The 37th ACM-ICPC Asia Hangzhou Regional Contest --- Online Contest

Problem List

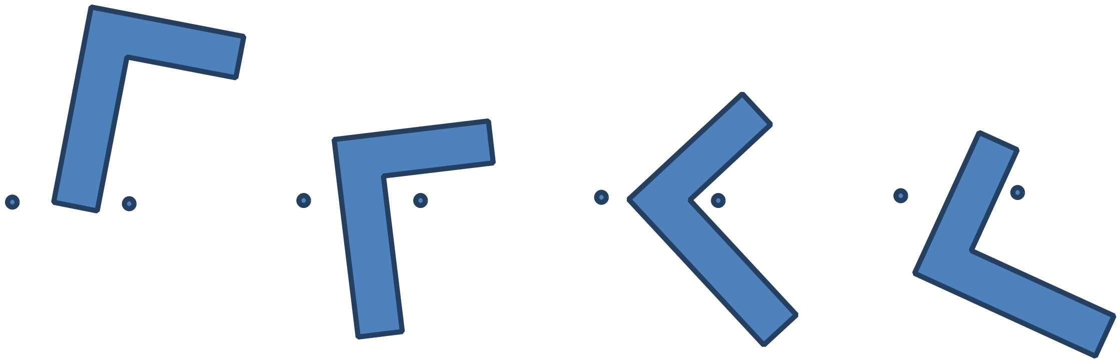
Problem A. Boomerang

Description

Australia original inhabitants used to hunt by a weapon called "boomerang". When a boomerang is thrown out, it rotates, hits the target, and then return to the thrower. ZXX is the best boomerang thrower in Australia. His skill is so wonderful that his boomerang can do any rotation he wants in the air. He travels around to show his skill and make money. One of his classic show is to throw out the boomerang, and it will pass through between two very close pillars. Of course the boomerang must fly parallel to the ground. If not so, everybody can do it. ZXX always puts the two pillars as close as possible to show his skill, but he wants you to figure out the smallest distance between two pillars which allows his boomerang to go through.

To simplify the problem, you can consider the boomerang as a simple polygon in a 2D plane, and each of its edge is parallel to x-axis or y-axis. Each interior angle is either 90 degrees or 270 degrees. Two pillars can be considered as two points.

This illustration simply shows how a boomerang passes through two pillars:



Input

The input consists of several test cases (less than 500).Each test case begins with an integer n (4<=n<=8), representing the number of vertices of a polygon. Next n lines give coordinates of n vertices in order. Each line contains two integers x_i, y_i($|x_i|$, $|y_i|$<=100000). For each test case there are no two vertices is in the same place.

Input ends with n=0.

Output

For each test case, print a single real number w, representing the minimum distance to ensure the boomerang can pass through. The answer should be rounded to 2 digits after the decimal point.

Sample Input

```
6
0 0
100 0
100 1
1 1
1 100
0 100
8
0 0
30 0
30 30
20 30
20 10
10 10
10 30
0 30
0
```

Sample Output

```
1.41
14.14
```

Problem B. Arrest

Description

There are (N+1) cities on TAT island. City 0 is where police headquarter located. The economy of other cities numbered from 1 to N ruined these years because they are all controlled by mafia. The police plan to catch all the mafia gangs in these N cities all over the year, and they want to succeed in a single mission. They figure out that every city except city 0 lives a mafia gang, and these gangs have a simple urgent message network: if the gang in city i (i>1) is captured, it will send an urgent message to the gang in city i-1 and the gang in city i -1 will get the message immediately.

The mission must be carried out very carefully. Once a gang received an urgent message, the mission will be claimed failed.

You are given the map of TAT island which is an undirected graph. The node on the graph represents a city, and the weighted edge represents a road between two cities(the weight means the length). Police headquarter has sent k squads to arrest all the mafia gangs in the rest N cities. When a squad passes a city, it can choose to arrest the gang in the city or to do nothing. These squads should return to city 0 after the arrest mission.

