair two sided coin, the result is 3. Because when you first throw the coin, you will definitely see a new face. If you throw the coin again, the chance of getting the opposite side is 0.5, and the chance of getting the same side is 0.5. So, the result is

$1 + (1 + 0.5 * (1 + 0.5 * ...))$

$= 2 + 0.5 + 0.5^2 + 0.5^3 + ...$

$= 2 + 1 = 3$

Input

Input starts with an integer T (≤ 100), denoting the number of test cases.

Each case starts with a line containing an integer n ($1 \leq n \leq 10^5$).

Output

For each case, print the case number and the expected number of times you have to throw the dice to see all its faces at least once. Errors less than 10^{-6} will be ignored.

Sample

Input	Output
5 1 2 3 6 100	Case 1: 1 Case 2: 3 Case 3: 5.5 Case 4: 14.7 Case 5: 518.7377517640

LightOJ

Digits of Factorial

1 seconds 64 MB Easy LOJ-1045 Debug

Statement Submissions Statistics Tutorial

English

Factorial of an integer is defined by the following function

$$f(n) = \begin{cases} 1, & \text{if } n \text{ is } 0 \\ n \times f(n-1), & \text{if } n > 0 \end{cases}$$

So, factorial of 5 is 120. But in different bases, the factorial may be different. For example, factorial of 5 in base 8 is 170.

In this problem, you have to find the number of digit(s) of the factorial of an integer in a certain base.

Input

Input starts with an integer T (≤ 50000), denoting the number of test cases.

Each case begins with two integers n ($0 \le n \le 10^6$) and base ($2 \le base \le 1000$). Both of these integers will be given in decimal.

Output

For each case of input you have to print the case number and the digit(s) of factorial n in the given base.

Sample

Input	Output
5 5 10 8 10 22 3 1000000 2 0 100	Case 1: 3 Case 2: 5 Case 3: 45 Case 4: 18488885 Case 5: 1

LightOJ

Discovering Gold

1 seconds 64 MB Medium LOJ-1030 Debug

Statement Submissions Statistics Tutorial

English

You are in a cave, a deep cave! The cave can be represented by an 1 x N grid. Some cells in the cave might contain gold!

Initially, you are in position 1. In each move, you throw a perfect 6 sided dice. If you get X in the dice after throwing, you add X to your position and collect all the gold from the new position. If your new position is outside of the cave, you keep throwing the dice again until you get a suitable result. When you reach the N^{th} position you stop your journey.

Given the information about the cave, you have to find the expected amount of gold you can collect using the procedure described above.

Input

Input starts with an integer T (≤ 100), denoting the number of test cases.

Each case contains a blank line and an integer N (1≤ N ≤ 100) denoting the dimension of the cave. The next line contains N space separated integers. The i^{th} integer denotes the amount of gold you will get if you come to the i^{th} cell. You may safely assume that all the given integers will be non-negative and not greater than 1000.

Output

For each case, print the case number and the expected number of gold you will collect. Errors less than 10^{-6} will beignored.

Sample

Input	Output

LightOJ

Efficient Pseudo Code

1 seconds 64 MB Medium LOJ-1054 Debug

Statement Submissions Statistics Tutorial

English

Sometimes it's quite useful to write pseudo codes forproblems. Actually you can write the necessary steps to solve a particularproblem. In this problem you are given a pseudo code to solve a problem and youhave to implement the pseudo code efficiently. Simple! Isn't it? :)

```
pseudo code

{

    take two integers n and m

    let p = n ^ m (n to the power m)

    let sum = summation of all the divisors of p

    let result = sum MODULO 1000,000,007

}
```

Now, given n and m you have to find the desiredresult from the pseudo code. For example if n = 12 and m = 2.Then if we follow the pseudo code, we get:

```
pseudo code

{

    take two integers n and m

    so, n = 12 and m = 2

    let p = n ^ m (n to the power m)

    so, p = 144

    let sum = summation of all the divisors of p

    so, sum = 403, since the divisors of p are:
        {1, 2, 3, 4, 6, 8, 9, 12, 16, 18, 24, 36, 48, 72, 144}

    let result = sum MODULO 1000,000,007
```

```
}
```

Input

Input starts with an integer T (≤ 5000),denoting the number of test cases.

Each test case will contain two integers, n (1 ≤ n)and m (0 ≤ m). Each of n and m will be fit into a 32 bit signed integer.

Output

For each case of input you have to print the case number andthe result according to the pseudo code.

