Problem A. Bank Transfer

Time limit:	1 second
Memory limit:	512 megabytes

Tanya has an account in "Redgotts" bank. The bank has the commission to transfer money to "Bluegotts" bank that her friend Vanya has the account in.

Tanya has read her bank rules and learned the following:

The commission for the bank transfer is 25 *tugriks plus* 1% *of the sum transferred. However, the commission is never smaller than* 100 *tugriks, and cannot exceed* 2000 *tugriks.*

Tanya is planning to transfer k tugriks to Vanya. Help her find out what the commission would be.

Input

Input is a single integer k ($200 \le k \le 10^9$).

Output

Output one floating point value: the commission for the transfer. The value must be printed with at least two exact digits after the decimal point.

Examples

standard input	standard output
20210	227.10
9000	115.00
300000	2000.00

Problem B. Bacteria

Time limit: 1 second
Memory limit: 512 megabytes

The Berland University of Biology (BUB) studies bacteria. It is known that the behaviour of a bacterium is determined by the structure of its DNA. In this problem, we assume that the bacterial DNA is a string consisting of zeros and ones.

Recently, scientists at the BUB have discovered a new type of bacteria. Its main feature is that when a bacterium divides, its DNA does not replicate, but divides into two halves. More precisely, let's suppose that the DNA of the original bacterium is a string $S = s_1 s_2 \ldots s_k$ of even length k (s_i denotes the i-th character of the string S and is equal to either 0 or 1). Then, after the division there are two bacteria with DNA equal to $s_1 s_2 \ldots s_{\frac{k}{2}}$ and $s_{\frac{k}{2}+1} \ldots s_{k-1} s_k$, respectively.

For the experiment, the scientists are planning to take a bacterium with DNA of length 2^n. The experiment consists of $n+1$ steps. At the end of each step, except for the last one, each currently available bacterium divides. So, in the first step, there will be only one bacterium with DNA of length 2^n, in the second — two bacteria with DNA of length 2^{n-1} each, and so on. Finally, in the $n+1$-st step, there will be 2^n bacteria, each of them will have only one character in its DNA.

Of course, it is not interesting to study bacteria with the same DNA. Determine what the DNA that the first bacterium should have, so that there are as many different types of DNA as possible obtained during the experiment.

Input

The first line contains one integer n ($1 \le n \le 20$) denoting that the DNA of first bacterium should have the length of 2^n.

Output

Print one string of characters 0 and 1, its length must be equal to 2^n — the DNA of the first bacterium, so that the number of different DNA during the experiment is greatest possible. If there are many possible answers, print any of them.

Example

standard input	standard output
3	00100111

Note

There will be 9 different DNA to appear during the experiment in the first sample test: 00100111, 0010, 0111, 00, 10, 01, 11, 0 and 1.

Problem C. Check Markers

Time limit:	2 seconds
Memory limit:	512 megabytes

Alexander Markovich should start his lecture in five minutes, but at this moment he is just entering the university! He would be just in time unless it wasn't that the lecture had been scheduled in that huge room where lots of spent markers are all around. Now Alexander Markovich should find at least two markers of different colors that are not completely spent yet.

University professors use markers of n different colors, and they are all in a single pile initially. We know that among the markers of color i there are a_i spent and b_i good (that can still be used for writing) markers in the pile. It is impossible to distinguish whether a marker is spent or good by its appearance. In order to find two good markers of different colors, Alexander Markovich will repeat the following procedure:

1. he takes two markers of different colors from the pile;
2. then he simultaneously checks whether each of these markers can be used for writing;
3. if both markers are good, Alexander Markovich takes them and starts the lecture;
4. otherwise, if at least one of the markers is spent, he throws both markers to trash and returns to step 1.

Alexander Markovich chooses a pair of markers arbitrarily. Is it possible that he never finds two good markers of different colors, i. e. at some iteration of step 1 there are no two markers of different colors left in the pile?

You are to solve the problem for t test cases.

Input

The first line contains a single integer t ($1 \le t \le 2 \cdot 10^5$) — the number of test cases.

Each of the test cases is described in three lines. The first line contains a single integer n ($2 \le n \le 2 \cdot 10^5$) — the number of different colors of markers.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 10^9$) — the number of spent markers of each color.

The third line of each test case contains n integers $b_1, b_2, \ldots, b_n$ ($0 \le b_i \le 10^9$) — the number of good markers of each color.

