Codeforces Round #777 (Div. 2)

A. Madoka and Math Dad

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Madoka finally found the administrator password for her computer. Her father is a well-known popularizer of mathematics, so the password is the answer to the following problem.

Find the maximum decimal number without zeroes and with no equal digits in a row, such that the sum of its digits is n.

Madoka is too tired of math to solve it herself, so help her to solve this problem!

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \le t \le 1000$) — the number of test cases. Description of the test cases follows.

The only line of each test case contains an integer n ($1 \le n \le 1000$) — the required sum of the digits.

Output

For each test case print the maximum number you can obtain.

Example

input

```
5
1
2
3
4
5
```

output

```
1
2
21
121
212
```

Note

The only numbers with the sum of digits equal to 2 without zeros are 2 and 11. But the last one has two ones in a row, so it's not valid. That's why the answer is 2.

The only numbers with the sum of digits equal to 3 without zeros are 111, 12, 21, and 3. The first one has 2 ones in a row, so it's not valid. So the maximum valid number is 21.

The only numbers with the sum of digits equals to 4 without zeros are 1111, 211, 121, 112, 13, 31, 22, and 4. Numbers 1111, 211, 112, 22 aren't valid, because they have some identical digits in a row. So the maximum valid number is 121.

B. Madoka and the Elegant Gift

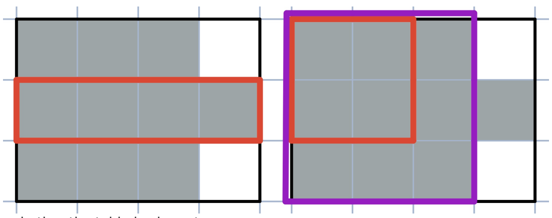
time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Madoka's father just reached 1 million subscribers on Mathub! So the website decided to send him a personalized award — The Mathhub's Bit Button!

The Bit Button is a rectangular table with n rows and m columns with 0 or 1 in each cell. After exploring the table Madoka found out that:

- A subrectangle A is contained in a subrectangle B if there's no cell contained in A but not contained in B.
- Two subrectangles intersect if there is a cell contained in both of them.
- A subrectangle is called **black** if there's no cell with value 0 inside it.
- A subrectangle is called **nice** if it's **black** and it's not contained in another **black** subrectangle.
- The table is called **elegant** if there are no two **nice** intersecting subrectangles.

For example, in the first illustration the red subrectangle is nice, but in the second one it's not, because it's contained in the purple subrectangle.



Help Madoka to determine whether the table is elegant.

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \le t \le 200$) — the number of test cases. Description of the test cases follows.

The first line of each test case contains two positive integers n, m ($1 \le n, m \le 100$).

The next n lines contain strings of length m consisting of zeros and ones — the description of the table.

It is guaranteed that the sum of the values of n and the sum of the values of m for all test cases do not exceed 777.

Output

For each test case print "YES" if its table is elegant or print "NO" otherwise.

You may print each letter in any case (for example, "YES", "Yes", "yes", "yEs" will all be recognized as positive answer).

Example

input

```
5
3 3
100
011
011
3 3
110
111
110
1 5
01111
4 5
11111
01010
01000
01000
3 2
11
00
```

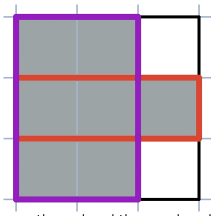
```
11
```

output

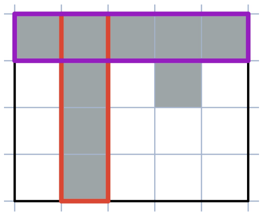
```
YES
NO
YES
NO
YES
```

Note

In the second test case the table is not elegant, because the red and the purple subrectangles are nice and intersect.



In the fourth test case the table is not elegant, because the red and the purple subrectangles are nice and intersect.