You should ensure the mission to be successful, and then minimize the total length of these squads traveled.

Input

There are multiple test cases.

Each test case begins with a line with three integers N, M and k, here M is the number of roads among N+1 cities. Then, there are M lines. Each of these lines contains three integers X, Y, Len, which represents a Len kilometer road between city X and city Y. Those cities including city 0 are connected.

The input is ended by “0 0 0”.

Restrictions: 1 ≤ N ≤ 100, 1 ≤ M ≤ 4000, 1 ≤ k ≤ 25, 0 ≤ Len ≤ 1000

Output

For each test case,output a single line with a single integer that represents the minimum total length of these squads traveled.

Sample Input

```
3 4 2
0 1 3
0 2 4
1 3 2
2 3 2
0 0 0
```

Sample Output

```
14
```

Problem C. Sky Soldiers

Description

An airplane carried k soldiers with parachute is plan to let these soldiers jump off the plane along a straight air route. The landing point of these soldiers is undetermined because of various weather conditions. However, the statisticians of the army can analysis the probability of landing in a certain place through landing history records. To make it simple, the statistician suggests that these sky soldiers will land on finite discrete points of a straight line.

This mission plans to place m provisions for the soldiers on the line for landing. These soldiers will be informed the direction of the nearest provision point by a special device after landing, and then walk to the point. The manager of this mission is asking you for help: to determine m points for provisions so that the expected sum of walking distance should be minimized. You can put provisions on any point of the landing line.

Input

There are multiple test cases. For each case, the first line contains two integers k and m (1 ≤ k ≤ 1,000, 1 ≤ m ≤ 50), which represent the number of sky soldiers and the number of positions to place provisions separately.

The following k lines contain descriptions of landing parameters for the soldiers numbered from 1 to k. Each description consists of an integer L followed by L pairs of (x, p), which indicates that the probability of the soldier's landing on integer coordination x is p. It is guaranteed that all the p values are positive real numbers, and the sum of p in a single line is exactly 1. The same x may appear more than once on the same line which you should simply add up all the probability p of the pairs with equal x.
The number of places on which all the soldiers could land is no more than 1000 and it can not be less than m.
The input ends with k=m=0.

Output

For each test case, output a line containing only one real number which indicates the minimum expected sum of distance these soldiers will move and should be rounded to two digits after the decimal point.

Sample Input

```
2 1
2 0 0.5 1 0.5
```

```
2 1 0.1 3 0.9
0 0
```

Sample Output

```
2.30
```

Problem D. Logical Expression

Description

Happy birthday to you!

Your professor gives you a strange device as your birthday gift. But your experience tells you, it is a trap.

Just as expected, he smirks, and then says slowly: "This device is my new invention. Only you, my most brilliant student, deserve to have it."

"Thanks, I will keep it up." You reply.

"However," silent for a while, he starts again: "unfortunately, I have lost its original circuit diagram. So I need you to do some researches on it and use a simple logical expression to describe its characteristics. You won't let me down, will you?"

He laughs, pats you on the back, then turns and stalks off, leaves you, the poor guy, with the "gift" on your hands.

You collect yourself, and start looking at the "lovely" thing. It is a black box, with n inputs labeled from the first capital letter (A) to the n-th capital letter, as well as one unique output. All the inputs and the output are one-bit binary numbers (either 0 or 1).

Then you sit down, put the device on the desk, take out the tools, examine all possible combinations of input values and record the output on the paper.

After everything gets ready, now it is just a piece of cake for you to give a correct logical expression for it.

