Problem A
Partial Fractions
Time limit: 2 seconds

Decomposing rational numbers is an important technique in mathematics. Consider the rational number r/pq where r, p and q are positive integers and $r < pq$. You can always find integral solution x and y for the following equation

$$\frac{r}{pq} = \frac{x}{p} + \frac{y}{q}$$

regardless what the value of r is as long as p and q are relatively prime. For example, take $p = 3$, $q = 5$ and $r = 7$. Then you can check that $7/15 = 2/3 + (-1)/5$. The solution is not unique. For example, $(-1)/3 + 4/5 = 7/15$ is a second solution. In fact, there are infinitely many solutions for this kind of indefinite equation. You are asked to find a solution such that $|x| + |y|$ is minimized.

Input Format

The test data file contains many test cases. The first line gives you the number of test cases, which is no more than 10. From the second line, each line specifies one test case, which contains three space-delimited positive integers r, p and q. Notice that p and q are relatively prime and $r < pq$ and $r < 10^9$. Also, assume that $1 < p, q < 10^9$.

Output Format

Find a solution of x and y for the specified indefinite equation such that the sum of the absolute values of x and y is minimized. Output the value of $|x| + |y|$ in one line.

Sample Input

```
3
2 3 5
4 4 3
13 5 7
```

Sample Output for the Sample Input

```
2
1
5
```

Problem B
The magic square
Time limit: 2 seconds

The magic square is a traditional problem that has attracted the attention of researchers for many years and has essentially many interesting variants, one of which is defined as follows:

> Given a set of numbers, $\{1, 2, \ldots, n^2\}$ where n is odd and $n \leq 5$, the goal is to assign the set of numbers to a square array in such a way that (1) the sum of each row and column and (2) the sum of the two main diagonals are the same.

For example, for the following magic square example,

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

the sum of each row and column and the sum of the two main diagonals are all 15.

Input Format

Since there are many ways to assign the numbers to the square array, to ensure that the magic square generated is what we are looking for, the input contains two parts: (1) the order of the magic square n, and (2) the numbers assigned to the first row of the square array, as the following sample inputs show. Note that '-1' indicates "don't care." Note that the program will terminate when $n = 0$.

Sample Input

```
3 6 7 2
0
```

Output for the Sample Input

```
6 7 2
1 5 9
8 3 4
- - -
```

Problem C
eBike Battery Drain
Time limit: 2 seconds

The NCPC delivery service employees many retirees, whom all ride company e-bikes on fixed routes to deliver goods between different service stations. There is always one and only one route (could be direct or passing through one or more service points) between any two service stations. And the company requires e-bikes travel on those company approved fixed routes only. Riding through a road that directly connects two service stations will drain a certain amount of battery power. Since the road condition do change from time to time, the company updates the drainage amount regularly.

Given a day of initial and updated battery drainage on any road, please help the NCPC company determines the maximum battery power drain at a given time for traveling between two possible service stations.

Input File Format

The first line of input is an integer indicating the number of test cases to follow, which is at most 10. For each test case, the first line contains 2 integers, n, q, where $n \leq 1,500$ and $q \leq 20$, indicating there are n service stations (numbered from 1 to n), and q successive battery power drainage updates.

The next $n-1$ lines each contains three positive integers, i, j, w, indicating there is a direct road connecting between service stations i and j and that it would drain $w \leq 10,000$ unit of battery if traveled on that road. The road are numbered from 1 to $n-1$ in the order of the input.

The next q lines each contains two positive integers, r, w, indicating that road r's battery drainage is updated to w.

Output Format

For each test case and for each update, print out the maximum battery drain for delivery between any two possible service stations after that update.

Sample Input

```
1
4 3
1 2 10
2 3 100
2 4 100
3 150
2 200
3 9
```

Sample Output for the Sample Input

```
250
350
210
```

Problem D
Mysterious Message
Time limit: 1 second

The explorer Diego Márquez discovered an unexplored cave in South Asia. He found interesting writings on a wall.

BNN⋆AAA

Diego shortly realized that these symbols were produced from a popular word guessing game in that area during the years 1838-1842, called *mysterious transformation.*

The mysterious transformation receives a sequence of length N, consisting of $N-1$ capital English letters (A-Z) and a star symbol $\star$ representing the end of the sequence. Given a sequence of length N, we create $N-1$ other sequences where each of these $N-1$ sequences is a cyclic right shift of the original sequence. Next, we sort the N sequences in lexicographic order. The message Diego found is the sequence that takes the last letter of each sorted, cyclically shifted sequence.

For example, suppose the original sequence is BANANA*, which is the answer that Diego wants to find for the instance 'BNN⋆AAA', the mysterious transformation creates the following sequences:
BANANA⋆
⋆BANANA
A⋆BANAN
NA⋆BANA
ANA⋆BAN
NANA⋆BA
ANANA⋆B

Then we sort the sequences in lexicographic order:
ANANA⋆B
ANA⋆BAN
A⋆BANAN
BANANA⋆
NANA⋆BA
NA⋆BANA
⋆BANANA

The sequence of last letters is therefore BNN*AAA.