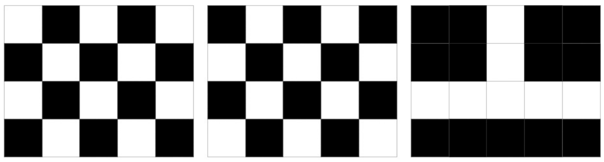
C. Madoka and Childish Pranks

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Madoka as a child was an extremely capricious girl, and one of her favorite pranks was drawing on her wall. According to Madoka's memories, the wall was a table of n rows and m columns, consisting only of zeroes and ones. The coordinate of the cell in the i-th row and the j-th column ($1 \le i \le n$, $1 \le j \le m$) is (i, j).

One day she saw a picture "Mahou Shoujo Madoka Magica" and decided to draw it on her wall. Initially, the Madoka's table is a table of size $n \times m$ filled with zeroes. Then she applies the following operation any number of times:

Madoka selects any rectangular subtable of the table and paints it in a chess coloring (the upper left corner of the subtable always has the color 0). Note that some cells may be colored several times. In this case, the final color of the cell is equal to the color obtained during the last repainting.



White color means 0, black means 1. So, for example, the table in the first picture is painted in a chess coloring, and the others are not.

For better understanding of the statement, we recommend you to read the explanation of the first test.

Help Madoka and find some sequence of no more than $n \cdot m$ operations that allows you to obtain the picture she wants, or determine that this is impossible.

Input

Each test contains multiple test cases. The first line contains a single integer t ($1 \le t \le 10$) — the number of test cases. Description of the test cases follows.

The first line of each test case contains two integers n and m ($1 \le n, m \le 100$) — the size of the table. Each of the following n lines contains a string of length m consisting only of 1 and 0 — description of the picture that Madoka wants to obtain.

Output

If it is impossible to obtain the given picture, print -1.

Otherwise, print in the first line a single integer q ($0 \le q \le n \cdot m$) — the number of operations you need to obtain the picture. Note that you do **not** need to minimize the number of operations.

Then for each operation (in the order of execution) print a single line containing four numbers — the coordinates of the upper-left corner and the lower-right corner of the rectangle.

Example

input

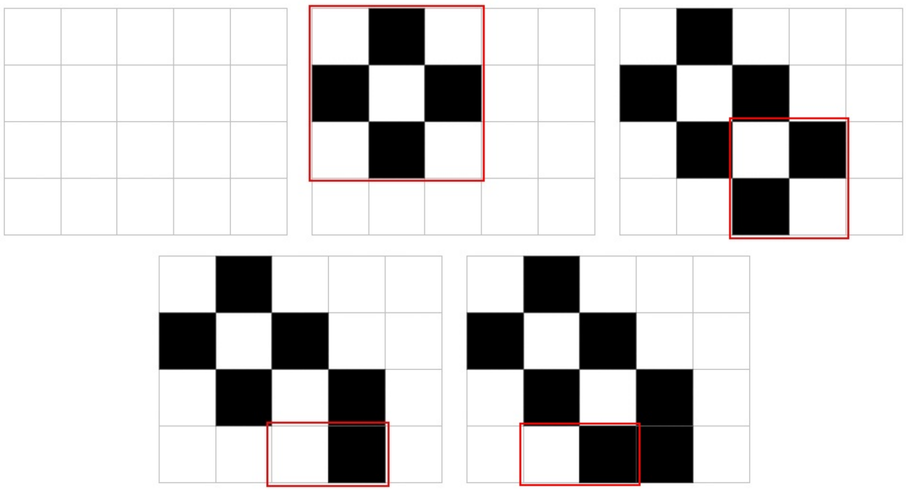
```
4
4 5
01000
10100
01010
00110
2 3
001
010
3 3
110
101
000
1 1
0
```

output

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

Note

The description of the first test case is below.



In the third test case, it is impossible to paint the desired picture.

In the fourth test case, the initial table is already the desired picture.

D. Madoka and the Best School in Russia

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Madoka is going to enroll in "TSUNS PTU". But she stumbled upon a difficult task during the entrance computer science exam:

- A number is called *good* if it is a multiple of d.
- A number is called *beatiful* if it is **good** and it **cannot** be represented as a product of two good numbers.

Notice that a beautiful number must be good.

