Shiraz University Local Contest 2011

A – Number Coding

In a special coding proposed by ACM group of Shiraz University, each integer number can be represented by the number of its prime factors followed by the frequency numbers of factors arranged in an increasing order. For example, $32=2^5$ has only 1 prime factor and its frequency is 5. So, it can be represented by 15. Also, $43560=2^3\times3^2\times5^1\times11^2$ has 4 factors with frequencies 3, 2, 1 and 2, hence coded as 41223. As another example, 7168 is coded into 2110. Considering only 9 prime factors {2, 3, 5, 7, 11, 13, 17, 19, 23} and given a coded number C, you should identify how many numbers share the coded representation C.

Input

The first line of the input contains an integer T as the number of test-cases. Each following line gives a code word C which will not exceed 20 characters.

Output

Print for each test-case, in a separate line, the number of possible integers that have identical translation C.

Sample Input	Sample Output
4 123 213 31312 311111	9 72 504 504

B – Ball Blasting Game

Morteza is playing a ball blasting game. In this game there is a chain of different colored balls. He is expected to explode as many balls as possible by aligning same-colored balls and making a sequence of them. To align balls, he can aim and shoot a new ball into a position in the chain, thus adding it there. If the new ball makes a sequence of two or more same-colored balls with its immediate neighbors, then the sequence blows up breaking the chain up into two parts. The two sections draw together to reform a single chain. Again, if the colors of the newly aligned balls (on joining ends of the two sections) match, the entire run of the suit explodes. New explosions ensue as long as joining sections bear new matches.

For instance, consider a chain symbolized in the string "GGGBWW**RR**WRR", with each letter representing the color of the corresponding ball in the chain. The train of balls therefore is composed of 6 sequences in 4 colors. Here, a ball can be added in 12 different positions. Shooting a red ball between the two middle red balls triggers two successive explosions which leave the chain "GGGBRR".

Morteza reaches a challenge stage in which he has only one ball to shoot but the color of which he can choose. He may not advance to the next stage unless he takes a shot that sets off the largest possible number of explosions. He has to replay all through this repetitive stage should he fail and needs your help in finding the largest possible number of successive explosions to save him a great deal of suffering.

Input

The first line of input contains an integer T≤100 denoting the number of test-cases. Each test-case is represented by a single line containing a string of upper case letters ('A'-'Z') of size of at most 100,000.

Output

For each test-case, output on a single line the maximum possible number of explosions.

Sample Input	Sample Output
3 GGGBWWRRWRR XAABCBAXAAAA CCC	2 4 1

C – Central Post Office

One of the post services companies in a country plans to designate one of its branches as the central office. The company has a branch in each and every city in the country. The cities are so connected by roads that to go from any city to another, there is a unique sequence of roads to take. The central office is in charge of dispatching parcels to all other branches. For this purpose, a car is used that starting from the central office goes through all cities to the last one delivering their parcels. As time is always a top priority in post services, the company's administration wants a designation which minimizes dispatching times. If the car travels the distance between any two adjacent cities in one hour, calculate the minimum total dispatching time T_m, considering the optimal designation.

Input

The first line of input contains an integer T≤100 denoting the number of test-cases. Each test-case begins with an integer 1≤N≤10,000 denoting the number of cities (numbered from 1 to N) of the country, on a separate line. The i^{th} line of the following N lines starts with the number M_i of the cities adjacent to the i^{th} city followed by M_i integers, the neighboring city indexes.

Output

For each test-case, output on a single line the minimum dispatching time T_m.

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

D – Glimmr in Distress

In a special compression technique used by a photo sharing site Glimmr to compress its database of images, a simple preprocessing is used. As you may know, any digital image is comprised of several matrices, one for each channel, of binary values. Each of these matrices can further be broken up into a couple of Boolean matrices associated with binary places (2^0 2^1 2^2…). In other words, each digit of a binary pixel value appears in the corresponding coordinates of a B&W (black-and-white) image pertinent to the digit's place-value. Thus, we can represent a color image with a collection of same-size B&W images. The preprocessing phase is carried out on each B&W image which without loss of generality we assume to be of size $2^n \times 2^n$. As a result, the image is converted to a string of characters '0', '1' and '2'. More specifically, the string representation of an image of uniform 1s is a single-character string "1" and that of uniform 0s, the string "0". For non-uniform values, the image is split into four sub-images of size $2^{n-1} \times 2^{n-1}$, as seen below. The string translation in this case starts with the character '2' followed by the string codes for sub-images in order from left to right, top to bottom, that is $S_1S_2S_3S_4$.