Please write a program to help Diego decode the mysterious message.

Input Format

The test data file may contain many test cases. Each test case contains two lines. The first line contains one integer N ($1 < N < 10^4$), indicating the length of the sequence. The second line contains a sequence of length N. You will find only one $\star$ in each sequence. The last test case is followed by a line containing a single 0.

Output Format

The output for each test case is the decoded message.

Sample Input

```
7
BNN*AAA
5
OSL*O
0
```

Sample Output for the Sample Input

```
BANANA*
SOLO*
```

Problem E
Navigation Problem
Time limit: 2 seconds

Bob is working on a navigation system project. He writes a program to find the shortest route between any two cities, where each city is reachable from any other cities. Bob models the system as an un-directed graph. Suppose there are N cities, indexed from 1 to N. For any two cities i and j, $\ell_{i,j}$ denotes the road length between them. If there is a road between city i and j and there is no other city in between, then $\ell_{i,j}$ is finite; otherwise, $\ell_{i,j}$ is infinite.

Bob computes all-pair of shortest paths with his program and stores the distances in a matrix M. However, after a severe storm that caused a power outage and road system damage, Bob lost some of the data. Luckily, he restored the information stored in the matrix M, but lost the original road length information $\ell_{i,j}$. After a couple of months, the transportation bureau repaired the road system one by one. Some roads between cities may reduce the length, or some new roads are built to connect two cities originally not directly linked! Note that, after the reconstruction, the specific road length does NOT increase! Bob wants to recompute the all-pair of shortest paths, given the data in M and the updated road information. Your task is to write a program that outputs the number of pairs of cities, whose shortest path distance have been shortened for each pair of repaired road. If the repaired road does not change the all-pairs shortest paths, then output 0.

For example, given three cities, let $\ell_{1,2} = 1, \ell_{1,3} = 2$ and $\ell_{2,3} = \infty$. The all-pair shortest path distance matrix M is:

$$\begin{pmatrix} 0 & 1 & 2 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \end{pmatrix}$$

Now if $\ell_{1,3}$ is repaired and reduced to 1 after reconstruction, then the shortest distance matrix M becomes:

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 2 \\ 1 & 2 & 0 \end{pmatrix}$$

So the program should output 2, since there are two pairs with shortest distance reduced. Afterwards, suppose a new road between city 2 and city 3 is constructed with distance 1. Now the distance between any two cities is 1. So the output should be 1, since the shortest distance between city 2 and city 3 is reduced from 2 to 1!

Input File Format

The first line of the input file contains a positive integer $T < 10$, indicating the number of test cases. For each case, the first line gives two integers, $N < 1000$ and $K < 50$, denoting, respectively, the number of cities and the number of roads with length reduced or newly constructed roads after the storm. Then, there are N lines, each containing N integers. The j-th value on the i-th of these lines, i.e., $M_{i,j}$ $(0 < M_{i,j} < 2^{32})$, represents the shortest path length between city i and j before the storm. Then, K more lines follow, and each line contains three integers i, j, d, indicating the road length, between city i and j, becomes d, which is at most $\ell_{i,j}$.

Output Format

For each case and each of the K reduced distances, output the number of pairs of cities that reduce the shortest path. Note the change of road length is accumulative, i.e., the earlier change affects the laters. So for each case, you only need to output the number of pairs with reduced distance compared with the immediately previous case. If the reduction of road length does not change the length of all-pairs of shortest paths, then output 0 in the corresponding line.

Sample Input

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

Sample Output for the Sample Input

```
2
1
3
1
1
```

Problem F
Number Combination Problem
Time limit: 1 second

You are given two positive integers m and n, where $m \leq n$. Please pick some distinct numbers from a sequence $\hat{L} = \{1, 2, 3, \cdots, m\}$ such that the sum of these numbers must be equal to n. Write a program to list all possible combinations of numbers from sequence $\hat{L}$.

Technical Specification

1. There are 10 test cases.
2. $8 \leq n \leq 30$.

Input Format

Each test case includes both m and n. The last test case is followed by a line containing a single 0.

Output Format

The output for each test case is all possible combinations of numbers whose sum is equal to n. For each combination, you need to list the numbers in an increasing order, where each number is separated by space. Besides, you should use a semicolon (;) and a blank to separate different combinations. A combination with fewer numbers should be listed before other combinations with more numbers. When two combinations have the same number of numbers, the one whose last number is smaller should be listed first. If there is still a tie, we then compare the first distinct number of these combinations. The one whose number is smaller is listed first. For example, the combination "1 2 5 6" should be listed before the combination "1 3 4 6".

For different cases, their results should be separated by a blank line. In case that you cannot find any combination, please output -1.

Sample Input

```
3 4
8 10
0
```

Sample Output for the Sample Input

```
1 3

4 6; 3 7; 2 8; 1 4 5; 2 3 5; 1 3 6; 1 2 7; 1 2 3 4
```

Problem G
National Challengingly Proud Curiosity
Time limit: 3 seconds

NCPC (National Challengingly Proud Curiosity) is a robot designed to explore a faraway planet. NCPC can move around and climb-up or go-down by a limited height difference (H) which is called the moving ability. Scientists want to land NCPC at a point such that NCPC can move around freely in an assigned district. Here "freely" means that NCPC can move successfully along any path in the district. These paths consist of only vertical or horizontal edges (no diagonal edges).

