INTRA HSTU PROGRAMMING CONTEST 2019

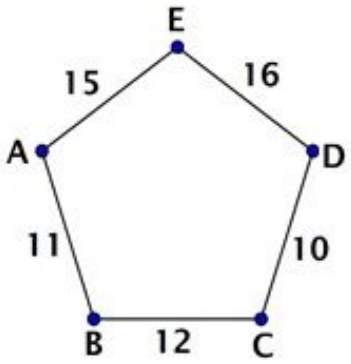
Hosted by Programmers Arena

Sponsored by IT Service Dynamics

A. Puzzling Pentagon

Time limit: 1 sec

Suppose, you are chief inspector Halum. Now you have to crack a puzzle for solving a criminal case. The criminal records indicate that this criminal doesn't have very complex mind. He is as simple as water. He thinks like that as well. Now, look at the picture given at the right side.



Halum is not smart enough to figure out that the cost of each edge is the summation of neighboring vertices. Now you have to help him to evaluate the cost of vertices to solve the case.

Input

First line contains an integer, T ($T \leq 10^5$), the number of test cases.
Next T lines contain 5 integers AB, BC, CD, DE and AE (all edges $\leq |10^9|$) edges of polygon.

Output

For each case output a single line containing the space separated value of A, B, C, D and E correct up to 2 decimal places.

Sample

Input	Output
2 11 12 10 16 15 17 22 26 12 11	4.00 7.00 5.00 5.00 11.00 10.00 7.00 15.00 11.00 1.00

B. Decipher

Time limit: 1 sec

We all know about the popular three investigators. Kishor, Musa and Robin. They are solving a case. The mad scientist Unda has created a secret lab where he does all of his dangerous experiments. Kishor has been able to decipher a secret massage the mad scientist sent to his lab assistant. The massage contains the information about the password of the secret lab. The mad scientist has created an AI that encrypts the password. When someone is going to enter the secret lab, the AI gives a string of numbers in a line where the decimal numbers are written in text form and any combination of the spelling might be given. The lines can contain any combination of [**zero, one, two, three, four, five, six, seven, eight, nine**] with their any possible combination of the spelling.

From the secret massage Kishor came to know that the password will be the addition of the numbers in the line. Kishor wants your help to solve the problem.

Input

The input file will contain several test cases. Each test case consists of one line containing one or more numbers in word. There can be at most 5000 words in a single line of input. All the number in words are in small letter. Input is terminated by a blank line.

Output

For each line of input, print the case number and the answer. The answer does not exceed 5000. For better understanding see the sample input and output.

Sample

Input	Output
tow tow ourf eno four one fiev nien	Case 1: 8 Case 2: 20

C. Sana Loves Factorial!

Time limit: 1 sec

Have patience with all things, but chiefly have patience with yourself to solve the problem.

p^k is not a divisor of n!

where, **p** is a prime number.

n! is the factorial of **n**.

k is a strictly positive integer

Find the minimum value of **p^k** which is not a divisor of **n!** .

Input

Input starts with an integer **T ($\leq 10^3$)**, denoting the number of test cases. Every case will contain two integer **n** and **p ($1 \leq n, p \leq 10^7$) .**

Output

For each case, output a single line containing the value of **k**.

Sample

Input	Output
2 1000 109 720 3	10 357

D. JADU Got a Strange Number

Time limit: 1 sec

One day JADU saw some strange Numbers in his elder sister's book. His sister told that these numbers are known as "Roman Number". Then he thought that " how can I write 1000 in Roman number system?". He searched in that book but didn't find anything. Then he searched in google and found an article. The article is given below.

Roman numerals originated, as the name might suggest, in ancient Rome. There are seven basic symbols: I, V, X, L, C, D and M. The first usage of the symbols began showing up between 900 and 800 B.C.

The numerals developed out of a need for a common method of counting, essential to communications and trade. Counting on one's fingers got out of hand, so to speak, when you reached 10. So, a counting system was devised based on a person's hand.
A single line, or "I," referred to one unit or finger; the "V" represented five fingers, specifically, the V-shape made by the thumb and forefinger. "X" equaled two hands. (See how an X could be two Vs touching at their points?)
Larger Roman numerals developed from other symbols.
M = 1,000 — Originally, the Greek letter phi — Φ — represented this value. It was sometimes represented as a C, I and backwards C, like this: CIↃ — which sort of looks like an M. It's only a coincidence that mille is the Latin word for a thousand.
D = 500 — The symbol for this number was originally IↃ — half of CIↃ.
C = 100 — The original symbol was probably theta — Θ — and later became a C. It only coincidentally also stands for centum, the Latin word for a hundred.
L = 50 — This value was originally represented by a superimposed V and I, or by the letter psi — Ψ — which flattened out to look like an inverted T, and then eventually came to resemble an L.
You have to help JADU to answer his question.

Sample

Input	Output
There is no input.	X (This is not the actual answer, print the actual answer in place of X)

E. Points

Time limit: 1 sec

Mr. Zumzum loves to keep track of all statistics of a football league. But nowadays he is very busy with someone. So he wants some help from you.

In a Football league the point distribution for a single match is like – 3 points for a win, 1 point for a draw and 0 point for a loss. Mr. Zumzum only knows a team played G games and their total points is P. He wants to know how many possible arrangements are there to achieve that P points in G games.

Input

The first line of the input contains an integer T ($T \leq 2*10^4$) indicating the number of cases to be analyzed. Each of the next T lines contain two integers G and P ($1 \leq G \leq 5000$, $0 \leq P \leq 2*10^9$), the number of games played and total points achieved.

