Problem A. Almost String Matching

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

Fcrh is the god of string. He can solve string matching with many different algorithms: KMP, Gusfield, Boyer, Suffix Array, Suffix Automaton, Hash, etc. Just string matching is way too easy and way too boring for him. So he try to challenge you with "Almost String Matching".

We say two characters are almost same if they are same or adjacent english letters. For example, `a` and `b` are almost same, `t` and `s` are almost same, `g` and `g` are almost same, but `a` and `z` are not. We say two strings a, b are almost same if $|a| = |b|$ and all a_i, b_i are almost same for all $1 \leq i \leq |a|$. For example, `any` and `boy` are almost same, `stride` and `rushed` are almost same, but `shik` and `fcrh` are not (The later is much stronger!).

To solve "Almost String Matching", you need to find all occurrences of a pattern s within a text t as an almost same substring.

Input

The input contains 2 lines. The first line is the pattern s, and the second line is the text t.

- $1 \leq |s| \leq |t| \leq 250000$
- The strings s and t are consisting of lowercase English letters.

Output

Please output one or two lines. The first line is the number of possible match positions. If there is any match position, output one more line with match positions in ascending order and separated by a space.

Example

standard input	standard output
any amyisaboy	2 1 7
fcrh ismuchstrongerthanshik	0

Problem B. BWT

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

Hrcf used to be a ICPC competitor in NTU (NTU Tmt514 University), and he hates BWT (Burrows–Wheeler transform).

It all began on a windy night in 2015, when he was still a ICPC competitor. Hrcf have been dreaming of going to ICPC World Final since he's in senior high. And that contest, NTU PK 2015, is the last chance to accomplish his dream as he was about to graduate from NTU.

He spent all of his time learning algorithms, participating contests on codeforces, solving problems on OJs, optimizing workflow. Despite all the hardwork, he failed. That's all because Hrcf was not familiar with BWT and got his AC a bit slower than a rival, Fcrh.

From then on, he has been trying to alter all words that seems like a BWT result as a revenge. Unfortunately, there're too many words on the world to be altered, even "`ICPC`" could be a BWT result of "`PICC`".

Being Hrcf's best friend, you decide to write a program to help Hrcf. So that he only needs to replace minimal amount of characters to make a word not a result of BWT of any string.

As a friend of Hrcf, you're not familiar with BWT, either. Luckily, there's a note from Fcrh.

A BWT of a string is to sort all rotations of such string in lexical order and collect the last character of each rotation in order. For example, after sorting "`PICC`"'s all rotations, you'll get "`CCPI`", "`CPIC`", "`ICCP`", "`PICC`". Hence, the BWT result will be "`ICPC`".

Input

There're multiple test cases in one file, one for each line. In each test case, you'll be given a string S consist of lower case alphabets.

- $1 \leq |S| \leq 10^6$
- $\sum |S| \leq 10^7$

Output

Output a **ibwtified** string based on S or "`life is hard`" if that's impossible. You should modify as less character as possible and select the result with minimal lexical order under such constraint.

Examples

standard input	standard output
icpc ntu	acpc ntu

Problem C. Calculus is Quite Easy

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

The students in NTU would take lots of course in the first year. Calculus is a disaster for lots of students.

Since you're simply good at ACM (Advanced Computer Mathematics), we'll give you a challenge.

Given a polynomial p with $2n$ variables $x_1, \ldots, x_n$ and $y_1, \ldots, y_n$ where $p = p_1 + p_2 + \cdots + p_m$ and p_i is in form of "x_a" , "y_b" or "$x_a y_b$".

Determine whether the following equation is satisfied.

$$\frac{\partial}{\partial x_1}\frac{\partial}{\partial y_1}\frac{\partial}{\partial x_2}\frac{\partial}{\partial y_2}\cdots\frac{\partial}{\partial x_n}\frac{\partial}{\partial y_n}\ p^k\Big|_{x_1,x_2,y_1,y_2,\ldots,x_n,y_n=0} = 0$$

For example, if $n = 1, k = 2$ and $p = x_1 + y_1$, we have

$$\begin{aligned}\frac{\partial}{\partial x_1}\frac{\partial}{\partial y_1}\ (x_1+y_1)^2\Big|_{x_1=0,y_1=0} &= \frac{\partial}{\partial y_1}\ 2x_1+2y_1\Big|_{x_1=0,y_1=0}\\ &= 2|_{x_1=0,y_1=0}\\ &= 2\end{aligned}$$

Input

The first line contains two integers n, k.

The second line contains a string S indicating the polynomial p.

- $1 \le n \le 300$
- $1 \le k \le 10^5$
- $1 \le |S| \le 20000$

Output

Output "`YES`" if the equation is satisfied. Otherwise output "`NO`".