Scientists remotely measure all the terrain data of a region. The region is analyzed as a rectangle grid ($N \times M$) where a height datum($d_{i,j}$) is associated with each cell (i, j). Consider a district(a subgrid) with a left-top cell (A, B) and a right-down cell $(min(A+L-1, N-1), min(B+L-1, M-1))$. In order to move "freely", the moving ability has to be larger than or equal to the height difference between neighbor cells. Scientists want you to write a program to calculate the maximum height difference between neighbor cells in an assigned district. In order to check the correctness of your program, scientists produce simulated data as follows. In $N \times M$ grid, $d_{0,0} = S$ and an integer Q are given.
$d_{0,j} = (d_{0,j-1} + floor(R10((j + d_{0,j-1})^{(j+d_{0,j-1}) \bmod Q^2} \bmod Q^2)/Q)) \bmod Q^2$ for $j > 0$
$d_{i,0} = (d_{i-1,0} + floor(R10((i^2 + d_{i-1,0})^{(i^2+d_{i-1,0}) \bmod Q^2} \bmod Q^2)/Q)) \bmod Q^2$ for $i > 0$
$d_{i,j} = (floor(sum/3) + floor(R10((i^2 + j + sum)^{(i^2+j+sum) \bmod Q^2} \bmod Q^2)/Q)) \bmod Q^2$ for $i > 0$ and $j > 0$ where $sum = d_{i-1,j-1} + d_{i-1,j} + d_{i,j-1}$.

Here $floor(x) = max\{z | z \text{ is an integer and } z \leq x.\}$. $x \bmod y$ is the remainder of x divided by y. Given an integer x, we represent x in decimal base as $a_0a_1a_2 \cdots a_w$ with $a_i \in \{0, 1, 2, \cdots, 9\}$ and $a_0 > 0$. $R10(x)$ is the number in decimal base $a_wa_{w-1}a_{w-2} \cdots a_0$. For example, $floor(3.14) = 3$, $5 \bmod 3 = 2$, and $R10(1230) = 0321 = 321$.

For example, let $N = 4$, $M = 5$, $S = 7$, and $Q = 10$. The following table is determined.

$d_{i,j}$	0	1	2	3	4
0	7	13	18	19	26
1	13	16	15	23	25
2	20	22	26	22	27
3	29	24	31	27	25

Given $d_{0,0} = S = 7$, we can obtain the followings. $d_{0,3}$
$= (d_{0,2} + floor(R10((d_{0,2} + 3)^{(d_{0,2}+3) \bmod Q^2} \bmod Q^2)/Q)) \bmod Q^2$
$= (18 + floor(R10(21^{21} \bmod 100)/10))) \bmod 100$
$= (18 + floor(R10(21)/10))) \bmod 100$
$= (18 + 1) \bmod 100$
$= 19$

$d_{1,0} = (d_{0,0} + floor(R10((d_{0,0}+1)^{(d_{0,0}+1) \bmod Q^2} \bmod Q^2)/Q)) \bmod Q^2$
$= (7 + floor(R10(8^8 \bmod 100)/10))) \bmod 100$
$= (7 + floor(R10(16)/10))) \bmod 100$
$= (7 + 6) \bmod 100$
$= 13$

For $d_{2,2}$, $sum = d_{1,1} + d_{1,2} + d_{2,1} = 16 + 15 + 22 = 53$. $d_{2,2}$
$= (floor(53/3) + floor(R10((53 + 2^2 + 2)^{(53+2^2+2) \bmod Q^2} \bmod Q^2)/Q)) \bmod Q^2$
$= (17 + floor(R10((59^{59} \bmod 100)/10)) \bmod 100$
$= (17 + floor(R10(39)/10)) \bmod 100$
$= (17 + 9) \bmod 100$
$= 26$

If $A = 0, B = 0, L = 2$, then height data of the district are $7, 13, 13, 16$, and the height differences between neighbor cells are $6, 6, 3, 3$, and the maximum height difference is $13 - 7 = 6$.

Technical Specification

1. $0 < N, M, L, C, S < 2020$
2. $0 < Q \leq 1024$
3. $0 \leq A < N$
4. $0 \leq B < M$

Input Format

The first line contains 4 integers N, M, S, Q. The second line contains an integer C. For the following C lines, each line contains 3 integers A, B, L. N, M, S, Q construct the simulated grid, and you have to consider the district with left-top cell (A, B) and right-down cell $(min(A + L - 1, N - 1), min(B + L - 1, M - 1))$.

Output Format

For each A, B, L, please output the maximum height difference between neighbor cells of the district in one line.

Sample Input

```
4 5 7 10
4
0 0 2
0 0 3
1 3 2
3 3 5
```

Sample Output for the Sample Input

```
6
11
5
2
```