Output

For each line of input, print the number of possible arrangements to achieve that P points in G games.

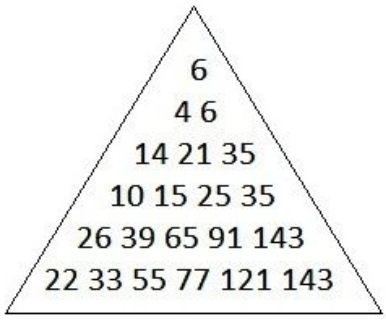
Sample

Input	Output
2 6 6 10 30	3 1

F. Triangle

Time limit: 1 sec

Famous explorer Dr. Indiana Jones has discovered a very old pyramid in MOYNA DIP. The top floor of the pyramid has only one room and the second floor has two rooms and the N^{th} floor has n rooms. The rooms are numbered from left to right. In each floor the leftmost room is numbered 1 and the rightmost room is numbered N. Every room has a fixed number of golds. The first few floors of the pyramid are showed in the picture with the number of golds in it.



As shown in the picture the 1st room of the 1st floor has 6 golds, 1st room of the 2nd floor has 4 golds and the 3rd room of 4th floor has 25 golds. Dr. Jones wants to know the number of golds in Rth room of Nth floor. But he is very busy in searching for another pyramid in MOYNA DIP, so he wants your help.

Input

The first line of the input contains an integer T ($T \leq 1000$) indicating the number of cases to be analyzed.
Each of the next T lines contains two integers N and R ($1 \leq N \leq 10^5$, $1 \leq R \leq N$), indicates the floor and room number of the pyramid.

Output

For each line of input, print the amount of gold.

Sample

Input	Output
3 2 1 5 4 10 3	4 91 115

G. Source to Destination

Time limit: 1 sec

There are **N** cities in your country. Cities are numbered **1** to **N** and connected through **M** roads. You live in city **A** and need to go to city **B**. Before reaching city **B**, you must visit a road which is connected to city **C** and **D**. You can visit city **C** and **D** in any order. i.e. First you can visit city **C** and then visit **D** or vice-versa. Visiting one city to another city have some cost **W**.
You have to find the minimum cost to visit from A to B using a connecting road of C and D.

Input

First line contains an integer T ($T \leq 10$) denotes number of test cases in an input file. Next line contains two integers N ($N \leq 100$) and M ($M < N*(N-1)/2$) denote number of cities and number of roads. Next M lines contain three integers X, Y ($X \neq Y$) and W ($W \leq 100$) represent X city is connected to city Y with cost W. Then a single integer Q ($Q \leq 1000$) which represents number of queries. Next Q lines contain four integers A, B, C and D which described above in problem description.

Output

For each query output a single line.

Sample

Input	Output
1 5 8 1 2 10 3 1 12 3 4 8 2 5 6 2 4 7 3 5 5 4 5 2 1 4 15 2 1 5 3 4 1 4 2 5	22 18

H. Game of Integers

Time limit: 3 sec

Hello Programmers! This is Shiku again!

The problem is very simple. Shiku has ***n*** (to make the problem easy ***n*** will be **at most 30 and is an even number**) numbers. You are given ***q*** ($1 \leq q \leq 100$) queries. In each query you are asked to form a number ***x*** ($0 \leq x \leq 10^{18}$) by summing up the number(s) from ***n*** numbers. You can take any number randomly from ***n*** numbers but you have to remember, you can take each number **once for each query**.
If ***x*** can be formed, then print '***Yes'*** otherwise '***No'.***
For further explanation, see sample input output.

Input

Input a number, ***n***. Next line contains ***n*** numbers (value of each number can be at most 10^9).
After that, input ***q*** (the number of queries). For each query input a number ***x***.

Output

If ***x*** can be formed then print **Yes** otherwise print **No**, format given in Sample I/O.

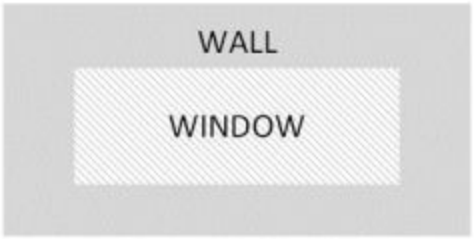
Sample

Input	Output
6 10 12 15 7 9 20 3 10 14 26	Case 1: Yes Case 2: No Case 3: Yes

I. Rupomoni and Window

Time limit: 1 sec

Can you remember Oviman and Ovimani, the cute couple and their brillient child Rupmoni. Rupomoni would like to add a window on the wall in her room. She asks Ovimani to figure out the largest window she can have on her wall. Ovimani consults Oviman to see if there are any structural constraints. Oviman explains that there must be a minimum distance between the wall perimeter and the window perimeter for the wall to hold the window; otherwise the entire structure collapses.



Given the width and height of a rectangular wall and the window-border gap (minimum distance required between the perimeter of the wall and the perimeter of the window), determine the area of the largest rectangular window that can be installed on the wall.

Input

Input will starts with an positive integer T (<100) denotes test cases. Each test case contains one line with three space-separated positive integers, w, h, and d (w, h < 1000, d < 100), representing, respectively, the wall's width, wall's height, and the minimum window-border gap amount needed.

Output

For each test case the output should be an integer, which will represent the area of the largest rectangular window that can be installed. If it is not possible to install a window, output 0 (zero).

Sample

Input	Output
3 40 25 5 30 20 12 999 888 7	450 0 860890