Sample

Input	Output
3 12 2 12 1 36 2	Case 1: 403 Case 2: 28 Case 3: 3751

C++ 17 | Editor | Upload code

```
1   # Add some code
```

LightOJ

Eid

2 seconds 64 MB Medium LOJ-1024 Debug

Statement Submissions Statistics Tutorial

English

In a strange planet there are n races. They are completely different as well as their food habits. Each race has a feast period. Means that the i^{th} race feasts after every x_i de-sec (de-sec is the unit they use for counting time and it is used in both singular and plural forms). And at that particular de-sec, they spend the entire day feasting.

The planet declared the de-sec as 'Eid' in which all the races feast together.

Now given the feasting period for every race you have to find the number of de-sec between two consecutive Eids.

Input

Input starts with an integer T (≤ 225), denoting the number of test cases.

Each case of input will contain an integer n ($2 \leq n \leq 1000$) in a single line. The next line will contain n integers separated by spaces. The i^{th} integer of this line will denote the eating period for the i^{th} race. These integers will be between 1 and 10000.

Output

For each case of input you should print a line containing the case number and the number of de-sec between two consecutive Eids. Check the sample input and output for more details. The result can be big. So, use big integer calculations.

Sample

Input	Output
2 3 2 20 10 4 5 6 30 60	Case 1: 20 Case 2: 60

LightOJ

Extreme GCD

1 seconds 64 MB Medium Hard LOJ-1161 Debug

Statement Submissions Statistics Tutorial

English

All of you know that GCD means the greatest common divisor. So, you must have thought that this problem requires finding some sort of GCD. Don't worry, you are absolutely right!

Given N positive integers, not necessarily distinct, how many ways you can take 4 integers from the N numbers such that their GCD is 1.

Input

Input starts with an integer T (≤ 20), denoting the number of test cases.

Each case starts with an integer N (4 ≤ N ≤ 10000). The next line contains N integers separated by spaces. The integers will be positive and not greater than 10000.

Output

For each case, print the case number and the number of ways you can take the integers as mentioned above.

Sample

Input	Output
3 4 2 4 6 1 5 1 2 4 6 8 10 12 46 100 131 5 6 7 8 9 10	Case 1: 1 Case 2: 4 Case 3: 195

C++ 17 Editor Upload code

```
1  # Add some code
```

LightOJ

Fantasy of a Summation

1 seconds 64 MB Medium LOJ-1213 Debug

Statement Submissions Statistics Tutorial

English

If you think codes, eat codes then sometimes you may get stressed. In your dreams you may see huge codes, as I have seen once. Here is the code I saw in my dream.

```
#include <stdio.h>

int cases, caseno;
int n, K, MOD;
int A[1001];

int main() {
    scanf("%d", &cases);
    while( cases-- ) {
        scanf("%d %d %d", &n, &K, &MOD);
        int i, i1, i2, i3, ... , iK;
        for( i = 0; i < n; i++ ) scanf("%d", &A[i]);

        int res = 0;
        for( i1 = 0; i1 < n; i1++ ) {
            for( i2 = 0; i2 < n; i2++ ) {
                for( i3 = 0; i3 < n; i3++ ) {
                    ...
                    for( iK = 0; iK < n; iK++ ) {
                        res = ( res + A[i1] + A[i2] + ... + A[iK] ) % MOD;
                    }
                    ...
                }
            }
        }
        printf("Case %d: %d\n", ++caseno, res);
    }
    return 0;
}
```

Actually the code was about: 'You are given three integers n, K, MOD and n integers: A_0, A_1, A_2 ... A_{n-1}, you have to write K nested loops and calculate the summation of all A_i where i is the value of any nested loop variable.'

Input

Input starts with an integer T (≤ 100), denoting the number of test cases.

into a 32 bit signed integer.

Output

For each case, print the case number and result of the code.

Sample

Input	Output
2 3 1 35000 1 2 3 2 3 35000 1 2	Case 1: 6 Case 2: 36

C++ 17 Editor Upload code

```
1   # Add some code
```

LightOJ

Finding LCM

1 seconds 64 MB Medium LOJ-1215 Debug

Statement Submissions Statistics Tutorial

English

LCM is an abbreviation used for Least Common Multiple in Mathematics. We say LCM (a, b, c) = L if and only if L is the least integer which is divisible by a, b and c.

You will be given a, b and L. You have to find c such that LCM (a, b, c) = L. If there are several solutions, print the one where c is as small as possible. If there is no solution, report so.

Input

Input starts with an integer T (≤ 325), denoting the number of test cases.