A correct logical expression should be:

1. A **not null** string only contains '+', '-' and capital letters from the first (A) to the n-th;
2. A single capital letter in range is a legal logical expression, and its output equals to the value of the input labeled by this letter;
3. If E is a legal logical expression, then NOT operation -E is a legal logical expression;
4. If E and F are legal logical expressions, then AND operation EF is a legal logical expression;

5. If E and F are legal logical expressions, then OR operation E+F is a legal logical expression;
6. In a legal logical expression, you should first do NOT operations from the left to the right, then AND operations from the left to the right, and finally the OR operations from the left to the right;
7. A logical expression is correct if and only if with any possible combinations of input values, its output is the same as which the device outputs.

But as a down-to-earth idealist, you are never satisfied with this. You want to give the most beautiful logical expression like this:

1. It is a correct logical expression;
2. It is the shortest one (**not null**) among all correct logical expressions;
3. If there are still many available logical expressions, choose the lexicographically least one.

Input

There are several test cases in the input.

Each test case begins with one integer n (1 <= n <= 5), indicating the number of inputs of the device.
Then 2^n lines follow. Each line contains (n + 1) one-bit binary numbers, separated by spaces. The first n numbers indicate the value of the n inputs in order, and the last one is the corresponding output.
The 2^n sets of inputs are different with each other.
The input ends with n = 0.

Output

For each test case, output only one line -- the most beautiful logical expression.

Sample Input

```
3
0 0 0 0
0 0 1 1
0 1 0 1
0 1 1 0
1 0 0 0
1 0 1 1
1 1 0 0
1 1 1 1
0
```

Sample Output

```
-A-CB+-BC+AC
```

Hint

In ASCII table, '+' < '-' < 'A' < 'B' < 'C' < 'D' < 'E'.

Problem E. Finding crosses

Description

The Nazca Lines are a series of ancient geoglyphs located in the Nazca Desert in southern Peru. They were designated as a UNESCO World Heritage Site in 1994. The high, arid plateau stretches more than 80 kilometres (50 mi) between the towns of Nazca and Palpa on the Pampas de Jumana about 400 km south of Lima. Although some local geoglyphs resemble Paracas motifs, scholars believe the Nazca Lines were created by the Nazca culture between 400 and 650 AD.[1] The hundreds of individual figures range in complexity from simple lines to stylized hummingbirds, spiders, monkeys, fish, sharks, orcas, llamas, and lizards.

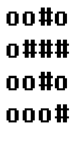
Above is the description of Nazca Lines from Wikipedia. Recently scientists found out that those lines form many crosses. Do those crosses have something to do with the Christian religion? Scientists are curious about this. But at first, they want to figure out how many crosses are there. So they took a huge picture of Nazca area from the satellite, and they need you to write a program to count the crosses in the picture.

To simplify the problem, we assume that the picture is an N*N matrix made up of 'o' and '#', and some '#' can form a cross. Here we call three or more consecutive '#' (horizontal or vertical) as a "segment".

The definition of a cross of width M is like this:

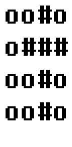
1) It's made up of a horizontal segment of length M and a vertical segment of length M.
2) The horizontal segment and the vertical segment overlap at their centers.
3) A cross must not have any adjacent '#'.
4) A cross's width is definitely odd and at least 3, so the above mentioned "centers" can't be ambiguous.

For example, there is a cross of width 3 in figure 1 and there are no cross in figure 2 ,3 and 4.



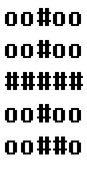
```
oo#o
o###
oo#o
ooo#
```

figure 1



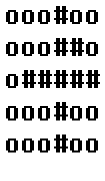
```
oo#o
o###
oo#o
oo#o
```

figure 2



```
oo#oo
oo#oo
#####
oo#oo
oo##o
```

figure 3



```
ooo#oo
ooo##o
o#####
ooo#oo
ooo#oo
```

figure 4

You may think you find a cross in the top 3 lines in figure 2.But it's not true because the cross you find has a adjacent '#' in the 4th line, so it can't be called a "cross". There is no cross in figure 3 and figure 4 because of the same reason.