Given a good number x, determine whether it can be represented in at least two different ways as a product of several (possibly, one) **beautiful** numbers. Two ways are different if the sets of numbers used are different.

Solve this problem for Madoka and help her to enroll in the best school in Russia!

Input

The first line contains a single integer t ($1 \leq t \leq 100$) — number of test cases. Below comes their description.

Each test case consists of two integers x and d, separated by a space ($2 \leq x, d \leq 10^9$). It is guaranteed that x is a multiple of d.

Output

For each set of input data, output "NO" if the number **cannot be** represented in at least two ways. Otherwise, output "YES".

You can output each letter in any case (for example, "YES", "Yes", "yes", "yEs", "yEs" will be recognized as a positive answer).

Example

input

```
8
6 2
12 2
36 2
8 2
1000 10
2376 6
128 4
16384 4
```

output

```
NO
NO
YES
NO
YES
YES
NO
YES
```

Note

In the first example, 6 can be represented as 6, $1 \cdot 6$, $2 \cdot 3$. But 3 and 1 are not a good numbers because they are not divisible by 2, so there is only one way.

In the second example, 12 can be represented as $6 \cdot 2$, 12, $3 \cdot 4$, or $3 \cdot 2 \cdot 2$. The first option is suitable. The second is— no, because 12 is not beautiful number ($12 = 6 \cdot 2$). The third and fourth are also not suitable, because 3 is not good number.

In the third example, 36 can be represented as $18 \cdot 2$ and $6 \cdot 6$. Therefore it can be decomposed in at least two ways.

E. Madoka and the Sixth-graders

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

After the most stunning success with the fifth-graders, Madoka has been trusted with teaching the sixth-graders.

There's n single-place desks in her classroom. At the very beginning Madoka decided that the student number b_i ($1 \leq b_i \leq n$) will sit at the desk number i. Also there's an infinite line of students with numbers $n+1, n+2, n+3, \ldots$ waiting at the door with the hope of being able to learn something from the Madoka herself. Pay attention that each student has his **unique** number.

After each lesson, the following happens in sequence.

1. The student sitting at the desk i moves to the desk p_i. All students move simultaneously.
2. If there is more than one student at a desk, the student with the lowest number keeps the place, and the others are removed from the class **forever**.
3. For all empty desks in ascending order, the student from the lowest number from the outside line occupies the desk.

Note that in the end there is exactly one student at each desk again. It is guaranteed that the numbers p are such that at least one student is removed after each lesson. Check out the explanation to the first example for a better understanding.

After several (possibly, zero) lessons the desk i is occupied by student a_i. Given the values $a_1, a_2, \ldots, a_n$ and $p_1, p_2, \ldots, p_n$, find the lexicographically smallest suitable initial seating permutation $b_1, b_2, \ldots, b_n$.

The permutation is an array of n different integers from 1 up to n in any order. For example, $[2, 3, 1, 5, 4]$ is a permutation, but $[1, 2, 2]$ is not (2 occurs twice). $[1, 3, 4]$ is not a permutation either ($n = 3$ but there's 4 in the array).

For two different permutations a and b of the same length, a is lexicographically less than b if in the first position where a and b differ, the permutation a has a smaller element than the corresponding element in b.

Input

The first line of input data contains an integer n ($2 \le n \le 10^5$) — a number of desks in the classroom.

The second line contains n integers $p_1, p_2, \ldots, p_n$ ($1 \le p_i \le n$) — desks where the students move. It is guaranteed that p has at least two equal elements.

The third line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$) — the final seating of the students. It is guaranteed that there is an initial permutation from which the seating a can be obtained.

Output

In the only line print n integers $b_1, b_2, \ldots, b_n$ ($1 \le b_i \le n$) — lexicographically minimum permutation describing the initial seating of the sixth-graders that can lead to the final seating a.