Each case starts with a line containing three integers a b L ($1 \le a, b \le 10^6$, $1 \le L \le 10^{12}$).

Output

For each case, print the case number and the minimum possible value of c. If no solution is found, print `impossible`.

Sample

Input	Output
3 3 5 30 209475 6992 77086800 2 6 10	Case 1: 2 Case 2: 1 Case 3: impossible

C++ 17 Editor Upload code

```
1   # Add some code
```

LightOJ

Harmonic Number (II)

2 seconds 64 MB Medium LOJ-1245 Debug

Statement Submissions Statistics Tutorial

English

I was trying to solve problem **1234 - Harmonic Number**, I wrote the following code

```
long long H( int n ) {
    long long res = 0;
    for( int i = 1; i <= n; i++ )
        res = res + n / i;
    return res;
}
```

Yes, my error was that I was using the integer divisions only. However, you are given n, you have to find H(n) as in my code.

Input

Input starts with an integer T (≤ 1000), denoting the number of test cases.

Each case starts with a line containing an integer n ($1 \le n < 2^{31}$).

Output

For each case, print the case number and H(n) calculated by the code.

Sample

Input	Output
11 1 2 3 4 5 6 7 8 9 10 2147483647	Case 1: 1 Case 2: 3 Case 3: 5 Case 4: 8 Case 5: 10 Case 6: 14 Case 7: 16 Case 8: 20 Case 9: 23 Case 10: 27 Case 11: 46475828386

LightOJ

Harmonic Number

2 seconds 64 MB Medium LOJ-1234 Debug

Statement Submissions Statistics Tutorial

English

In mathematics, the n^{th} harmonic number is the sum of the reciprocals of the first n natural numbers:

$$H_n = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \ldots + \frac{1}{n}$$

$$= \sum_{k=1}^{n} \frac{1}{k}$$

In this problem, you are given n, you have to find H_n.

Input

Input starts with an integer T (≤ 10000), denoting the number of test cases.

Each case starts with a line containing an integer n ($1 \le n \le 10^8$).

Output

For each case, print the case number and the n^{th} harmonic number. Errors less than 10^{-8} will be ignored.

Sample

Input	Output
12 1 2 3 4 5 6 7 8 9 90000000 99999999 100000000	Case 1: 1 Case 2: 1.5 Case 3: 1.8333333333 Case 4: 2.0833333333 Case 5: 2.2833333333 Case 6: 2.450 Case 7: 2.5928571429 Case 8: 2.7178571429 Case 9: 2.8289682540 Case 10: 18.8925358988 Case 11: 18.9978964039 Case 12: 18.9978964139

LightOJ

Help Hanzo

1 seconds 64 MB Medium LOJ-1197 Debug

Statement Submissions Statistics Tutorial

English

Amakusa, the evil spiritual leader has captured the beautiful princess Nakururu. The reason behind this is he had a little problem with Hanzo Hattori, the best ninja and the love of Nakururu. After hearing the news Hanzo got extremely angry. But he is clever and smart, so, he kept himself cool and made a plan to face Amakusa.

Before reaching Amakusa's castle, Hanzo has to pass some territories. The territories are numbered as a, a+1, a+2, a+3 ... b. But not all the territories are safe for Hanzo because there can be other fighters waiting for him. Actually he is not afraid of them, but as he is facing Amakusa, he has to save his stamina as much as possible.

He calculated that the territories which are primes are safe for him. Now given a and b he needs to know how many territories are safe for him. But he is busy with other plans, so he hired you to solve this small problem!

Input

Input starts with an integer T (≤ 200), denoting the number of test cases.

Each case contains a line containing two integers a and b ($1 \le a \le b < 2^{31}$, $b - a \le 100000$).

Output

For each case, print the case number and the number of safe territories.

Sample

Input	Output
3 2 36 3 73 3 11	Case 1: 11 Case 2: 20 Case 3: 4

Notes

A number is said to be prime if it is divisible by exactly two different integers. So, first few primes are {2, 3, 5, 7, 11, 13, 17, ...}.

LightOJ

Helping Cicada

2 seconds 64 MB Medium LOJ-1117 Debug

Statement Submissions Statistics Tutorial

English

Cicada is an insect with large transparent eyes and well-veined wings similar to the "jar flies". The insects are thought to have evolved 1.8 million years ago during the Pleistocene epoch. There are about 2,500 species of cicada around the world which live in temperate tropical climates.

