

# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 1:

Time limit: 3.000 seconds

Input: standard input

Output: standard output

## 10071 - Back to High School Physics

A particle has initial velocity and constant acceleration. If its velocity after certain time is  $v$  then what will its displacement be in twice of that time?

### Input

The input will contain two integers in each line. Each line makes one set of input. These two integers denote the value of  $v$  ( $-100 \leq v \leq 100$ ) and  $t$  ( $0 \leq t \leq 200$ ) ( $t$  means at the time the particle gains that velocity)

### Output

For each line of input print a single integer in one line denoting the displacement in double of that time.

### Sample Input & Output

INPUT	OUTPUT
0 0	0
5 12	120
10 10	200

# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 2:

Time limit: 1.000 seconds

## 11547 - Automatic Answer

Last month Alice nonchalantly entered her name in a draw for a Tapmaster 4000. Upon checking her mail today, she found a letter that read:

“Congratulations, Alice! You have won a Tapmaster 4000. To claim your prize, you must answer the following skill testing question.”

Alice’s initial feelings of surprised joy turned quickly to those of dismay. Her lifetime record for skill testing questions is an abysmal 3 right and 42 wrong.

Mad Skills, the leading skill testing question Development Company, was hired to provide skill testing questions for this particular Tapmaster 4000 draw. They decided to create a different skill testing question to each winner so that the winners could not collaborate to answer the question.

Can you help Alice win the Tapmaster 4000 by solving the skill testing question?

### Input

The input begins with  $t$  ( $1 \leq t \leq 100$ ), the number of test cases. Each test case contains an integer  $n$  ( $-1000 \leq n \leq 1000$ ) on a line by itself. This  $n$  should be substituted into the skill testing question below.

### Output

For each test case, output the answer to the following skill testing question on a line by itself: “Multiply  $n$  by 567, then divide the result by 9, then add 7492, then multiply by 235, then divide by 47, then subtract 498. What is the digit in the tens column?”

### Sample Input & Output

INPUT	OUTPUT
2	
637	1
-120	3

# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 3:

Time limit: 3.000 seconds

## 11044 - Searching for Nessy

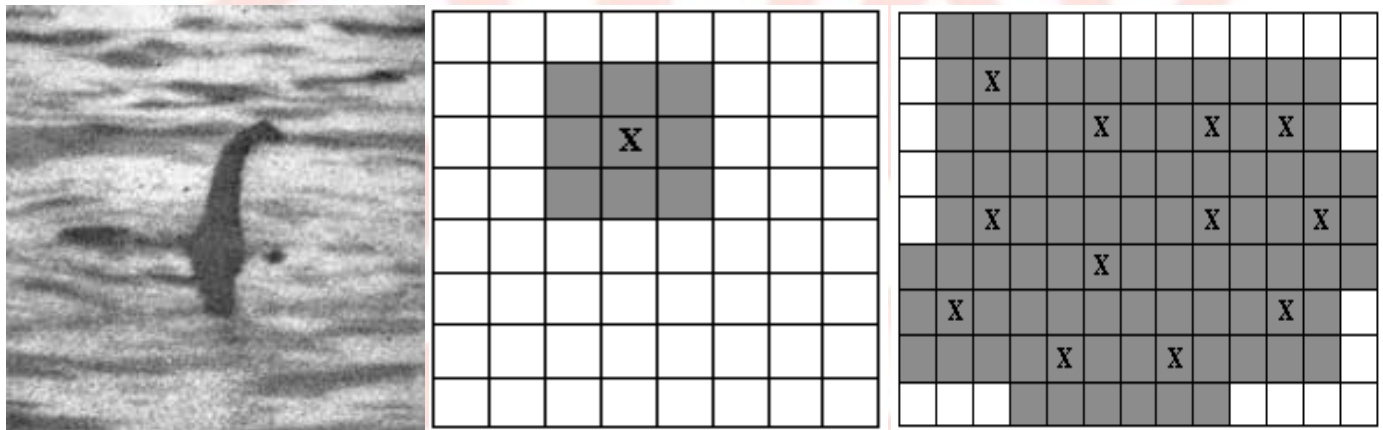
The Loch Ness Monster is a mysterious and unidentified animal said to inhabit Loch Ness, a large deep freshwater loch near the city of Inverness in northern Scotland. Nessie is usually categorized as a type of Lake Monster. [http://en.wikipedia.org/wiki/Loch\\_Ness\\_Monster](http://en.wikipedia.org/wiki/Loch_Ness_Monster).

In July 2003, the BBC reported an extensive investigation of Loch Ness by a BBC team, using 600 separate sonar beams, found no trace of any "sea monster" (i.e., any large animal, known or unknown) in the loch. The BBC team concluded that Nessie does not exist. Now we want to repeat the experiment.

Given a grid of  $n$  rows and  $m$  columns representing the loch,  $6 \leq n, m \leq 10000$ , find the minimum number  $s$  of sonar beams you must put in the square such that we can control every position in the grid, with the following conditions:

one sonar occupies one position in the grid; the sonar beam controls its own cell and the contiguous cells; the border cells do not need to be controlled, because Nessy cannot hide there (she is too big).