Input

There are several test cases.
In each test case:
The First line is a integer N, meaning that the picture is a N * N matrix (3<=N<=50) .
Next N line is the matrix.
The input end with N = 0

output

For each test case, output the number of crosses you find in a line.

Sample Input

```
4
oo#o
o###
oo#o
ooo#
4
oo#o
o###
oo#o
oo#o
5
oo#oo
oo#oo
#####
oo#oo
oo##o
6
ooo#oo
ooo##o
o#####
ooo#oo
ooo#oo
oooooo
0
```

Sample Output

```
1
0
0
0
```

Problem F. Assassin’s Creed

Description

Ezio Auditore is a great master as an assassin. Now he has prowled in the enemies’ base successfully. He finds that the only weapon he can use is his cuff sword and the sword has durability m. There are n enemies he wants to kill and killing each enemy needs Ai durability. Every time Ezio kills an enemy he can use the enemy’s sword to kill any other Bi enemies without wasting his cuff sword’s durability. Then the enemy’s sword will break. As a master, Ezio always want to do things perfectly. He decides to kill as many enemies as he can using the minimum durability cost.

Input

The first line contains an integer T, the number of test cases.
For each test case:
The first line contains two integers, above mentioned n and m (1<=n<=10^5, 1<=m<=10^9).
Next n lines, each line contains two integers Ai, Bi. (0<=Ai<=10^9, 0<=Bi<=10).

output

For each case, output "Case X: " (X is the case number starting from 1) followed by the number of the enemies Ezio can kill and the minimum durability cost.

Sample Input

```
2
3 5
4 1
5 1
7 7
2 1
2 2
4 0
```

Sample Output

```
Case 1: 3 4
Case 2: 0 0
```

Problem G. Good Article Good sentence

Description

In middle school, teachers used to encourage us to pick up pretty sentences so that we could apply those sentences in our own articles. One of my classmates ZengXiao Xian, wanted to get sentences which are different from that of others, because he thought the distinct pretty sentences might benefit him a lot to get a high score in his article.

Assume that all of the sentences came from some articles. ZengXiao Xian intended to pick from Article A. The number of his classmates is n. The i-th classmate picked from Article Bi. Now ZengXiao Xian wants to know how many different sentences she could pick from Article A which don't belong to either of her classmates' Article. To simplify the problem, ZengXiao Xian wants to know how many different strings, which is the substring of string A, but is not substring of either of string Bi. Of course, you will help him, won't you?

Input

The first line contains an integer T, the number of test data.

For each test data

The first line contains an integer meaning the number of classmates.

The second line is the string A;The next n lines,the ith line input string Bi.

The length of the string A does not exceed 100,000 characters , The sum of total length of all strings Bi does not exceed 100,000.and assume all string consist only lowercase characters 'a' to 'z'

output

For each case, print the case number and the number of substrings that ZengXiao Xian can find.

Sample Input

```
3
2
abab
ab
ba
1
aaa
bbb
2
aaaa
aa
```

```
aaa
```

Sample Output

```
Case 1: 3
Case 2: 3
Case 3: 1
```

Problem H. Super Mario

Description

Mario is world-famous plumber. His “burly” figure and amazing jumping ability reminded in our memory. Now the poor princess is in trouble again and Mario needs to save his lover. We regard the road to the boss’s castle as a line (the length is n), on every integer point i there is a brick on height hi. Now the question is how many bricks in [L, R] Mario can hit if the maximal height he can jump is H.

Input

The first line follows an integer T, the number of test data.

For each test data:

The first line contains two integers n, m (1 <= n <=10^5, 1 <= m <= 10^5), n is the length of the road, m is the number of queries.

Next line contains n integers, the height of each brick, the range is [0, 1000000000].

Next m lines, each line contains three integers L, R,H.(0 <= L <= R < n 0 <= H <= 1000000000.)

Output

For each case, output "Case X: " (X is the case number starting from 1) followed by m lines, each line contains an integer. The ith integer is the number of bricks Mario can hit for the ith query.