These are all sucking insects, which pierce plants with their pointy mouthparts and suck out the juices. But there are some predators (like birds, the Cicada Killer Wasp) that attack cicadas. Each of the predators has a periodic cycle of attacking Cicadas. For example, birds attack them every three years; wasps attack them every 2 years. So, if Cicadas come in the 12^{th} year, then birds or wasps can attack them. If they come out in the 7^{th} year then no one will attack them.

So, at first they will choose a number N which represents possible life-time. Then there will be an integer M indicating the total number of predators. The next M integers represent the life-cycle of each predator. The numbers in the range from 1 to N which are not divisible by any of those M life-cycles numbers will be considered for cicada's safe-emerge year. And you want to help them.

Input

Input starts with an integer T (≤ 125), denoting the number of test cases.

Each case contains two integers N ($1 \leq N < 2^{31}$) and M ($1 \leq M \leq 15$). The next line contains M positive integers (fits into 32 bit signed integer) denoting the life cycles of the predators.

Output

For each test case, print the case number and the number of safe-emerge days for cicada.

Sample

Input	Output

LightOJ

Iftar Party

1 seconds 64 MB Medium LOJ-1014 Debug

Statement Submissions Statistics Tutorial

English

I have an Ifter party at the 5^{th} day of Ramadan for the contestants. For this reason I have invited C contestants and arranged P piaju's (some kind of food, specially mader for Ifter). Each contestant ate Q piaju's and L piaju's were left (L < Q).

Now you have to find the number of piaju's each contestant ate.

Input

Input starts with an integer T (≤ 325) denoting the number of test cases.

Each case contains two non-negative integers P and L ($0 \le L < P < 2^{31}$).

Output

For each case, print the case number and the number of possible integers in ascending order. If no such integer is found print `impossible`.

Sample

Input	Output
4 10 0 13 2 300 98 1000 997	Case 1: 1 2 5 10 Case 2: 11 Case 3: 101 202 Case 4: impossible

C++ 17 Editor Upload code

```
1  # Add some code
```

LightOJ

Internet Service Providers

1 seconds 64 MB Medium LOJ-1275 Debug

Statement Submissions Statistics Tutorial

English

A group of N Internet Service Provider companies (ISPs) use a private communication channel that has a maximum capacity of C traffic units per second. Each company transfers T traffic units per second through the channel and gets a profit that is directly proportional to the factor T(C - T * N). The problem is to compute the smallest value of T that maximizes the total profit the N ISPs can get from using the channel. Notice that N, C, T, and the optimal T are integer numbers.

Input

Input starts with an integer T (≤ 20), denoting the number of test cases.

Each case starts with a line containing two integers N and C ($0 \le N, C \le 10^9$).

Output

For each case, print the case number and the minimum possible value of T that maximizes the total profit. The result should be an integer.

Sample

Input	Output
6 1 0 0 1 4 3 2 8 3 27 25 1000000000	Case 1: 0 Case 2: 0 Case 3: 0 Case 4: 2 Case 5: 4 Case 6: 20000000

C++ 17 Editor Upload code

```
1  # Add some code
```

LightOJ

Island of Survival

1 seconds 64 MB Medium LOJ-1265 Debug

Statement Submissions Statistics Tutorial

English

You are in a reality show, and the show is way too real that they threw you into an island. Only two kinds of animals are on the island, the tigers and the deer. Though unfortunate but the truth is that, each day exactly two animals meet each other. So, the outcomes are one of the following:

1. If you and a tiger meet, the tiger will surely kill you.
2. If a tiger and a deer meet, the tiger will eat the deer.
3. If two deer meet, nothing happens.
4. If you meet a deer, you may or may not kill the deer (depends on you).
5. If two tigers meet, they will fight each other till death. So, both will be killed.

If someday you are sure that you will not be killed, you leave the island immediately and thus win the reality show. And you can assume that two animals in each day are chosen uniformly at random from the set of living creatures on the island (including you).

Now you want to find the expected probability of you winning the game. Since in outcome (4), you can make your own decision, you want to maximize the probability.

Input

Input starts with an integer T (≤ 200), denoting the number of test cases.

Each case starts with a line containing two integers t ($0 \leq t \leq 1000$) and d ($0 \leq d \leq 1000$) where t denotes the number of tigers and d denotes the number of deer.

Output

For each case, print the case number and the expected probability. Errors less than 10^{-6} will be ignored.

