A. Kamilka and the Sheep

Input file:	standard input
Output file:	standard output
Time limit:	1 second
Memory limit:	256 megabytes

Kamilka has a flock of n sheep, the i-th of which has a beauty level of a_i. All a_i are distinct. Morning has come, which means they need to be fed. Kamilka can choose a non-negative integer d and give each sheep d bunches of grass. After that, the beauty level of each sheep increases by d.

In the evening, Kamilka must choose **exactly two** sheep and take them to the mountains. If the beauty levels of these two sheep are x and y (after they have been fed), then Kamilka's *pleasure* from the walk is equal to $\gcd(x, y)$, where $\gcd(x, y)$ denotes the greatest common divisor (GCD) of integers x and y.

The task is to find the maximum possible pleasure that Kamilka can get from the walk.

Input

Each test consists of several test cases. The first line contains one integer t ($1 \le t \le 500$), the number of test cases. The description of the test cases follows.

The first line of each test case contains one integer n ($2 \le n \le 100$), the number of sheep Kamilka has.

The second line of each test case contains n **distinct** integers $a_1, a_2, \ldots, a_n$ $(1 \le a_i \le 10^9)$ — the beauty levels of the sheep.

It is guaranteed that all a_i are distinct.

Output

For each test case, output a single integer: the maximum possible pleasure that Kamilka can get from the walk.

Standard Input	Standard Output
4 2 1 3 5 5 4 3 2 1 3 5 6 7 3 1 11 10	2 4 2 10

Note

In the first test case, $d = 1$ works. In this case, the pleasure is $\gcd(1 + 1,\ 1 + 3) = \gcd(2,\ 4) = 2$. It can be shown that a greater answer cannot be obtained.

In the second test case, let's take $d = 3$. In this case, the pleasure is $\gcd(1 + 3,\ 5 + 3) = \gcd(4,\ 8) = 4$. Thus, for this test case, the answer is 4.

,

B. Lady Bug

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

As soon as Dasha Purova crossed the border of France, the villain Markaron kidnapped her and placed her in a prison under his large castle. Fortunately, the wonderful Lady Bug, upon hearing the news about Dasha, immediately ran to save her in Markaron's castle. However, to get there, she needs to crack a complex password.

The password consists of two bit strings a and b, each of which has a length of n. In one operation, Lady Bug can choose any index $2 \le i \le n$ and perform one of the following actions:

1. swap(a_i, b_{i-1}) (swap the values of a_i and b_{i-1}), or
2. swap(b_i, a_{i-1}) (swap the values of b_i and a_{i-1}).

Lady Bug can perform any number of operations. The password is considered cracked if she can ensure that the first string consists only of zeros. Help her understand whether or not she will be able to save the unfortunate Dasha.

Input

Each test consists of several test cases. The first line of the input data contains one integer t ($1 \le t \le 10^4$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains one integer n ($2 \le n \le 2 \cdot 10^5$) — the length of the bit strings of the password.

The next two lines contain the bit strings of length n, a and b, which represent the password. Each of the strings contains only the characters `0` and `'1'`.

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

Output

For each test case, output "YES" if Lady Bug can crack the password after any number of operations; otherwise, output "`NO`".

You can output each letter in any case (lowercase or uppercase). For example, the strings "`yEs`", "`yes`", "`Yes`", and "YES" will be accepted as a positive answer.

Standard Input	Standard Output
4 3 000 000 6 010001 010111 5 10000 01010 2 11 00	YES YES NO YES

Note

In the first test case, the string a immediately consists only of zeros.

In the second test case, a possible sequence of operations is:

1. swap$(a_2,\ b_1)$

   010001

   010111

2. swap$(b_5,\ a_4)$

000001

110111

3. swap$(a_4,\ b_3)$

000101

110101

4. swap$(a_5,\ b_4)$

000001

111101

C. Asuna and the Mosquitoes

Input file:	standard input
Output file:	standard output
Time limit:	2 seconds
Memory limit:	256 megabytes

For her birthday, each of Asuna's n admirers gifted her a tower. The height of the tower from the i-th admirer is equal to a_i.

Asuna evaluates the beauty of the received gifts as $\max(a_1, a_2, \ldots, a_n)$. She can perform the following operation an arbitrary number of times (possibly, zero).

- Take such $1 \le i \ne j \le n$ that $a_i + a_j$ is odd and $a_i > 0$, then decrease a_i by 1 and increase a_j by 1.

It is easy to see that the heights of the towers remain non-negative during the operations.
Help Asuna find the maximum possible beauty of the gifts after any number of operations!