For example,



Where  $X$  represents a sonar, and the shaded cells are controlled by their sonar beams; the last figure gives us a solution.

### Input

The first line of the input contains an integer  $t$ , indicating the number of test cases. For each test case, there is a line with two numbers separated by blanks,  $6 \leq n, m \leq 10000$ , that is, the size of the grid ( $n$  rows and  $m$  columns).

### Output

For each test case, the output should consist of one line showing the minimum number of sonars that verifies the conditions above.

### Sample Input & Output

INPUT	OUTPUT
3	
6 6	4
7 7	4
9 13	12

# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 4:

Time limit: 3.000 seconds

## 10300 - Ecological Premium

German farmers are given a premium depending on the conditions at their farmyard. Imagine the following simplified regulation: you know the size of each farmer's farmyard in square meters and the number of animals living at it. We won't make a difference between different animals, although this is far from reality. Moreover you have information about the degree the farmer uses environment-friendly equipment and practices, expressed in a single integer greater than zero. The amount of money a farmer receives can be calculated from these parameters as follows. First you need the space a single animal occupies at an average. This value (in square meters) is then multiplied by the parameter that stands for the farmer's environment friendliness, resulting in the premium a farmer is paid per animal he owns. To compute the final premium of a farmer just multiply this premium per animal with the number of animals the farmer owns.

### Input

The first line of input contains a single positive integer  $n$  ( $n < 20$ ), the number of test cases. Each test case starts with a line containing a single integer  $f$  ( $0 < f < 20$ ), the number of farmers in the test case. This line is followed by one line per farmer containing three positive integers each: the size of the farmyard in square meters, the number of animals he owns and the integer value that expresses the farmer's environment friendliness. Input is terminated by end of file. No integer in the input is greater than 100000 or less than 0.

### Output

For each test case output one line containing a single integer that holds the summed burden for Germany's budget, which will always be a whole number. Do not output any blank lines.

### Sample Input & Output

INPUT	OUTPUT
3	
5	
1 1 1	
2 2 2	
3 3 3	
2 3 4	
8 9 2	38
3	
9 1 8	
6 12 1	
8 1 1	86
3	
10 30 40	
9 8 5	
100 1000 70	7445

# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 5:

Time limit: 3.000 seconds

Input: standard input

Output: standard output

## 10055 - Hashmat the Brave Warrior

Hashmat is a brave warrior who with his group of young soldiers moves from one place to another to fight against his opponents. Before fighting he just calculates one thing, the difference between his soldier number and the opponent's soldier number. From this difference he decides whether to fight or not. Hashmat's soldier number is never greater than his opponent.

### Input

The input contains two integer numbers in every line. These two numbers in each line denotes the number of soldiers in Hashmat's army and his opponent's army or vice versa. The input numbers are not greater than  $2^{32}$ . Input is terminated by End of File.

### Output

For each line of input, print the difference of number of soldiers between Hashmat's army and his opponent's army. Each output should be in separate line.

### Sample Input & Output

INPUT	OUTPUT
10 12	2
10 14	4
100 200	100

# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 6:

Time limit: 3.000 seconds

**Input:** Standard Input

**Output:** Standard Output

## 11172 - Relational Operator

Some operator's checks about the relationship between two values and these operators are called relational operators. Given two numerical values your job is just to find out the relationship between them that is,

- i. First one is greater than the second
- ii. First one is less than the second or
- iii. First and second one is equal.

### Input

First line of the input file is an integer  $t$  ( $t < 15$ ) which denotes how many sets of inputs are there. Each of the next  $t$  lines contain two integers  $a$  and  $b$  ( $|a|, |b| < 1000000001$ ).

### Output

For each line of input produce one line of output. This line contains any one of the relational operators ">", "<" or "=", which indicates the relation that is appropriate for the given two numbers.

### Sample Input & Output

INPUT	OUTPUT
3	
10 20	<
20 10	>
10 10	=



# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 7:

Time limit: 3.000 seconds

## 12372 - Packing for Holiday

Mr. Bean used to have a lot of problems packing his suitcase for holiday. So he is very careful for this coming holiday. He is more serious this time because he is going to meet his fiancée and he is also keeping frequent communication with you as a programmer friend to have suggestions. He gets confused when he buys a gift box for his fiancée because he can't decide whether it will fit in his suitcase or not. Sometimes a box doesn't fit in his suitcase in one orientation and after rotating the box to a different orientation it fits in the suitcase. This type of behavior makes him puzzled.

So to make things much simpler he bought another suitcase having same length, width and height, which is 20 inches. This measurement is taken from inside of the box. So a box which has length, width and height of 20 inches will just fit in this suitcase. He also decided to buy only rectangular shaped boxes and keep a measuring tape in his pocket. Whenever he chooses one gift box, which must be rectangular shaped, he quickly measures the length, width and height of the box. But still he can't decide whether it will fit in his suitcase or not. Now he needs your help. Please write a program for him which calculates whether a rectangular box fits in his suitcase or not provided the length, width and height of the box. Note that, sides of the box must be parallel to the sides of the suitcase.

