Greetings From Globussoft

- Given below are 5 Programming questions, you have to solve any 3 out of 5 questions.

- These 5 questions you can attempt in any technology like C/C++, java, .Net, PHP

- To solve these 3 questions you've max. 3 hours.

- While Solving these questions you are not allowed to use any **Search Engine** like Google, Yahoo, Bing …

All the best for your test

Globussoft

QUESTION - 1

Transform the algebraic expression with brackets into RPN form (Reverse Polish Notation). Two-argument operators: +, -, *, /, ^ (priority from the lowest to the highest), brackets ( ). Operands: only letters: a,b,...,z. Assume that there is only one RPN form (no expressions like a*b*c).

Input

```
t [the number of expressions <= 100]
expression [length <= 400]
[other expressions]
```

Text grouped in [ ] does not appear in the input file.

Output

```
The expressions in RPN form, one per line.
```

Example

```
Input:
3
(a+(b*c))
((a+b)*(z+x))
((a+t)*((b+(a+c))^(c+d)))

Output:
abc*+
ab+zx+*
at+bac++cd+^*
```

QUESTION – 2

The most important part of a GSM network is so called *Base Transceiver Station* (*BTS*). These transceivers form the areas called *cells* (this term gave the name to the cellular phone) and every phone connects to the BTS with the strongest signal (in a little simplified view). Of course, BTSes need some attention and technicians need to check their function periodically.

ACM technicians faced a very interesting problem recently. Given a set of BTSes to visit, they needed to find the shortest path to visit all of the given points and return back to the central company building. Programmers have spent several months studying this problem but with no results. They were unable to find the solution fast enough. After a long time, one of the programmers found this problem in a conference article. Unfortunately, he found that the problem is so called "Travelling Salesman Problem" and it is very hard to solve. If we have *N*

BTSes to be visited, we can visit them in any order, giving us $N!$ possibilities to examine. The function expressing that number is called factorial and can be computed as a product $1.2.3.4....N$. The number is very high even for a relatively small N.

The programmers understood they had no chance to solve the problem. But because they have already received the research grant from the government, they needed to continue with their studies and produce at least *some* results. So they started to study behaviour of the factorial function.

For example, they defined the function Z. For any positive integer N, $Z(N)$ is the number of zeros at the end of the decimal form of number $N!$. They noticed that this function never decreases. If we have two numbers $N_1<N_2$, then $Z(N_1) <= Z(N_2)$. It is because we can never "lose" any trailing zero by multiplying by any positive number. We can only get new and new zeros. The function Z is very interesting, so we need a computer program that can determine its value efficiently.

Input

There is a single positive integer T on the first line of input (equal to about 100000). It stands for the number of numbers to follow. Then there are T lines, each containing exactly one positive integer number N, $1 <= N <= 1000000000$.

Output

For every number N, output a single line containing the single non-negative integer $Z(N)$.

Example

Sample Input:

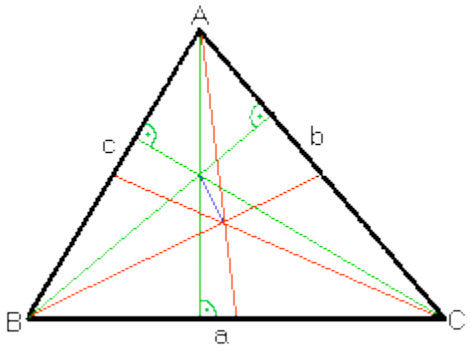
```
6
3
60
100
1024
23456
8735373
```

Sample Output:

```
0
14
24
253
5861
2183837
```

QUESTION – 3

Given the length of side a of a triangle and the distances from the centroid (the point of concurrence of the medians - red in the picture) to all sides: a, b and c, calculate this triangle's area and the distance (blue line) from the orthocenter (the point of concurrence of the heights - green in the picture) to the centroid.



Input

In the first line integer n - the number of test cases (equal to about 1000). The next n lines - 4 floating point values: the length of side a, and distances from the centroid to sides a, b and c.