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case print "`YES`" in a separate line if it is possible that Alexander Markovich never finds two good markers of different colors, otherwise print "`NO`".

Example

<table>
<tr><th>standard input</th><th>standard output</th></tr>
<tr><td><pre>3
3
1 2 1
2 1 1
2
1 1
2 2
4
1 1 1 1
2 1 2 1</pre></td><td><pre>YES
NO
YES</pre></td></tr>
</table>

Problem D. Multiple Subject Lessons

Time limit:	1 second
Memory limit:	512 megabytes

Kate's school has introduced multiple subject lessons.

The students have the following task for the lesson of Math, Art and Sociology in the seventh grade. They are given an integer n. Each student has a set of k colored pencils, let the colors be numbered from 1 to k. Each student takes a sheet of paper and writes down one or several integers on it, so that their sum was equal to n. Each integer is written using one of the pencils, so it has one of the k possible colors.

The students must agree to do the task in such way that there are no two students with the same solution. Two solutions are the same if for each integer a and each color i the number of integers a of color i at student's sheets is the same.

The teacher of Math is sure that the students will be able to complete the task. However, she wants to know how many solutions are there, maybe there are not enough for all the students to have different solutions. Help her to find that out!

Input

The input contains two integers n and k ($1 \le n, k \le 15$).

Output

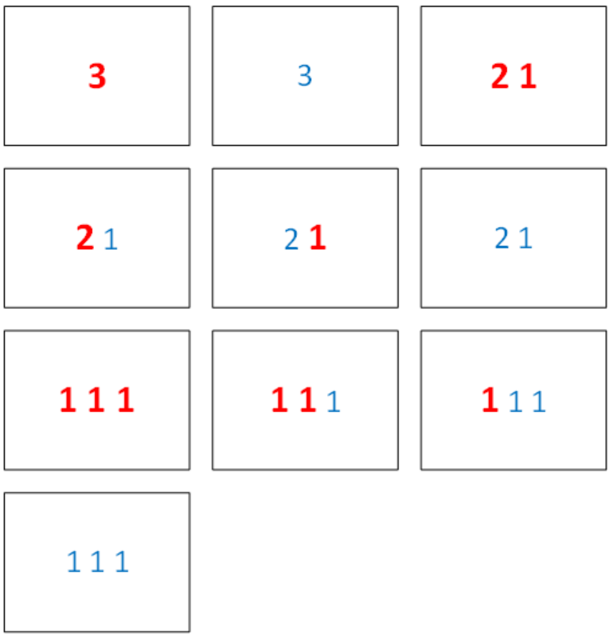
Print one integer — the number of solutions to the task.

Example

standard input	standard output
3 2	10

Note

The following picture shows all possible ways to solve the task in the first sample test. Note that the order of integers written doesn't matter, only the number of integers written with each color.



Problem E. Prank at IKEA

Time limit: 2 seconds
Memory limit: 512 megabytes

Popular video blogger Pasha has decided to make a prank video in order to increase his subscribers count. He has chosen a well-known furniture store IKEA as a shooting location.

One of the store halls can be represented as a grid of size $n \times m$. Some grid cells are occupied by couches. Currently all the couches are folded, so each couch occupies exactly two adjacent cells. No two couches are intersecting, so each cell is occupied by at most one couch. The cells that are not occupied are empty.

Pasha has decided to unfold as many couches as possible to make it difficult to walk around the hall, and then capture customers reactions. When a couch is unfolded, it occupies a 2×2 square that contains two cells initially occupied by this couch, and another two cells that are uniquely determined by the direction of couch unfolding.

Help Pasha to determine the maximum number of couches that can be unfolded. Output the instruction which couches should be unfolded.

Input

The first line contains two integers n and m — the size of the grid ($1 \le n, m \le 1\,000$).

The next n lines describe hall. Each line contains m characters. Character "`.`" corresponds to empty cells. Cells occupied by couches are denoted by lowercase or uppercase English letters. Two cells occupied by the same couch are denoted by the same character. Cells of two adjacent couches are denoted by distinct characters. If a couch is denoted by a lowercase letter, it unfolds upwards or to the left. If a couch is denoted by an uppercase letter, it unfolds downwards or to the right.