### Input

Input starts with an integer  $T$  ( $T \leq 100$ ), which indicates the number of test cases. Each of the next  $T$  line contains three integers  $L, W$  and  $H$  ( $1 \leq L, W, H \leq 50$ ) denoting the length, width and height of a rectangular shaped box.

### Output

For each test case, output a single line. If the box fits in the suitcase in any orientation having the sides of the box is parallel to the sides of the suitcase, this line will be "Case #: good", otherwise it will be "Case #: bad". In your output # will be replaced by the case number. Please see the sample input and sample output for exact format.

### Sample Input & Output

INPUT	OUTPUT
2	
20 20 20	Case 1: good
1 2 21	Case 2: bad

# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 8:

Time limit: 1.000 seconds

## 11727 - Cost Cutting

Company XYZ have been badly hit by recession and is taking a lot of cost cutting measures. Some of these measures include giving up office space, going open source, reducing incentives, cutting on luxuries and issuing pink slips.

They have got three (3) employees working in the accounts department and are going to lay-off two (2) of them. After a series of meetings, they have decided to dislodge the person who gets the most salary and the one who gets the least. This is usually the general trend during crisis like this. You will be given the salaries of these 3 employees working in the accounts department. You have to find out the salary of the person who survives.

### Input

The first line of input is an integer  $T$  ( $T < 20$ ) that indicates the number of test cases. Each case consists of a line with 3 distinct positive integers. These 3 integers represent the salaries of the three employees. All these integers will be in the range [1000, 10000].

### Output

For each case, output the case number followed by the salary of the person who survives.

### Sample Input & Output

INPUT	OUTPUT
3	
1000 2000 3000	Case 1: 2000
3000 2500 1500	Case 2: 2500
1500 1500 1200	Case 3: 1500



# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 9:

Time limit: 1.000 seconds

## 11364 - Parking

When shopping on Long Street, Michael usually parks his car at some random location, and then walks to the stores he needs. Can you help Michael choose a place to park which minimizes the distance he needs to walk on his shopping round?

Long Street is a straight line, where all positions are integer. You pay for parking in a specific slot, which is an integer position on Long Street. Michael does not want to pay for more than one parking though. He is very strong, and does not mind carrying all the bags around.

### Input

The first line of input gives the number of test cases,  $1 \leq t \leq 100$ . There are two lines for each test case. The first gives the number of stores Michael wants to visit,  $1 \leq n \leq 20$ , and the second gives their  $n$  integer positions on Long Street,  $0 \leq x_i \leq 99$ .

### Output

Output for each test case a line with the minimal distance Michael must walk given optimal parking.

### Sample Input & Output

INPUT	OUTPUT
2	
4 24 13 89 37	152
6 7 30 41 14 38 42	70

# INTRODUCTION TO COMPETITIVE PROGRAMMING

## Question # 10:

Time limit: 3.000 seconds

## 11332 - Summing Digits

For a positive integer  $n$ , let  $f(n)$  denote the sum of the digits of  $n$  when represented in base 10. It is easy to see that the sequence of numbers  $n, f(n), f(f(n)), f(f(f(n))), \dots$  eventually becomes a single digit number that repeats forever. Let this single digit be denoted  $g(n)$ .

For example, consider  $n = 1234567892$ . Then:

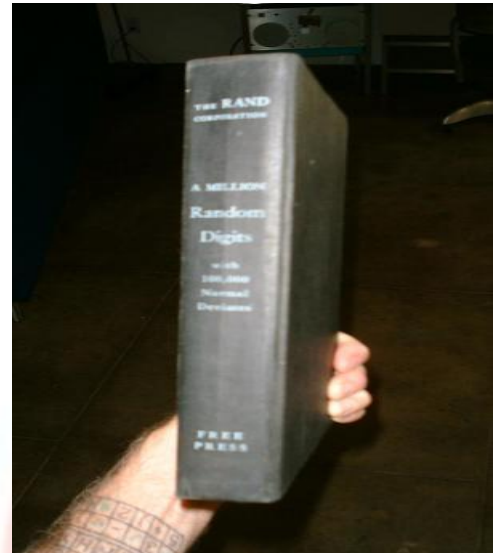
$$f(n) = 1+2+3+4+5+6+7+8+9+2 = 47$$

$$f(f(n)) = 4+7 = 11$$

$$f(f(f(n))) = 1+1 = 2$$

$$\text{Therefore, } g(1234567892) = 2.$$

Each line of input contains a single positive integer  $n$  at most 2,000,000,000. For each such integer, you are to output a single line containing  $g(n)$ . Input is terminated by  $n = 0$  which should not be processed.



## Sample Input & Output

INPUT	OUTPUT
2	2
11	2
47	2
1234567892	2
0	