Input

Each test consists of several test cases. The first line of the input data contains one integer t ($1 \le t \le 10^4$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains a single integer n ($1 \le n \le 2 \cdot 10^5$) — the number of admirers of Asuna.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ $(1 \le a_i \le 10^9)$ — the heights of the towers.

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

Output

For each test case, output a single integer: the maximum value of the beauty of the gifts that Asuna can achieve.

Standard Input	Standard Output
4 3 5 3 9 2 3 2 4 1 2 2 1 5 5 4 3 2 9	9 5 5 21

Note

In the first test case, no pair of towers satisfies the required condition for applying the operation, so no operations can be performed. In this case, the answer is $\max(5,\ 3,\ 9) = 9$.

In the second test case, the operation with $i = 2$ and $j = 1$ can be applied twice. After that, the array becomes: $a = [5,\ 0]$. Thus, the answer is 5.

In the third test case, the following sequence of operations can be applied:

1. Operation with $i = 1$ and $j = 2$.
   $[1,\ 2,\ 2,\ 1] \quad \to \quad [0,\ 3,\ 2,\ 1]$
2. Operation with $i = 3$ and $j = 2$.
   $[0,\ 3,\ 2,\ 1] \quad \to \quad [0,\ 4,\ 1,\ 1]$
3. Operation with $i = 3$ and $j = 2$.
   $[0,\ 4,\ 1,\ 1] \quad \to \quad [0,\ 5,\ 0,\ 1]$

$\max(0,\ 5,\ 0,\ 1) = 5.$

Therefore, the answer is 5.

,

D. Mishkin Energizer

```
Input file:     standard input
Output file:    standard output
Time limit:     2 seconds
Memory limit:   256 megabytes
```

In anticipation of a duel with his old friend Fernan, Edmond is preparing an energy drink called "`Mishkin Energizer`". The drink consists of a string s of length n, made up only of the characters `L`, `I`, and `T`, which correspond to the content of three different substances in the drink.

We call the drink *balanced* if it contains an equal number of all substances. To boost his aura and ensure victory in the duel, Edmond must make the initial string balanced by applying the following operation:

1. Choose an index i such that $s_i \neq s_{i+1}$ (where $i + 1$ must not exceed the **current** size of the string).
2. Insert a character x, either `L`, `I`, or `T`, between them such that $x \neq s_i$ and $x \neq s_{i+1}$.

Help Edmond make the drink balanced and win the duel by performing **no more than $\mathbf{2n}$ operations**. If there are multiple solutions, any one of them can be output. If it is impossible, you must report this.

Input

Each test consists of several test cases. The first line of the input data contains one integer t ($1 \le t \le 100$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains one integer n ($1 \le n \le 100$) — the length of the string s.

The second line of each test case contains a string s of length n, consisting only of the characters `L`, `I`, and `T`.

Output

For each test case, output -1 if there is no solution. Otherwise, in the first line, output a single integer m ($0 \le m \le 2n$) — the number of operations you performed.

Then the l-th of the following m lines should contain a single integer i ($1 \le i < n + l - 1$), indicating the operation of inserting a character between s_i and s_{i+1}. It must hold that $s_i \neq s_{i+1}$.

If there are multiple solutions, any one of them can be output. Note that you do not need to minimize the number of operations in this problem.

Standard Input	Standard Output
3 5 TILII 1 L 3 LIT	4 1 2 3 4 -1 0

Note

In the first test case, the following sequence of operations can be performed: `TILII` $\to$ `T`**`L`**`ILII` $\to$ `TL`**`T`**`ILII` $\to$ `TLT`**`L`**`ILII` $\to$ `TLTL`**`T`**`ILII`.

In the second test case, no operations can be performed, so the answer is -1.

In the third test case, the initial string already has equal quantities of all substances.

,

E. She knows...

```
Input file:     standard input
Output file:    standard output
Time limit:     2 seconds
Memory limit:   256 megabytes
```

D. Pippy is preparing for a "black-and-white" party at his home. He only needs to repaint the floor in his basement, which can be represented as a board of size $n \times m$.

After the last party, the entire board is painted green, except for some k cells $(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)$, each of which is painted either white or black. For the upcoming party, D. Pippy wants to paint **each** of the remaining green cells either black or white. At the same time, he wants the number of pairs of adjacent cells with different colors on the board to be even after repainting.

Formally, if

$$A = \{((i_1, j_1), (i_2, j_2)) \mid 1 \le i_1, i_2 \le n, 1 \le j_1, j_2 \le m, i_1 + j_1 < i_2 + j_2, |i_1 - i_2| + |j_1 - j_2| = 1, \text{color}(i_1, j_1) \ne \text{color}(i_2, j_2)\},$$

where $\text{color}(x, y)$ denotes the color of the cell (x, y), then it is required that $|A|$ be even.

