



A • Sixth Grade Math

In sixth grade, students are presented with different ways to calculate the *Least Common Multiple (LCM)* and the *Greatest Common Factor (GCF)* of two integers. The **LCM** of two integers **a** and **b** is the smallest positive integer that is a multiple of both **a** and **b**. The **GCF** of two non-zero integers **a** and **b** is the largest positive integer that divides both **a** and **b** without remainder.

For this problem you will write a program that determines both the **LCM** and **GCF** for positive integers.

Input

The first line of input contains a single integer **N**, ($1 \leq N \leq 1000$) which is the number of data sets that follow. Each data set consists of a single line of input containing two positive integers, **a** and **b**, ($1 \leq a, b \leq 1000$) separated by a space.

Output

For each data set, you should generate one line of output with the following values: The data set number as a decimal integer (start counting at one), a space, the **LCM**, a space, and the **GCF**.

Sample Input	Sample Output
3	1 10 5
5 10	2 161 1
7 23	3 168 14
42 56	



B • Cryptoquote

A *cryptoquote* is a simple encoded message where one letter is simply replaced by another throughout the message. For example:

Encoded: **HPC PJVYMIY**
Decoded: **ACM CONTEST**

In the example above, **H=A**, **P=C**, **C=M**, **J=O**, **V=N**, **Y=T**, **M=E** and **I=S**. For this problem, you will decode messages.

Input

The first line of input contains a single integer N , ($1 \leq N \leq 1000$) which is the number of data sets that follow. Each data set consists of two lines of input. The first line is the encoded message. The second line is a 26 character string of upper case letters giving the character mapping for each letter of the alphabet: the first character gives the mapping for **A**, the second for **B** and so on. Only upper case letters will be used. Spaces may appear in the encoded message, and should be preserved in the output string.

Output

For each data set, you should generate one line of output with the following values: The data set number as a decimal integer (start counting at one), a space and the decoded message.

Sample Input	Sample Output
2 HPC PJVYMIY BLMRGJIASOPZEFDCCKWYHUNXQTV FDY GAI BG UKMY KIMHOTSQYRLCUZPAGWJNBVDXEF	1 ACM CONTEST 2 THE SKY IS BLUE



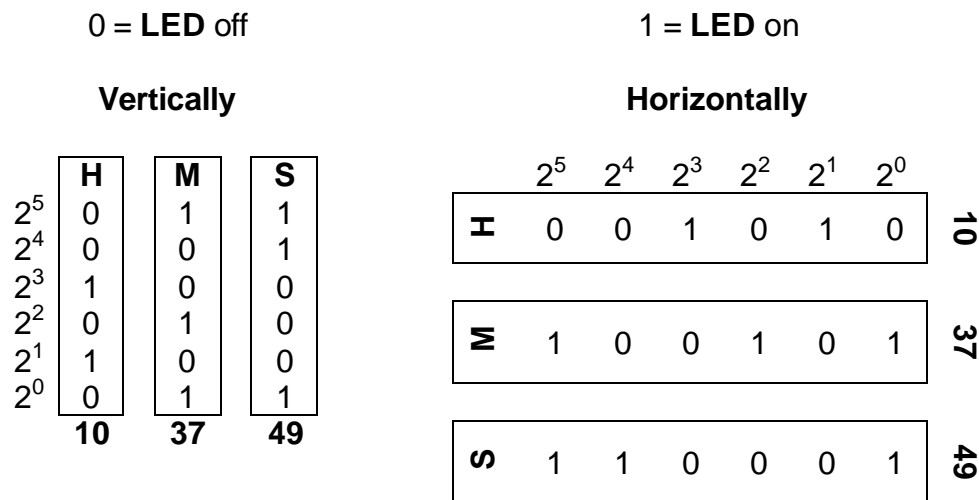
C • Binary Clock

A *binary clock* is a clock which displays traditional sexagesimal time (military format) in a binary format. The most common binary clock uses three columns or three rows of LEDs to represent zeros and ones. Each column (or row) represents a time-unit value.