Sample

Input	Output

LightOJ

Josephus Problem

1 seconds 64 MB Medium LOJ-1179 Debug

Statement Submissions Statistics Tutorial

English

The historian Flavius Josephus relates how, in the Romano-Jewish conflict of 67 A.D., the Romans took the town of Jotapata which he was commanding. Escaping, Josephus found himself trapped in a cave with 40 companions. The Romans discovered his whereabouts and invited him to surrender, but his companions refused to allow him to do so. He therefore suggested that they kill each other, one by one, the order to be decided by lot. Tradition has it that the means for affecting the lot was to stand in a circle, and, beginning at some point, count round, every third person being killed in turn. The sole survivor of this process was Josephus, who then surrendered to the Romans. Which begs the question: had Josephus previously practiced quietly with 41 stones in a dark corner, or had he calculated mathematically that he should adopt the 31^{st} position in order to survive?

Now you are in a similar situation. There are n persons standing in a circle. The persons are numbered from 1 to n circularly. For example, 1 and n are adjacent and 1 and 2 are also. The count starts from the first person. Each time you count up to k and the k^{th} person is killed and removed from the circle. Then the count starts from the next person. Finally one person remains. Given n and k you have to find the position of the last person who remains alive.

Input

Input starts with an integer T (≤ 200), denoting the number of test cases.

Each case contains two positive integers n ($1 \leq n \leq 10^5$) and k ($1 \leq k < 2^{31}$).

Output

For each case, print the case number and the position of the last remaining person.

Sample

Input	Output

LightOJ

Just another Robbery

2 seconds 64 MB Medium LOJ-1079 Debug

Statement Submissions Statistics Tutorial

English

As the 'Harry Potter' series is over, Harry has no job. As he wants to make quick money, (he wants everything quickly!), he decided to rob banks. He wants to take calculated risks, and grab as much money as possible. But his friends - Hermione and Ron have decided upon a tolerable probability P of getting caught. They feel that it would be safer if he robs a subset of the banks under probablity P.

Input

Input starts with an integer T (≤ 100), denoting the number of test cases.

Each case contains a real number P, the probability as described above, and an integer N ($0 < N \le 100$), the number of banks he is planning to rob.

Then follow N lines, where line j gives an integer M_j ($0 < M_j \le 100$) and a real number p_j . Bank j contains M_j million dollars, and the probability of getting caught if he decides to rob bank j is p_j. A bank goes bankrupt if it is robbed, and you may assume that all the given probabilities are independent.

Output

For each case, print the case number and the maximum number of millions he can expect to get while the probability of getting caught is less than P.

Sample

Input	Output

3 0.04 3 1 0.02 2 0.03 3 0.05 0.06 3 2 0.03 2 0.03 3 0.05 0.10 3 1 0.03 2 0.02 3 0.05	Case 1: 2 Case 2: 4 Case 3: 6

Notes

For the first case, if he wants to rob bank 1 and 2, then the probability of getting caught is 0.02 + (1 - 0.02) * .03 = 0.0494 which is greater than the given probability (0.04). That's why he has only option, just to rob rank 2.

C++ 17 | Editor | Upload code

```
1   # Add some code
```

LightOJ

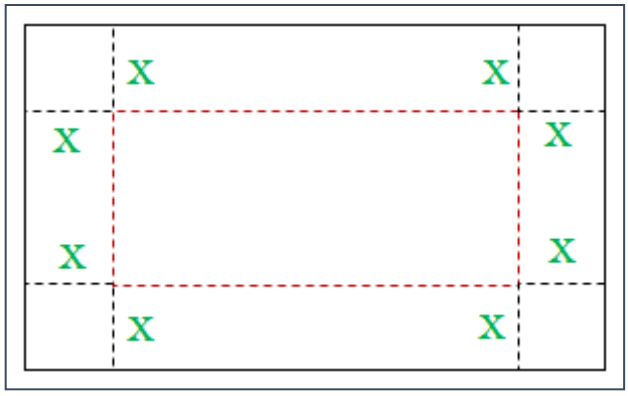
Largest Box

1 seconds 64 MB Medium LOJ-1297 Debug

Statement Submissions Statistics Tutorial

English

In the following figure, you can see a rectangular card. The width of the card is W, length of the card is L and thickness is zero. Four (x*x) squares are cut from the four corners of the card shown by the black dotted lines. Then the card is folded along the red lines to make a box without a cover.



Given the width and height of the box, you will have to find the maximum volume of the box you can make for any value of x.

Input

Input starts with an integer T (≤ 10000), denoting the number of test cases.

Each case starts with a line containing two real numbers L and W (0 < L, W < 100).

Output

For each case, print the case number and the maximum volume of the box that can be made. Errors less than 10^{-6} will be ignored.