Help D. Pippy find the number of ways to repaint the floor so that the condition is satisfied. Since this number can be large, output the remainder of its division by $10^9 + 7$.

Input

Each test consists of several test cases. The first line of the input data contains one integer t ($1 \le t \le 10^4$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains three integers n, m, k ($3 \le n, m \le 10^9$; $1 \le k \le 2 \cdot 10^5$) — the dimensions of the board and the number of cells that are initially not green.

In the i-th of the following k lines of each test case, there are three integers x_i, y_i and c_i ($1 \le x_i \le n$; $1 \le y_i \le m$; $c_i \in \{0, 1\}$) — the coordinates of the cell and its color (if white, then $c_i = 0$; if black, then $c_i = 1$). It is guaranteed that all cells are distinct.

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

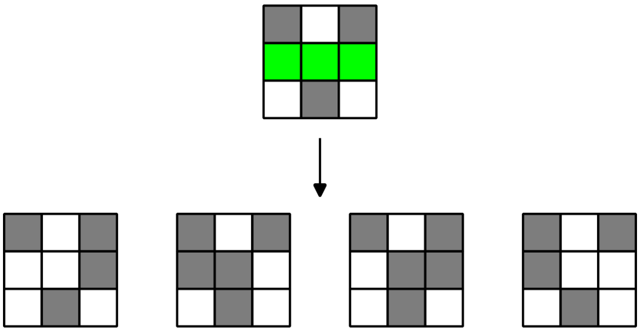
Output

For each test case, output a single integer — the answer modulo $10^9 + 7$.

Standard Input	Standard Output
2 3 3 6 1 1 0 1 2 1 1 3 0 3 1 1 3 2 0 3 3 1 3 4 12 1 1 0 1 2 1 1 3 0 1 4 1 2 1 1 2 2 0 2 3 1 2 4 0 3 1 0 3 2 1 3 3 0 3 4 1	4 0

Note

In the first test case, there are exactly 4 ways to paint the green cells $(2, 1), (2, 2), (2, 3)$, namely: $(1, 1, 0), (0, 0, 1), (1, 0, 0), (0, 1, 1)$ (the colors are indicated in the same order as the cells), as shown below.



Example 1.

In the second test case, all cells of the board are already painted, and the number of pairs of adjacent cells with different colors on the board is odd, so the answer is zero.

,

F. Andryusha and CCB

```
Input file:     standard input
Output file:    standard output
Time limit:     2 seconds
Memory limit:   256 megabytes
```

Let us define the *beauty* of a binary string z as the number of indices i such that $1 \le i < |z|$ and $z_i \ne z_{i+1}$.

While waiting for his friends from the CCB, Andryusha baked a pie, represented by a binary string s of length n. To avoid offending anyone, he wants to divide this string into k substrings such that each digit belongs to exactly one substring, and the beauties of all substrings are the same.

Andryusha does not know the exact number of friends from the CCB who will come to his house, so he wants to find the number of values of k for which it is possible to split the pie into exactly k parts with equal beauties.

However, Andryusha's brother, Tristan, decided that this formulation of the problem is too simple. Therefore, he wants you to find the number of such values of k **for each prefix of the string**. In other words, for each i from 1 to n, you need to find the number of values of k for which it is possible to split the prefix $s_1 s_2 \dots s_i$ into exactly k parts with equal beauties.

Input

Each test consists of several test cases. The first line of the input data contains one integer t ($1 \le t \le 10^5$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains a single integer n ($1 \le n \le 10^6$) — the length of the binary string.

The second line of each test case contains a binary string of length n, consisting only of digits 0 and 1.

It is guaranteed that the sum of n across all test cases does not exceed 10^6.

Output

For each test case, output a single line containing n integers c_i ($0 \le c_i \le n$) — the number of values of k for which it is possible to split the prefix $s_1 s_2 \dots s_i$ into exactly k parts with equal beauties.

Standard Input	Standard Output
3 5 00011 10 0101010101 7 0010100	1 2 3 4 5 1 2 2 3 2 4 2 4 3 4 1 2 3 3 4 3 4

Note

In the third case, the values of k that satisfy the conditions are:

1. $i = 1$: $k \in \{1\}$,
2. $i = 2$: $k \in \{1, 2\}$,
3. $i = 3$: $k \in \{1, 2, 3\}$,
4. $i = 4$: $k \in \{1, 3, 4\}$,
5. $i = 5$: $k \in \{1, 2, 4, 5\}$,
6. $i = 6$: $k \in \{1, 5, 6\}$,
7. $i = 7$: $k \in \{1, 5, 6, 7\}$.