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.

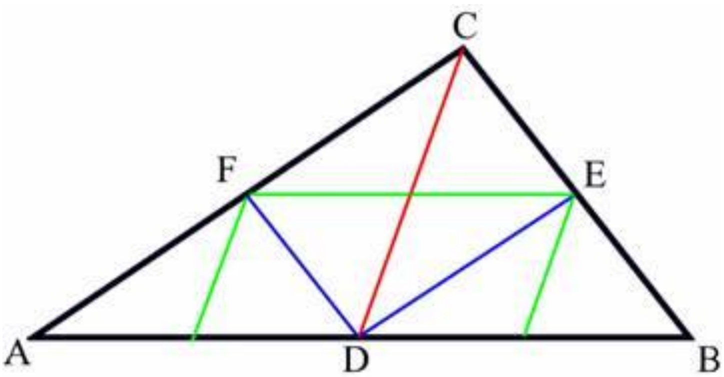
- While Solving these questions you are not allowed to use any **Search Engine** like Google, Yahoo, Bing …

All the best for your test

Globussoft

QUESTION – 1

A triangle can be divided into two equal triangles by drawing a median on its largest edge (in the figure below such a division is shown with the red line). Then the smaller two triangles can be divided in similar fashion into equal triangles (shown in the picture below with blue lines). This process can continue forever.



Some mathematicians have found that when we split a triangle into smaller ones using the method specified above we have only some "styles" of triangles that only differ in size. So now given the lengths of the sides of the triangle your job is to find out how many different styles of small triangles we have. (Two triangles are of same style if they are similar.)

Input

First line of the input file contains an integer N $(0 < N < 35)$ that indicates how many lines of input there are.

Each line contains three integers a, b, c $(0 < a,b,c < 100)$ which indicate the sides of a valid triangle. (A valid triangle means a real triangle with positive area.)

Output

For each line of input you should produce an integer T, which indicates the number of different styles of small triangles, formed for the triangle at input. Look at the example for details. You can safely assume that for any triangle T will be less than 100.

Example

```
Input:
2
3 4 5
12 84 90

Output:
3
41
```

QUESTION – 2

For each prefix of a given string S with N characters (each character has an ASCII code between 97 and 126, inclusive), we want to know whether the prefix is a periodic string. That is, for each i ($2 <= i <= N$) we want to know the largest $K > 1$ (if there is one) such that the prefix of S with length i can be written as A^K, that is A concatenated K times, for some string A. Of course, we also want to know the period K.

Input

The first line of the input file will contains only the number *T (1 <= T <= 10)* of the test cases.

Each test case consists of two lines. The first one contains N ($2 <= N <= 1\,000\,000$) – the size of the string S. The second line contains the string S.

Output

For each test case, output "Test case #" and the consecutive test case number on a single line; then, for each prefix with length **i** that has a period **K** > 1, output the prefix size **i** and the period **K** separated by a single space; the prefix sizes must be in increasing order. Print a blank line after each test case.

Example

Input:
```
2
3
aaa
12
aabaabaabaab
```

Output:
```
Test case #1
2 2
3 3

Test case #2
2 2
6 2
9 3
12 4
```

QUESTION – 3

In the late Middle Ages the University of Byteland was no different than any other university of the day. One of those gloomy places where philosophers brooded over the essence of life, theologians did likewise and quaralled with philosophers, while alchemists developed new caustic types of green shampoo in their futile search for gold. The thing that worried the Chancellor most was that none of the staff seemed to be in the least capable of making money in any form. When he complained about this to the Director of Human Resources, the Director came up with a brilliantly simple theory. He claimed that this lack of productivity was the direct consequence of the isolated model of work, and that wonders could be achieved by promoting teamwork.

The Director intends to assign every scientist to some 3-person workgroup. The members of the workgroup should then select which of them is to act as the group leader. And this of course is the root of the problem. Every scientist will tolerate either himself or one of his acquaintances as the leader of his group, but will never allow anyone else to have this privilege. So when creating workgroups it is necessary to bear in mind that every group should have at least one suitable candidate for the role of group leader, accepted by all its members.

Although everyone at the University knows *of* everyone else indirectly (as acquaintances of acquaintances of acquaintances of...), the number of direct acquaintances that every scientist has is relatively small - either equal to 2, or to 3. Even so, it ought to be possible to assign the vast majority of scientists to workgroups. Quite naturally, the dubious pleasure of performing this task has been left to you, the Acting University Algorithmist.

Input

Input starts with a single integer t, the number of test cases (t<=100). t test cases follow.

Each test case begins with a line containing two integers n m (4<=n<=m<=20000, n is the number of scientists and is divisible by 4). Exactly m lines follow containing a pair of integers a_i b_i each which denote that scientists a_i and b_i are acquaintances (1<=a_i, b_i<=n, each scientist has either 2 or 3 acquaintances). Acquaintanceship is mutual.

Output

For each test case, output a line containing a single integer k - the number of workgroups you have formed. In each of the next k lines output exactly 3 integers, representing the numbers of scientists belonging to respective workgroups.

Your solution will be regarded as incorrect if for some test case more than *25%* of all scientists are left without a valid assignment to a workgroup.

Example

```
Input:
1
8 10
1 2
1 3
2 5
4 6
3 7
2 3
5 6
6 7
7 8
8 4

Output:
2
1 3 7
4 5 6
```

QUESTION – 4

You are given scales for weighing loads. On the left side lies a single stone of known weight $W<2^N$. You own a set of N different weights, weighing 1, 2, 4, ..., 2^{N-1} units of mass respectively. Determine how many possible ways there are of placing some weights on the sides of the scales, so as to balance them (put them in a state of equilibrium). Output this value modulo a small integer D.

Input

The input begins with the integer t, the number of test cases. Then t test cases follow.

For each test case, the first line contains three integers: N L D, where N denotes the number of weights at your disposal, L is the length of the binary representation of number W, and D is the modulus (1<= L<= N<= 1000000, 2<= D<=100). The second line contains the value of W, encoded in the binary system as a sequence of exactly L characters 0 or 1 without separating spaces.

Output

For each test case, output a single line containing one integer - the calculated number of possible weight placements, modulo D.

Example

Sample input:

```
2
6 4 6
1000
6 6 100
100110
```

Sample output:

```
3
5
```

QUESTION – 5

A manufacturer of sweets has started production of a new type of sweet called *rock*. Rock comes in sticks composed of one-centimetre-long segments, some of which are sweet, and the rest are sour. Before sale, the rock is broken up into smaller pieces by splitting it at the connections of some segments.

Today's children are very particular about what they eat, and they will only buy a piece of rock if it contains more sweet segments than sour ones. Try to determine the total length of rock which can be sold after breaking up the rock in the best possible way.

Input

The input begins with the integer t, the number of test cases. Then t test cases follow.

For each test case, the first line of input contains one integer N - the length of the stick in centimetres (1<=N<=200). The next line is a sequence of N characters '0' or '1', describing the segments of the stick from the left end to the right end ('0' denotes a sour segment, '1' - a sweet one).

Output

For each test case output a line with a single integer: the total length of rock that can be sold after breaking up the rock in the best possible way.

Example

Sample input:

```
2
15
100110001010001
16
0010111101100000
```

Sample output:

```
9
13
```