Sample

Input	Output
3 2 10 3.590 2.719 8.1991 7.189	Case 1: 4.513804324 Case 2: 2.2268848896 Case 3: 33.412886

LightOJ

LCM from 1 to n

4 seconds 64 MB Medium LOJ-1289 Debug

Statement Submissions Statistics Tutorial

English

Given an integer n, you have to find:

$$lcm(1, 2, 3, \ldots, n)$$

lcm means least common multiple. For example lcm(2, 5, 4) = 20, lcm(3, 9) = 9, lcm(6, 8, 12) = 24.

Input

Input starts with an integer T (≤ 10000), denoting the number of test cases.

Each case starts with a line containing an integer n ($2 \le n \le 10^8$).

Output

For each case, print the case number and lcm(1, 2, 3, ..., n). As the result can be very big, print the result modulo 2^{32}.

Sample

Input	Output
5 10 5 200 15 20	Case 1: 2520 Case 2: 60 Case 3: 2300527488 Case 4: 360360 Case 5: 232792560

C++ 17 Editor Upload code

```
1  # Add some code
```

LightOJ

Leading and Trailing

1 seconds 64 MB Medium LOJ-1282 Debug

Statement Submissions Statistics Tutorial

English

You are given two integers: n and k, your task is to find the most significant three digits, and least significant three digits of n^k.

Input

Input starts with an integer T (≤ 1000), denoting the number of test cases.

Each case starts with a line containing two integers: n ($2 \le n < 2^{31}$) and k ($1 \le k \le 10^7$).

Output

For each case, print the case number and the three leading digits (most significant) and three trailing digits (least significant). You can assume that the input is given such that n^k contains at least six digits.

Sample

Input	Output
5 123456 1 123456 2 2 31 2 32 29 8751919	Case 1: 123 456 Case 2: 152 936 Case 3: 214 648 Case 4: 429 296 Case 5: 665 669

C++ 17 Editor Upload code

```
1   # Add some code
```

LightOJ

Mathematically Hard

1 seconds 64 MB Easy LOJ-1007 Debug

Statement Submissions Statistics Tutorial

English

Mathematically some problems look hard. But with the help of the computer, some problems can be easily solvable.

In this problem, you will be given two integers a and b. You have to find the summation of the scores of the numbers from a to b (inclusive). The score of a number is defined as the following function:

$$score(x) = n^2 \text{ , } n < x \text{ and } gcd(n, x) = 1$$

To illustrate, n is the number of relatively prime numbers with x, which are smaller than x.

For example, For 6, the relatively prime numbers with 6 are {1, 5}. So, score (6) = 2^2 = 4. For 16, the relatively prime numbers with 16 are {1, 3, 5, 7, 9, 11, 13, 15}. So, score (16) = 8^2 = 64.

Now, you have to solve this task.

Input

Input starts with an integer T (≤ 10^5), denoting the number of test cases.

Each case will contain two integers a and b (2 ≤ a ≤ b ≤ 5 * 10^6).

Output

For each case, print the case number and the summation of all the scores from a to b.

Sample

Input	Output
3 6 6 8 8 2 20	Case 1: 4 Case 2: 16 Case 3: 1237

Notes

Two integers are said to be relatively prime, if the greatest common divisor for them is 1.

LightOJ

Mysterious Bacteria

1 seconds 64 MB Medium LOJ-1220 Debug

Statement Submissions Statistics Tutorial

English

Dr. Mob has just discovered a Deathly Bacteria. He named it RC-01. RC-01 has a very strange reproduction system. RC-01 lives exactly x days. Now RC-01 produces exactly p new deadly Bacteria where $x = b^p$ (where b, p are integers). More generally, x is a perfect p^{th} power. Given the lifetime x of a mother RC-01 you are to determine the maximum number of new RC-01 which can be produced by the mother RC-01.

Input

Input starts with an integer T (≤ 50), denoting the number of test cases.

Each case starts with a line containing an integer x. You can assume that x will have magnitude at least 2 and be within the range of a 32 bit signed integer.

Output

For each case, print the case number and the largest integer p such that x is a perfect p^{th} power.

Sample

Input	Output
3 17 1073741824 25	Case 1: 1 Case 2: 30 Case 3: 2

C++ 17 Editor Upload code

```
1  # Add some code
```

LightOJ

Pairs Forming LCM

2 seconds 64 MB Medium LOJ-1236 Debug

Statement Submissions Statistics Tutorial

English

Find the result of the following code:

```
long long pairsFormLCM( int n ) {
    long long res = 0;
    for( int i = 1; i <= n; i++ )
        for( int j = i; j <= n; j++ )
           if( lcm(i, j) == n ) res++; // lcm means least common multiple
    return res;
}
```

A straight forward implementation of the code may time out. If you analyze the code, you will find that the code actually counts the number of pairs (i, j) for which lcm(i, j) = n and (i ≤ j).