Examples

input

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

output

```
1 3 2 5 4
```

input

```
5
1 3 2 5 2
3 2 5 4 1
```

output

```
3 2 5 4 1
```

input

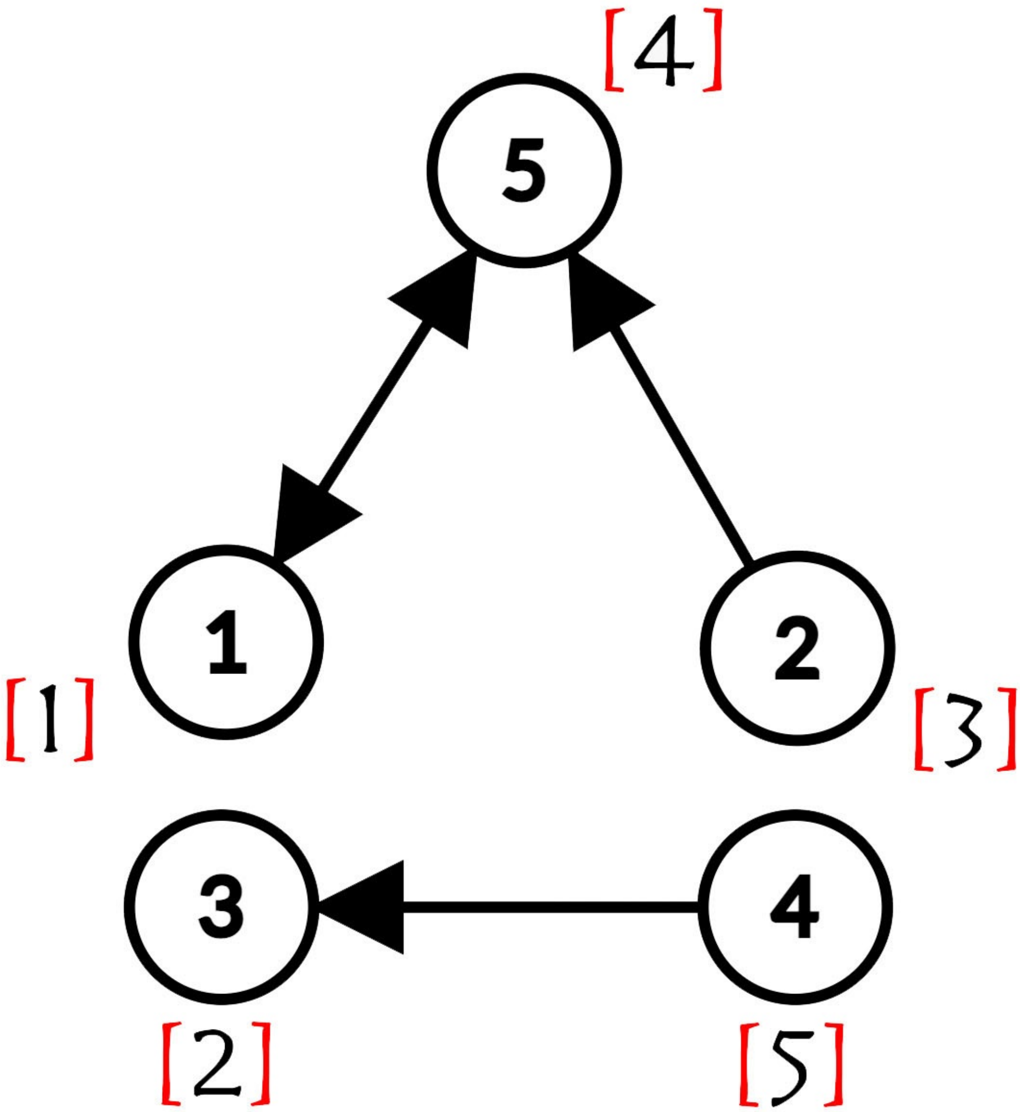
```
10
10 8 5 3 7 8 6 6 1 1
5 26 24 27 21 4 18 2 28 1
```