Output

The first line should contain a single integer — the maximum number of couches that can be unfolded. Each of the following n lines should contain m characters — the description of the hall with unfolded couches. Character "`.`" denotes an empty cell. Cells occupied by couches should be denoted by digits. Cells occupied by the same couch should be denoted by the same digit. Two adjacent couches should be denoted by distinct digits.

Examples

standard input	standard output
4 4 .AA. A..a A..a .aa.	2 .11. 0022 0022 .11.
3 4 .XX. .... YYZZ	1 .00. .00. 1122
3 4 .XX. .... yyzz	2 .00. 1122 1122

Problem F. SMS from MCHS

Time limit: 1 second
Memory limit: 512 megabytes

You work for MCHS (Russian Ministry of Emergency Situations). You have just received a report from Hydro-meteorological Center containing an information about today's weather and the forecast for tomorrow.

According to this report, the air temperature is t_1 degrees today, and the wind speed is v_1 meters per second. Tomorrow the air temperature will be t_2 degrees, and the wind speed will be v_2 meters per second.

You are given a task to notify citizens about the weather for tomorrow via SMS.

The most important goal is to warn citizens in case the storm is possible. If, according to the forecast, the temperature tomorrow will be negative, and the wind speed will be at least 10 meters per second, you should send a message with following text:

```
A storm warning for tomorrow! Be careful and stay home if possible!
```

Otherwise, you may just notify citizens about bad weather changes.

If the temperature tomorrow will be lower than today, then you should send a message with a warning about a cold snap. It should have the following text:

```
MCHS warns! Low temperature is expected tomorrow.
```

Otherwise, if wind speed tomorrow will be higher than today, then you should send a message with a warning about strong wind. It should have the following text:

```
MCHS warns! Strong wind is expected tomorrow.
```

If none of the above conditions is satisfied, the you don't have to send a message at all.

Given the report from Hydro-meteorological Center, determine, what message has to be sent.

Input

The first line of input contains two integers t_1 and v_1 — the temperature and the wind speed for today ($-50 \le t_1 \le 50$; $0 \le v_1 \le 20$). The second line contains two integers t_2 and v_2 — the temperature and the wind speed for tomorrow ($-50 \le t_1 \le 50$; $0 \le v_1 \le 20$).

Output

In case if any message has to be sent, output its text. Otherwise, output the phrase "`No message`".

You can separate message words with spaces and line feeds arbitrarily.

Examples

standard input	standard output
15 2 5 3	MCHS warns! Low temperature is expected tomorrow.
15 1 17 1	No message

Problem G. Cooking

Time limit: 1 second
Memory limit: 512 megabytes

Stephen and Sergey are enrolled at the university and live on campus. So now they need to learn to cook.

The friends have learned to cook n different dishes. After buying all the necessary food, the guys have decided that before they go shopping next time, they will cook the i-th dish exactly a_i times.

Each day Sergey and Stephen choose two dishes i and j and cook them, cooking takes $c_{i,j}$ units of time. It's possible that $i = j$, that means that the i-th dish is cooked twice on that day.

The guys are quite lazy, so they want to minimize the total cooking time for all the days before the next shopping. Help them to do it!

Input

The first line contains a single integer n ($1 \le n \le 10$) — the number of dishes that the guys can cook.

The second line contains n positive integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 50$) — the number of times it must be cooked for each dish.

The next n lines contain n space separated integers $c_{i,j}$ ($1 \le c_{i,j} \le 100$), the j-th number in the i-th line denotes cooking time of pair of dishes i and j. It is guaranteed that $c_{i,j} = c_{j,i}$.

Output

Print one integer: the minimum total time of cooking, or -1, if it is impossible to make a cooking plan such that the i-th dish is cooked exactly a_i times.

Examples

standard input	standard output
3 2 2 2 1 4 3 4 4 5 3 5 6	10
2 2 39 23 9 9 23	-1
1 2 100	100

Note

In the first test case it is optimal to cook the following pairs of dishes: $(1,3)$, $(1,3)$, $(2,2)$.

Problem H. Hard Work

Time limit: 1 second
Memory limit: 512 megabytes

Alexey works as a mathematician in a well-known company "WordCount". Since his project has recently been closed, he was given a rather strange kind of assignment as a replacement: he must write down consecutive integers in a certain range every month, and at the end of the month the accounting department makes some calculations that determine Alexey's salary.