Output

n lines consisting of 2 floating point values with 3 digits after the decimal point: the area of the triangle and the distance from the orthocenter to centroid.

Example

Input:

```
2
3.0 0.8660254038 0.8660254038 0.8660254038
657.8256599140 151.6154399062 213.5392629932 139.4878846649
```

Output:

```
3.897 0.000
149604.790 150.275
```

QUESTION – 4

The Antique Comedians of Malidinesia prefer comedies to tragedies. Unfortunately, most of the ancient plays are tragedies. Therefore the dramatic advisor of ACM has decided to transfigure

some tragedies into comedies. Obviously, this work is very hard because the basic sense of the play must be kept intact, although all the things change to their opposites. For example the numbers: if any number appears in the tragedy, it must be converted to its reversed form before being accepted into the comedy play.

Reversed number is a number written in arabic numerals but the order of digits is reversed. The first digit becomes last and vice versa. For example, if the main hero had 1245 strawberries in the tragedy, he has 5421 of them now. Note that all the leading zeros are omitted. That means if the number ends with a zero, the zero is lost by reversing (e.g. 1200 gives 21). Also note that the reversed number never has any trailing zeros.

ACM needs to calculate with reversed numbers. Your task is to add two reversed numbers and output their reversed sum. Of course, the result is not unique because any particular number is a reversed form of several numbers (e.g. 21 could be 12, 120 or 1200 before reversing). Thus we must assume that no zeros were lost by reversing (e.g. assume that the original number was 12).

Input

The input consists of *N* cases (equal to about 10000). The first line of the input contains only positive integer *N*. Then follow the cases. Each case consists of exactly one line with two positive integers separated by space. These are the reversed numbers you are to add.

Output

For each case, print exactly one line containing only one integer - the reversed sum of two reversed numbers. Omit any leading zeros in the output.

Example

```
Sample input:

3
24 1
4358 754
305 794

Sample output:

34
1998
1
```

QUESTION – 5

In the age of television, not many people attend theater performances. Antique Comedians of Malidinesia are aware of this fact. They want to propagate theater and, most of all, Antique Comedies. They have printed invitation cards with all the necessary information and with the programme. A lot of students were hired to distribute these invitations among the people. Each student volunteer has assigned exactly one bus stop and he or she stays there the whole day and gives invitation to people travelling by bus. A special course was taken where students learned how to influence people and what is the difference between influencing and robbery.

The transport system is very special: all lines are unidirectional and connect exactly two stops. Buses leave the originating stop with passangers each half an hour. After reaching the destination stop they return empty to the originating stop, where they wait until the next full half an hour, e.g. X:00 or X:30, where 'X' denotes the hour. The fee for transport between two stops is given by special tables and is payable on the spot. The lines are planned in such a way, that each round trip (i.e. a journey starting and finishing at the same stop) passes through a *Central Checkpoint Stop* (CCS) where each passenger has to pass a thorough check including body scan.

All the ACM student members leave the CCS each morning. Each volunteer is to move to one predetermined stop to invite passengers. There are as many volunteers as stops. At the end of the day, all students travel back to CCS. You are to write a computer program that helps ACM to minimize the amount of money to pay every day for the transport of their employees.

Input

The input consists of *N* cases. The first line of the input contains only positive integer *N*. Then follow the cases. Each case begins with a line containing exactly two integers *P* and *Q*, *1 <= P,Q <= 1000000*. *P* is the number of stops including CCS and *Q* the number of bus lines. Then there are *Q* lines, each describing one bus line. Each of the lines contains exactly three numbers - the originating stop, the destination stop and the price. The CCS is designated by number *1*. Prices are positive integers the sum of which is smaller than *1000000000*. You can also assume it is always possible to get from any stop to any other stop.

Output

For each case, print one line containing the minimum amount of money to be paid each day by ACM for the travel costs of its volunteers.

Example

```
Sample input:

2
2 2
1 2 13
2 1 33
4 6
1 2 10
2 1 60
1 3 20
```

```
3 4 10
2 4 5
4 1 50
```

Sample output:

```
46
210
```