Input

Input starts with an integer T (≤ 200), denoting the number of test cases.

Each case starts with a line containing an integer n ($1 \le n \le 10^{14}$).

Output

For each case, print the case number and the value returned by the function pairsFormLCM(n).

Sample

Input	Output

LightOJ

Parallelogram Counting

2 seconds 64 MB Medium LOJ-1058 Debug

Statement Submissions Statistics Tutorial

English

There are n distinct points on the plane, given by their integer coordinates. Find the number of parallelograms whose vertices lie on these points. In other words, find the number of 4-element subsets of these points that can be written as $\{A, B, C, D\}$ such that $AB \parallel CD$, and $BC \parallel AD$.

Input

Input starts with an integer T (≤ 15), denoting the number of test cases.

The first line of each test case contains an integer n ($1 \leq n \leq 1000$). Each of the next n lines, contains 2 space-separated integers x and y (the coordinates of a point) with magnitude (absolute value) of no more than 10^9. For each case, no four points will be on a straight line.

Output

For each case, print the case number and the number of parallelograms that can be formed.

Sample

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

LightOJ

Problem Makes Problem

1 seconds 64 MB Medium LOJ-1102 Debug

Statement Submissions Statistics Tutorial

English

As I am fond of creating easier problems, recently I discovered a new problem. Actually, the problem is 'how can you make n by adding k non-negative integers?' I think a small example will make things clear. Suppose n=4 and k=3. There are 15 solutions. They are:

#	Combination	#	Combination	#	Combination
1	0 0 4	6	1 0 3	11	2 1 1
2	0 1 3	7	1 1 2	12	2 2 0
3	0 2 2	8	1 2 1	13	3 0 1
4	0 3 1	9	1 3 0	14	3 1 0
5	0 4 0	10	2 0 2	15	4 0 0

As I have already told you that I like to make problems easier, you don't have to find the actual result. You should report the result modulo 1000,000,007.

Input

Input starts with an integer T (≤ 25000), denoting the number of test cases.

Each case contains two integer n ($0 \le n \le 10^6$) and k ($1 \le k \le 10^6$).

Output

For each case, print the case number and the result modulo 1000000007 ($10^9 + 7$, it's a prime).

Sample

Input	Output
4 4 3 3 5 1000 3 1000 5	Case 1: 15 Case 2: 35 Case 3: 501501 Case 4: 84793457

LightOJ

Race

1 seconds 64 MB Medium LOJ-1326 Debug

Statement Submissions Statistics Tutorial

English

Disky and Sooma, two of the biggest mega minds of Bangladesh went to a far country. They ate, coded and wandered around, even in their holidays. They passed several months in this way. But everything has an end. A holy person, Munsiji came into their life. Munsiji took them to derby (horse racing). Munsiji enjoyed the race, but as usual Disky and Sooma did their as usual task instead of passing some romantic moments. They were thinking- in how many ways a race can finish! Who knows, maybe this is their romance!

In a race there are n horses. You have to output the number of ways the race can finish. Note that, more than one horse may get the same position. For example, 2 horses can finish in 3 ways.

1. Both first
2. horse1 first and horse2 second
3. horse2 first and horse1 second

Input

Input starts with an integer T (≤ 1000), denoting the number of test cases.

Each case starts with a line containing an integer n (1 ≤ n ≤ 1000).

Output

For each case, print the case number and the number of ways the race can finish. The result can be very large, print the result modulo 10056.

Sample

Input	Output
3 1 2 3	Case 1: 1 Case 2: 3 Case 3: 13

LightOJ

Race to 1 Again

1 seconds 64 MB Medium LOJ-1038 Debug

Statement Submissions Statistics Tutorial

English

Rimi learned a new thing about integers, which is - any positive integer greater than 1 can be divided by its divisors. She is playing with this property now in the form of a game.

She first selects a number N. She then randomly chooses a divisor of N (1 to N) and divides N by the number to obtain a new N. She repeats this procedure until N becomes 1. What is the expected number of turns required for Rimi to end the game - meaning N becomes 1?

Input

Input starts with an integer T (≤ 10000), denoting the number of test cases.

Each case begins with an integer N ($1 \le N \le 10^5$).