I_1	I_2
I_3	I_4

For example, the string representing the following 2×2 image is "21001".

1	0
0	1

These strings are further processed to achieve a good level of compression and then stored. Unfortunately, a recent act of sabotage (probably with an EMP micro bomb) by a rival company has left their database damaged with garbled values. As a result, when one wants to decompress and reconstruct images they run into strings that may have some characters altered and some others lost. Therefore, Glimmr is pursuing a disaster recovery plan. To start with, they intend to identify damaged strings and for those seemingly unharmed, find their minimum consistent image size, since size information is also missing.

You are called in to help them with this task.

Input

The first line of input contains an integer T≤1,000 denoting the number of test-cases. Each test-case is represented by a single line containing a string of size of at most 2,500 of characters 0, 1 and 2.

Output

For each test-case, if the string is not a valid representation of a B&W image output "Not Possible" with no quotes. And if it is valid, output the minimum possible size of the image in the format 2^n*2^n as seen in the sample output.

Sample Input	Sample Output
4 2022111011111 2112002000001 20102102101010 1	2^3*2^3 Not Possible Not Possible 2^0*2^0

E – Our Fair Contest!

In an ACM regional contest held in Shiraz University, two problems A and B are given. The problems have positive weights and the final score of a team is computed as the weighted sum of the numbers of solved test-cases for the two problems. Only the best k teams will advance to the world finals where the integer 1≤k≤n-1 is randomly chosen from a uniform distribution by the ACM international headquarters and announced only after the list of the team scores is submitted.

The weights of the problems are not known to the participating teams prior to the end of the contest following which the site manager will work out the weights so as to maximize the expected number of qualifying teams from Shiraz University over the unknown value of k.

Given the number of solved test-cases for each problem by each team, compute the maximum possible value of the expected number of Shiraz University teams which will compete in the final world competition.

Input

On the first line of input is given an integer T of the number of test-cases. The first line of each test-case contains two integers 1<n≤10,000, the total number of teams, followed by m<20, the number of teams from Shiraz University. For the next n lines, the i^{th} line has the number of test-cases (not greater than 10,000) solved by the i^{th} team for problems A and B in the same order. The first m lines are associated with the teams from Shiraz University.

Output

For each test-case print on a new line the maximum possible value of the expected number of teams from Shiraz University that will have berths in the world finals. All values have to be printed in exactly 4 decimal places.

Sample Input	Sample Output
5	1.0000
2 2	0.0000
1 2	1.0000
2 1	0.0000
2 0	1.0000
1 2	
2 1	
2 1	
1 2	
2 1	
1 2	
2 2	
4 2	
4 3	
1 2	
3 4	
2 1	

F - Grid of Lamps

We have a grid of lamps. Some of the lamps are on, while others are off. The luminosity of a row/column is the number of its lighted lamps. You are given a permutation of the luminosities of the rows and a permutation of the columns'. Unfortunately, these values are not accurate but we know at least that they are not overestimates. You should tell us the minimum possible number of lighted lamps in this grid.

As an example, consider the following grid. The lighted lamps are shown by 1's and unlighted ones by 0's.

1	0	1	1	0
1	1	1	1	1
0	0	0	1	1
0	1	1	1	0
1	0	0	0	0

The actual luminosities of the rows are <3,5,2,3,1>. A permutation of them could be <1,2,5,3,3>, and the inexact values you'd be given could be <1,1,4,2,3>.

Input

The first line of input contains an integer T≤100 denoting the number of test-cases. Each test-case begins with two integers M and N, both in the interval [1, 1000], determining the number of rows and columns of the grid respectively. The next two lines give the luminosities, the first for rows (M values) and the second for columns (N values).

Output

For each test-case, on a single line, output the minimum conceivable number of lighted lamps.

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

G – The Game

A simple game, called KillHer-I, is investigated in Shiraz University game design group. This game involves two players, each with one piece, and is played on a board of n squares arranged in a line with a fair dice of m numbers (1, 2, 3, …, m). Initially, the two pieces are in squares on either ends of the board. The players take turns to move their pieces forward by the number of squares indicated by the dice roll. The player who completes their move in the square already occupied by the opponent's piece, captures it and wins the game. Otherwise, if a player overpasses their opponent's piece in their move, they lose. The game continues until one of the players wins (or the other one loses).