Alexey's salary is calculated as follows: first, the accounting department finds such x that there exists an integer with x identical consecutive digits among the integers that Alexey has written down, but there is no integer with $x + 1$ identical consecutive digits among them. Then the integers that have x identical consecutive digits are counted and the resulting amount is Alexey's salary.

Alexey is a smart mathematician so he doesn't want to work for peanuts. Today he was given a work plan for the next t months. During the i-th month he must write down integers from l_i to r_i inclusive. Help Alexey to calculate, what his salary would be each month, if the accounting department always does its calculations right.

Input

The first line of input contains a single integer t, the number of months for which Alexey's salary should be calculated ($1 \le t \le 10^4$).

The i-th of the following t lines contains two space-separated integers l_i and r_i, the first and the last integer Alexey will write down during the i-th month ($1 \le l_i \le r_i \le 10^{18}$).

Output

For each given month you should print a line with the calculated value of x and Alexey's salary for that month.

Examples

standard input	standard output
1 312 348	3 1
1 223 329	2 17

Note

Alexey writes down integers from 312 and 348 in the first example. There is an integer 333 that has three identical consecutive digits. There are no other integers between those that have three identical consecutive digits, so the answer is "`3 1`".

In the second example, there are no integers with three identical consecutive digits between 223 and 329. The integers with two identical consecutive digits can be divided into three groups:

- $\overline{22x}$ (from 223 to 229)
- $\overline{2xx}$ (from 233 to 299)
- $\overline{3xx}$ (300, 311 and 322)

There are exactly 17 integers so the answer is "`2 17`".

Problem I. Points and Segments

Time limit:	1 second
Memory limit:	512 megabytes

This is interactive problem.

Alice and Bob are playing a new game “Points and Segments” at the boring Geometry lesson.

The game board is a sheet of paper, it has n distinct points, no three of those are on the same line. The players make moves in turn, Alice moves first. The current player’s move is to choose two points and connect them by a segment. The new segment must not have common internal points with the previous segments (they can have a common endpoint). The player who cannot make a move loses.

In this problem your program will play “Points and Segments” with the judges program, and it must win. Your program will get a board with n points and must first choose whether it would like to play for Alice, or for Bob. After that your program must make moves for the chosen player until it wins.

Interaction Protocol

First your program must read a single integer n — the number of points on the board ($3 \le n \le 300$).

After that read n pairs of integers (x_i, y_i) — coordinates of the points ($-10^4 \le x_i, y_i \le 10^4$; all points are distinct, no three points are on the same line).

After analyzing the field, your program must decide whether it will play for Alice and move first, or play for Bob and move second. Print 1 if it wants to play for Alice, print 2 if it wants to play for Bob. Don’t forget to end the line after this output.

After that the players make moves in turn, according to the choice your program has made. If it’s your program’s move, it should print two integers on a line: i and j — the indices of points that it would like to connect by a segment ($1 \le i, j \le n$; i and j must be distinct, the segment must not have common internal points with existing segments). If it is judges program’s turn, it prints its move in the same format, your program must read it from its standard input.

If your program is in a situation that it cannot make a move, it will be forced to terminate by the judging system. The judging system will not wait for its reaction. If your program wins, and the judges program has no move, the judges program will print “`0 0`” instead of its move. Your program must read these two zeroes and terminate.

Examples

standard input	standard output
3 0 0 10 0 0 10 1 3 0 0	 1 1 2 2 3
4 0 0 10 0 5 7 5 3 1 2 1 3 2 3 0 0	 2 1 4 2 4 3 4

Note

In the example above the judges and participant's programs messages are formatted with empty lines to show which message is a response to which one. In real interaction there are no empty lines, do not print them. But you must end each printed message with a new line character.

Problem J. Straight

Time limit: 1 second
Memory limit: 512 megabytes

The tournament of generalized poker is going to start! The tournament uses a special deck. It consists of cards that contain integers from 1 to n. Every number has infinitely many occurrences in the deck.

There is only one hand in generalized poker: a straight. A straight is a sequence of m cards that contain consecutive integers: $i, i+1, \ldots, i+m-1$. The rules of poker are the following: each player has s hole cards, and there are m community cards. Each player therefore sees $s+m$ cards and tries to choose m of them so that they formed a straight.

You are watching the tournament, so you can only see community cards. You wonder how many distinct straights are possible with such community cards for some hole cards. Two straights are distinct if they start with different i value.