When three columns are used (vertically), the bottom row in each column represents **1** (or 2^0), with each row above representing higher powers of two, up to 2^5 (or **32**). To read each individual unit (hours, minutes or seconds) in the time, the user adds the values that each illuminated LED represents, and then reads the time from left to right. The first column represents the hour, the next column represents the minute, and the last column represents the second.

When three rows are used (horizontally), the right column in each row represents **1** (or 2^0), with each column left representing higher powers of two, up to 2^5 (or **32**). To read each individual unit (hours, minutes or seconds) in the time, the user adds the values that each illuminated LED represents, and then reads the time from top to bottom. The top row represents the hour, the next row represents the minute, and the bottom row represents the second.

For example:



Time is: **10 : 37 : 49**

For this problem you will read a time in sexagesimal time format, and output both the vertical and horizontal binary clock values. The output will be formed by concatenating together the bits in each column (or row) to form two 18 character strings of **1**'s and **0**'s as shown below.



Greater New York
Programming Contest
St. Joseph's College
Patchogue, NY



10:37:49 would be written vertically as **011001100010100011** and horizontally as **001010100101110001**.

Input

The first line of input contains a single integer N , ($1 \leq N \leq 1000$) which is the number of data sets that follow. Each data set consists of a single line of input containing the time in sexagesimal format.

Output

For each data set, you should generate one line of output with the following values: The data set number as a decimal integer (start counting at one), a space, the binary time in vertical format (18 binary digits), a space and the binary time in horizontal format (18 binary digits).

Sample Input	Sample Output
2	1 011001100010100011 001010100101110001
10:37:49	2 0000000000000000001 000000000000000001
00:00:01	



E • Text Messaging Improvement?

On a standard mobile phone the letters are distributed across the keys 2 through 9 as:

	2 ABC	3 DEF
4 GHI	5 JKL	6 MNO
7 PQRS	8 TUV	9 WXYZ

To enter the letter **C**, you press key 2 three times (seeing **A-B-C**). The number of keystrokes to enter a letter depends on where it is in the list of letters on its key.

The *Flathead Telephone Company* (FTC) is considering rearranging the letters on the keys to reduce the average number of keystrokes required to enter names etc. or send text messages. The letters must still appear in alphabetical order on the keys but different numbers of letters may appear on each key and possibly more keys could be used. FTC has several databases of letter frequencies used in different applications. For instance, it might help to move **S** from the **7** key to the **8** key. They need a program which is given the frequencies of the letters and a number of keys and returns the assignment of letters to keys with the smallest average number of keystrokes using the given frequencies. Each key used must have at least one letter and at most eight letters.

Input

The first line of input contains a single integer N , ($1 \leq N \leq 1000$) which is the number of data sets that follow. Each data set consists of three lines of input. The first line contains a single integer K , ($4 \leq K \leq 26$), the number of keys which are to be used. The second and third lines contain 13 decimal values each giving the percent frequency of the letters **A** through **Z** in order.

Output

For each data set, you should generate one line of output with the following values: The data set number as a decimal integer (start counting at one), the best average number of keystrokes to three decimal places, a space and the letters **A** through **Z**, for the best arrangement, in order with a single



Greater New York
Programming Contest
St. Joseph's College
Patchogue, NY

event sponsors

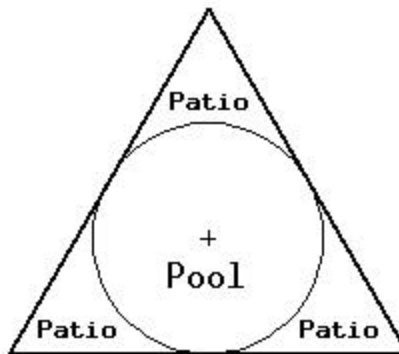


space at the break between letters on different keys. It is possible that the same input data set may produce different output.