Sample Input

```
1
10 10
0 5 2 7 5 4 3 8 7 7
2 8 6
3 5 0
1 3 1
1 9 4
0 1 0
3 5 5
5 5 1
4 6 3
1 5 7
5 7 3
```

Sample Output

```
Case 1:
4
```

```
0
0
3
1
2
0
1
5
1
```

Problem I. Time travel

Description

Agent K is one of the greatest agents in a secret organization called Men in Black. Once he needs to finish a mission by traveling through time with the Time machine. The Time machine can take agent K to some point (0 to n-1) on the timeline and when he gets to the end of the time line he will come back (For example, there are 4 time points, agent K will go in this way 0, 1, 2, 3, 2, 1, 0, 1, 2, 3, 2, 1, …). But when agent K gets into the Time machine he finds it has broken, which make the Time machine can't stop (Damn it!). Fortunately, the time machine may get recovery and stop for a few minutes when agent K arrives at a time point, if the time point he just arrive is his destination, he'll go and finish his mission, or the Time machine will break again. The Time machine has probability Pk% to recover after passing k time points and k can be no more than M. We guarantee the sum of Pk is 100 (Sum(Pk) (1 <= k <= M)==100). Now we know agent K will appear at the point X(D is the direction of the Time machine: 0 represents going from the start of the timeline to the end, on the contrary 1 represents going from the end. If x is the start or the end point of the time line D will be -1. Agent K want to know the expectation of the amount of the time point he need to pass before he arrive at the point Y to finish his mission.
If finishing his mission is impossible output "Impossible !" (no quotes)instead.

Input

There is an integer T (T <= 20) indicating the cases you have to solve. The first line of each test case are five integers N, M, Y, X .D (0< N,M <= 100, 0 <=X ,Y < 100). The following M non-negative integers represent Pk in percentile.

Output

For each possible scenario, output a floating number with 2 digits after decimal point
If finishing his mission is impossible output one line "Impossible !"
(no quotes)instead.

Sample Input

```
2
4 2 0 1 0
50 50
4 1 0 2 1
100
```

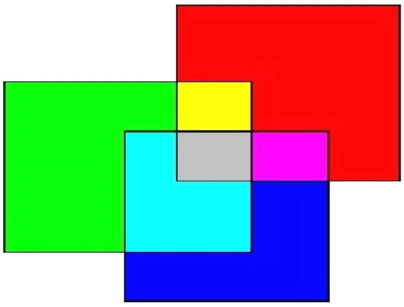
Sample Output

```
8.14
2.00
```

Problem J. Colourful Rectangle

Description

We use Red, Green and Blue to make new colours. See the picture below:



Now give you n rectangles, the colour of them is red or green or blue. You have calculate the area of 7 different colour. (Note: A region may be covered by same colour several times, but it’s final colour depends on the kinds of different colour)

Input

The first line is an integer T(T <= 10), the number of test cases. The first line of each case contains a integer n (0 < n <= 10000), the number of rectangles. Then n lines follows. Each line start with a letter C(R means Red, G means Green, B means Blue) and four integers x1, y1, x2, y2(0 <= x1 < x2 < 10^9, 0 <= y1 < y2 < 10^9), the left-bottom's coordinate and the right-top's coordinate of a rectangle.

Output

For each case, output a line "Case a:", a is the case number starting from 1,then 7 lines, each line contain a integer, the area of each colour. (Note: You should print the areas as the order: R, G, B, RG, RB, GB, RGB).

Sample Input

```
3
2
R 0 0 2 2
G 1 1 3 3
3
R 0 0 4 4
G 2 0 6 4
B 0 2 6 6
```

```
3
G 2 0 3 8
G 1 0 6 1
B 4 2 7 7
```

Sample Output

```
Case 1:
3
3
0
1
0
0
0
Case 2:
4
4
12
4
4
4
4
Case 3:
0
12
15
0
0
0
0
```