Output

For each case of input you have to print the case number and the expected value. Errors less than 10^{-6} will be ignored.

Sample

Input	Output
3 1 2 50	Case 1: 0 Case 2: 2.0 Case 3: 3.0333333333

C++ 17 Editor Upload code

```
1   # Add some code
```

LightOJ

Sigma Function

2 seconds 64 MB Medium LOJ-1336 Debug

Statement Submissions Statistics Tutorial

English

Sigma function is an interesting function in Number Theory. It is denoted by the Greek letter Sigma (σ). This function actually denotes the sum of all divisors of a number. For example σ(24) = 1+2+3+4+6+8+12+24=60. Sigma of small numbers is easy to find but for large numbers it is very difficult to find in a straight forward way. But mathematicians have discovered a formula to find sigma. If the prime power decomposition of an integer is:

$$n = p_1^{e_1} \times p_2^{e_2} \times \cdots \times p_k^{e_k}$$

Then we can write,

$$\sigma(n) = \frac{p_1^{e_1+1} - 1}{p_1 - 1} \times \frac{p_2^{e_2+1} - 1}{p_2 - 1} \times \cdots \times \frac{p_k^{e_k+1} - 1}{p_k - 1}$$

For some n the value of σ(n) is odd and for others it is even. Given a value n, you will have to find how many integers from 1 to n have even value of σ.

Input

Input starts with an integer T (≤ 100), denoting the number of test cases.

Each case starts with a line containing an integer n ($1 \le n \le 10^{12}$).

Output

For each case, print the case number and the result.

Sample

Input	Output
4 3 10 100 1000	Case 1: 1 Case 2: 5 Case 3: 83 Case 4: 947

LightOJ

Sum of Consecutive Integers

2 seconds 64 MB Medium LOJ-1278 Debug

Statement Submissions Statistics Tutorial

English

Given an integer N, you have to find the number of ways you can express N as sum of consecutive integers. You have to use at least two integers.

For example, N = 15 has three solutions, (1+2+3+4+5), (4+5+6), (7+8).

Input

Input starts with an integer T (≤ 200), denoting the number of test cases.

Each case starts with a line containing an integer N ($1 \le N \le 10^{14}$).

Output

For each case, print the case number and the number of ways to express N as sum of consecutive integers.

Sample

Input	Output
5 10 15 12 36 828495	Case 1: 1 Case 2: 3 Case 3: 1 Case 4: 2 Case 5: 47

C++ 17 Editor Upload code

```
1   # Add some code
```

LightOJ

Unlucky Bird

1 seconds 64 MB Medium LOJ-1311 Debug

Statement Submissions Statistics Tutorial

English

A bird was flying on a train line, singing and passing lazy times. After a while it saw a train coming from its behind, so, it speeded up a bit, but remained calm. After a while, it saw another train coming towards it from the front side. The bird remained calm thinking that the train coming towards him would be on another line. But it realized soon after that both the trains were on the same line!

So, the stupid-brave bird made a plan to stop the accident. It flew towards the train which was coming from the front side, and after touching the train the bird turned back immediately and flew back until it could touch the other train. And after that, it turned back and continued this procedure. The birds' intention was to signal the drivers such that they could stop the train.

When the trains were d meter way, the drivers realized the abnormal behavior of the strange bird and saw the opposite trains, and both drivers braked hard! They were able to stop the collision, but they managed to stop in front of each other leaving no space. A thousand lives saved, but the bird couldn't save itself!

For simplicity, we denote the train (that was behind the bird) as the left train and the other one as the right train. The left train had velocity v_1 m/s (meter per second) and the right train had velocity v_2 m/s and they saw each other when they were d meter away. The driver in the left train made a deceleration of a_1 m/s^2 and the driver in the right train made a deceleration of a_2 m/s^2. And the trains just avoided the collision. That means they just stopped when their distance was 0 meter. The bird had a constant velocity of v_3 m/s. And assume that the bird can turn immediately and can keep its constant velocity. When the trains were d meter away, the bird was somewhere between the trains. Your task is to find the distance covered by the brave bird (from this moment) in meters before sacrificing its life for the thousand lives.

Input

Input starts with an integer T (≤ 100), denoting the number of test cases.

Each case starts with a line containing five positive real numbers: v_1 v_2 v_3 a_1 a_2 ($v_1 < v_3$, $v_2 < v_3$). No real number will be greater than 1000. And no number contains more than three digits after the decimal point.

Output

For each case, print the case number, d and the distance covered by the bird. Errors less than 10^{-6} will be ignored.