This game's primary vice is that the whole game is completely decided by chance. Therefore, another version of this game is designed, called KillHer-II. In this variant, each player may choose to move their piece from 1 to throw-of-the-dice squares.

The designers want to know, the probability of the starting player winning the games KillHer-I and KillHer-II independently, given the board size n and dice sides m.

Input

The first line of input contains an integer T≤200 as the number of test-cases. Each of the following T lines feature two integers n and m both less than 100,000.

Output

For each test-case, you should print on a new line, the probability of the starting player winning first in KillHer-I and then in KillHer-II, separated by one space. Each probability should be expressed in exactly 4 decimal places.

Sample Input	Sample Output
5 1 1 1 4 2 2 10 1 18467 41 9514 22813	1.0000 1.0000 0.2500 1.0000 0.7500 0.5000 0.0000 0.0000 0.5000 0.5000 0.3410 0.7057

H - Span

Given an array of n integers $X_{1 \le i \le n}$, the span S of X is an array of n integers with S_i being the maximum number of consecutive elements X_j immediately preceding X_i such that $X_j \le X_i$. In mathematical notation, elements of S are thus defined,

$$S_i = |A_i|,$$

$$A_i = \{ j \le i | \forall k \ (j \le k \le i) \ (X_k \le X_i) \}.$$

As an example, the span of the array X=[40,2, 10, 50, 30, 15], is the array S=[1, 1, 2, 4, 1, 1].

Now suppose, for given values of integers m and n, that $X_{1 \le i \le n}$=(P_i mod m) where P_i is the i^{th} prime number. We need to compute the sum-modulus-m of the elements of array S, span of X. If m=10 and n=7, we have X=[2, 3, 5, 7, 1, 3, 7] and S=[1, 2, 3, 4, 1, 2, 7]. The desired value is then, ((1+2+3+4+1+2+7) mod 10) = 0.

Input

The input file provides an integer T, on the first line, as the number of test-cases. For the next T lines, each line represents a test-case with two integers n and m both in the interval [1, 100000].

Output

For each test-case print the sum of the elements of S mod m, as described above.

Sample Input	Sample Output
3 7 10 10 16 10 7	0 5 6

I - Interesting Sequences

For a sequence of integer numbers $<x_1,x_2,\ldots,x_n>$, a contiguous subsequence $<x_i,x_{i+1},\ldots,x_j>$ where $i<j\leq n$, is called "interesting" if its first and last elements are equal (i.e., $x_i=x_j$). Two interesting subsequences $S_1=<x_i,x_{i+1},\ldots,x_j>$ and $S_2=<x_a,x_{a+1},\ldots,x_b>$ are called conflict-free if either $a\geq j$ or $i\geq b$.

For a given sequence of known size, find the maximum number of interesting subsequences which are pairwise conflict-free.

Input

The first line of input contains an integer $T\leq 100$ denoting the number of test-cases. Each test-case begins with an integer $1\leq N\leq 100{,}000$, on a separate line, denoting the size of the sequence. The following line contains N positive integers each between 1 and 100,000 (inclusive).

Output

For each test-case, output on a single line the maximum number of pairwise conflict-free interesting subsequences.

Sample Input	Sample Output
3 6 1 2 1 3 1 2 4 2 4 2 4 9 10 2 2 10 3 4 5 4 3	2 1 2

J – Smallest Polygon

Given 3≤n≤10 distinct points (x,y) with integer coordinates, Mohammad wants to find a polygon with minimum area which has these points as its vertices. However, he would have liked to minimize the polygon perimeter as well. He is curious to know how long on perimeter he is compromising to get the smallest-area polygon.

Input

An integer T as the number of test-cases is given on the first line. Each test-case consists of an integer n, on a single line, as the number of points. For the following n lines, x and y coordinates of the points are given as two integers (0≤x,y≤100).

Output

Print on a single line per test-case, the difference between the perimeters of the minimum-area polygon and the minimum-perimeter polygon. Round the results to 4 decimal places.

Sample Input	Sample Output
2 3 1 1 1 2 2 1 4 0 0 1 1 0 2 2 1	0.0000 0.6503