Input

The first line contains three integers n, m and s ($1 \le n \le 10^9$; $1 \le s < m \le 10^5$) — the maximum value of the card, the number of community cards and the number of hole cards.

The second line contains m integers, each from 1 to n — community cards values.

Output

Output the number of distinct straights that can be potentially obtained by the players.

Examples

standard input	standard output
10 5 2 7 1 3 5 6	5
11 6 2 5 5 5 5 5 5	0

Note

The first example has the following possible straights: 1, 2, 3, 4, and 5.

Problem K. New Level

Time limit: 2 seconds
Memory limit: 512 megabytes

Robocity has n crossroads connected by bidirectional roads. There are m roads in total, and all crossroads are reachable from each other. There is a level assigned to each crossroad specified by a number from 1 to k, inclusive. Any pair of crossroads directly connected by a road has distinct levels.

The city leaders are planning a reform. Namely, they want to assign new levels to crossroads, so that each level still has a value from 1 to k, connected crossroads would have different levels, and an additional condition has to be met: for each pair of crossroads u and v there must exist a path between them, such that any two adjacent crossroads along it have levels that differ by 1 modulo k.

Formally, for each pair of crossroads (u, v) there should exist a sequence of crossroads $p_1, \ldots, p_l$, such that:

- $p_1 = u$;
- $p_l = v$;
- for each i from 1 to $l - 1$, crossroads p_i and p_{i+1} are connected, and either their levels differ by one, or one of them has level of 1 and another has level of k.

Robocity government is convinced that such level assignment exists and asks you to find it.

Input

The first line contains three integers n, m, k ($1 \le n, m, k \le 500\,000$), number of crossroads, roads, and levels.

The second line contains n integers $c_1, c_2, \ldots, c_n$ ($1 \le c_i \le k$), c_i is the level of the crossroad i.

Then m lines follow, each of them contains two integers u, v ($1 \le u, v \le n$; $u \ne v$), a pairs of crossroads connected by a road.

It is guaranteed that there are no two roads connecting the same pair of crossroads, and that there exists a path between each pair of crossroads.

Output

Output n integers $d_1, d_2, \ldots, d_n$ ($1 \le d_i \le k$), the levels of the crossroads in the new assignment.

Example

standard input	standard output
4 4 4 1 2 3 1 1 2 1 3 2 3 3 4	4 3 2 1

Problem L. The Firm Knapsack Problem

Time limit: 1 second
Memory limit: 512 megabytes

The Knapsack problem is a classic problem in Computer Science.

It is stated the following way. There are n items, for each item you know its weight w_i and cost $cost_i$. Also you know the capacity of the knapsack W — the upper limit for the total weight of taken items. The task is to select several items with total weight at most W so that their total cost is as large as possible.

In this problem you don't have to solve the classic Knapsack problem. The jury has already solved it and found the exact answer: x is the maximal possible total cost of items that fit into a knapsack of capacity W. The jury doesn't tell you this value.

Your task is to solve The Firm Knapsack Problem. Now the knapsack of claimed capacity W can hold the weight of items up to $\frac{3}{2}W$. You need to solve the problem with this weakened constraint not worse than the jury have solved the problem with the original constraint W.

In other words, you need to find a set of items with total cost at least x and total weight at most $\frac{3}{2}W$.

You are to solve the problem for t test cases.

Input

The first line contains the number of test cases. Then follow the tests in the following format.

The first line of the test case contains two integers n and W $(1 \le n \le 10^5; 1 \le W \le 10^{12})$ — the number of items and the claimed capacity of the knapsack.

The next n lines describe items. Each line contains two integers w_i and $cost_i$ $(1 \le w_i, cost_i \le 10^6)$ — the weight and the cost of an item.

The sum of n over all test cases is at most 10^5.

Output

For each test case, output the selected set of items for weight constraint $\frac{3}{2}W$ in the following format.

The first line should contain the number of taken items.

The second line should contain the indices of taken items $i_1\ i_2\ \ldots\ i_k$ $(1 \le i_j \le n)$. All indices i_j should be distinct. Items are numbered from 1 to n in the same order as they are given in the input.

If there are several solutions, output any one of them.

Example

standard input	standard output
3 3 10 5 100 5 100 4 99 3 100 97 100 98 101 99 90 3 100 55 100 99 150 200 200	3 3 1 2 1 2 1 2