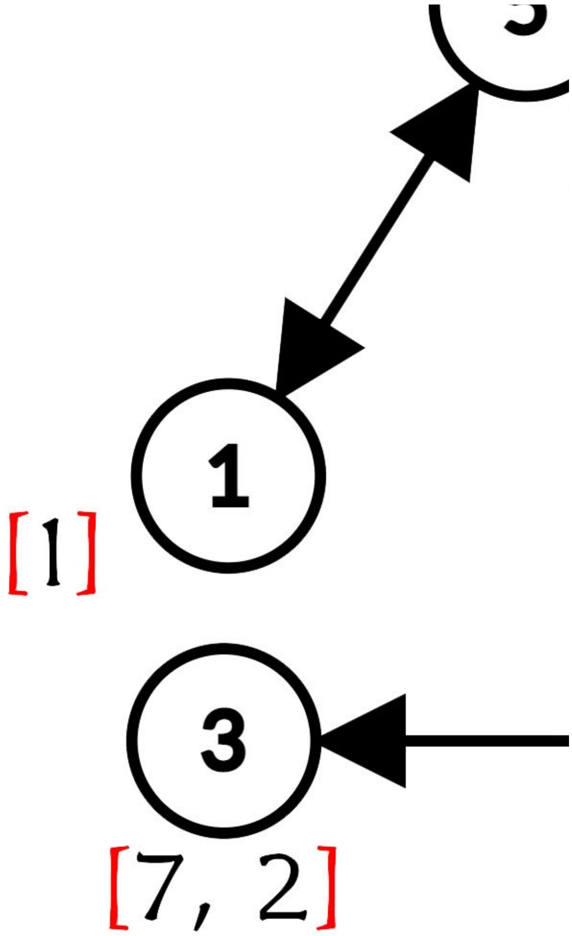
output

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

Note

The description of the first test is below:





The first picture shows the starting permutation, which is the answer. Then the students sitting at desks $1, 2$ are transferred to a 5 desk. Also, a 1 student moved from a 5 desk, and a student from a 4 disk is transferred to a 3 desk.

Thus, after all these transfers permutation shown in the second image is obtained. Then, at the desk with the number 5, the student with the number 3 is expelled, and at the desk with the number 3, the student with the number 5 is expelled. (Since their numbers are not the smallest) Then new students with numbers $6, 7$ sit at desks numbered $2, 4$. And this permutation (after the end of the first lesson) is shown in the third image.

The 4 image shows the seating arrangement, after the second lesson before all the extra ones were kicked out. And the fifth shows the final seating after 2 lesson.

F. Madoka and Laziness

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Madoka has become too lazy to write a legend, so let's go straight to the formal description of the problem.

An array of integers $a_1, a_2, \ldots, a_n$ is called a hill if it is not empty and there is an index i in it, for which the following is true: $a_1 < a_2 < \ldots < a_i > a_{i+1} > a_{i+2} > \ldots > a_n$.

A sequence x is a subsequence of a sequence y if x can be obtained from y by deletion of several (possibly, zero or all) elements keeping the order of the other elements. For example, for an array $[69, 1000, 228, -7]$ the array $[1000, -7]$ is a subsequence, while $[1]$ and $[-7, 1000]$ are not.

Splitting an array into two subsequences is called good if each element belongs to exactly one subsequence, and also each of these subsequences is a hill.

You are given an array of **distinct** positive integers $a_1, a_2, \ldots a_n$. It is required to find the number of different pairs of maxima of the first and second subsequences among all good splits. Two pairs that only differ in the order of elements are considered same.

Input

The first line of input contains a single integer n ($2 \le n \le 5 \cdot 10^5$) — array size.

The second line of input contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$) — the elements of the array. It is guaranteed that all a_i are pairwise distinct.

Output

In a single line, print exactly one number — the number of different pairs of maxima of the first and second subsequences among all good splits.

Examples

input
```
4
1 2 4 3
```
output
```
3
```

input
```
8
2 12 13 7 14 6 11 8
```
output
```
4
```

input
```
7
9 5 3 10 2 6 8
```
output
```
0
```

input
8 8 6 10 9 1 5 2 14

output
0

Note

In the first test case there are 3 possible pairs: $(3, 4)$, $(2, 4)$, $(1, 4)$. And they are achieved with the following partitions: $[1, 2, 3]$, $[4]$; $[4, 3]$, $[1, 2]$; $[1]$, $[2, 4, 3]$