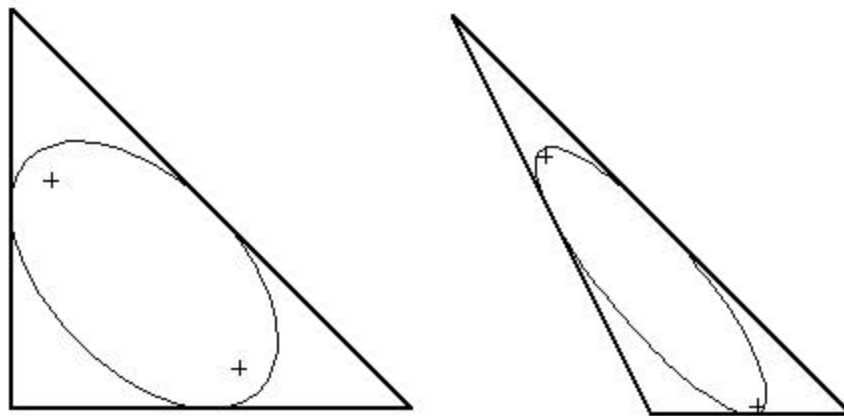
Sample Input	
2	
8	
8.167	1.492 2.782 4.253 12.702 2.228 2.015 6.094 6.966 0.153 0.772 4.025 2.406
6.749	7.507 1.929 0.095 5.987 6.327 9.056 2.758 0.978 2.360 0.150 1.974 0.075
9	
1.0	10.0 11.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
10.0	10.0 10.0 10.0 10.0 10.0 11.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
Sample Output	
1	1.647 AB CD EFG HIJK LM NOPQ RS TUVWXYZ
2	1.570 A B CDEFG HIJKLM N OP QR STUV WXYZ

I • Joe's Triangular Gardens

Joe's landscaping company specializes in gardens for computer geeks who have just had their company go public. One of his signature features is a round pool surrounded by a tiled patio in the form of an equilateral triangle where the edge of the pool is tangent to each side of the triangle at its midpoint.



Unfortunately, some of Joe's customers are not satisfied with an equilateral triangle, usually in the center of the garden. Some want it in a corner or next to a slope or some other layout. Joe would like the option of offering arbitrary triangular patios with an elliptical pool which is tangent to each side at the center of the side. For example:



Joe knows how to draw an ellipse by putting two stakes in the ground (at the foci of the ellipse), tying a rope between them and dragging a marker stick inside the rope. What Joe would like is for the customer to determine where the corners of the triangle will be and then measure the location of the triangle vertices and compute where to put the stakes and how long to make the rope.

Write a program, which takes as input the three vertices (x_1, y_1) , (x_2, y_2) and (x_3, y_3) of a triangle and computes an ellipse inscribed in the triangle, which is tangent to each side of the triangle at its



midpoint. The output is the coordinates of the two foci of the ellipse and the length of the rope (which is the sum of the distances from the foci to any point on the ellipse).

Input

The first line of input contains a single integer N , ($1 \leq N \leq 1000$) which is the number of data sets that follow. Each data set consists of a single line of input containing 6 space separated floating point numbers $x1$ $y1$ $x2$ $y2$ $x3$ $y3$ giving the coordinates of the vertices of a triangle.

Output

For each data set, you should generate one line of output with the following values: The data set number as a decimal integer (start counting at one), a space and five floating point values accurate to two decimal places each separated by a single space. The values are $fx1$ $fy1$ $fx2$ $fy2$ rl where $(fx1, fy1)$ is one focus of the ellipse, $(fx2, fy2)$ is the other focus of the ellipse and rl is the sum of the distances from the foci to any point on the ellipse (e.g. the length of the rope). The foci should be listed in *increasing lexicographical order* (i.e. $fx1 \leq fx2$ and if $fx1 = fx2$, $fy1 \leq fy2$). Note that in the case the ellipse is a circle, the two foci are the same (e.g. the center of the circle).

Sample Input	Sample Output
3	1 200.00 157.71 200.00 157.76 115.47
100 100 200 273.2051 300 100	2 119.53 213.81 213.81 119.53 163.30
100 100 100 300 300 100	3 103.94 253.14 229.40 146.86 170.51
100 200 100 300 300 100	