Examples

standard input	standard output
1 1 x1+y1	YES
1 2 x1+y1	NO
2 2 x1+x1*y1+x2*y2	NO
1 1 x1+y1+x1	YES

Problem D. Guess the Answer

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

In the previous problem, the answer of each test case is "Yes" or "No".

Shik hacked into the judge system with some incredible SQL injection tricks. Although Shik couldn't download all the test cases due to permission denied, he got something useful. There are only n test cases in the problem! And he also got an oracle which could tell him the number of test cases he guessed correctly. To interact with the oracle, we repesent "Yes" as 1 and "No" as 0.

For example, if there are 5 test cases and the answers are 00101, the oracle will return 2 if Shik asked 01110 (the first and the third ones are correct).

Since the contest would be over soon, Shik could only guess at most 300 times. Could you help Shik to find the answers of the problem?

Interaction Protocol

First, your program should read an integer n in one line, indicating the number of test cases in the problem.

Then, your program should start guessing the answers by output a string in one line. The length of the string should be n and the string should consist of "`0`" and "`1`".

After making each guess, you need to flush the standard output (like `fflush(stdout)` in C/C++).

After each guess, there will be a result from the standard input. The result would be a line with an integer, indicating the output of the oracle. When the result is n, your program should be terminated.

- $1 \le n \le 1000$

Problem E. Count on the integers

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

Count on a tree is a series of problems on SPOJ online judge. All of them are counting problems on the tree structure. It's a very nice problemset.

Bobo is simply good at it. He solves all the COT problems on SPOJ. When he says that he is a master of counting problems confidently, Hanhan challenges him.

"Hey! Don't say that you're master of counting problems. You ONLY solve the COT problems. Let's try some more interesting problems such as Count on the integers."

What? You say you've no idea what "Count on the integers" is? Let me introduce it as the following:

The problem is about counting the number of χ pairs (x, y). We say a pair (x, y), where $x \leq y$, is a χ pair if and only if $x \oplus y$ has odd number of ones in its binary representation. For example, $(2, 15)$ is a χ pair because $2 \oplus 15 = 1101_2$, which has odd number of ones.

Expression $x \oplus y$ means applying bitwise exclusive or operation to integers x and y. The given operation exists in all modern programming languages, for example, in languages C++ and Java it is represented as "^", in Pascal — as "xor".

Hanhan thinks the problem is not interesting enough, so he makes the problem harder. He will add some new ranges $[l_i, r_i]$. For each adding operation, you should answer the number of χ pairs (x, y) where both x, y belong to $[l_1, r_1] \cup \cdots \cup [l_i, r_i]$.

"I think it's quite easy."

After about 1 second, Bobo comes a solution. Hanhan is so stupid that he can't check whether Bobo is correct. Can you help Hanhan solve the problem?

Input

The first line contains an integer N indicating the total number of adding operations. Following N lines each contains two integers l_i, r_i indicating the i-th range Hanhan adds.

- $1 \leq N \leq 10^5$
- $1 \leq l_i \leq r_i \leq 2^{32} - 1$

Output

Output N lines each contains an integer. The integer in the i-th line is the number of χ pairs (x, y) where both x, y belong to $[l_1, r_1] \cup \cdots \cup [l_i, r_i]$.

Examples

standard input	standard output
3 1 3 2 10 100 110	2 25 110

Problem F. Almost GCD

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

In mathematics, the greatest common divisor (gcd) of two or more integers, which are not all zero, is the largest positive integer that divides each of the integers. For example, the gcd of 8 and 12 is 4.

Shik is simply good at finding the gcd of a sequence. To be a master of the gcd problems, Shik has to solve the following problem:

Given a seqeunce of n integers $a_1, a_2, \ldots, a_n$. You are allowed to increase or decrease some elements by 1. Your task is to find the maximum gcd.

For example with seqeunce $(3, 5, 2, 10)$, the maximum gcd is 3 because we can make the sequence $(3, 6, 3, 9)$.

Shik writes a program to solve the problem. Could you help him to check whether he is correct?

Input

The first line contains an integer T indicating the total number of test cases.

For each test case, the first line contains an integer n indicating the length of the sequence. The second line contains a sequence of n integers $a_1, a_2, \ldots, a_n$.

- $2 \le n \le 10^5$
- $\sum n \le 10^5$
- $2 \le a_i \le 10^{18}$

Output

For each test case, output an integer in one line indicating the maximum gcd.

Example

standard input	standard output
2 4 3 5 2 10 6 3 14 15 9 2 6	3 3

Problem G. Valid Parenthesis Substrings

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

For a given string S consists of lowercase letters '`a`'-'`z`', we would like to convert each character to either '`(`' or '`)`', such that

1. It is a valid (or balanced) parenthesis string.
2. For each pair of corresponding `(` and `)`, they are both from the same lowercase letter.

For example, if $S =$ `aabaab`, then `()(())` is a valid conversion. However, neither `()()()` nor `()((((` is a valid conversion.

In this task, you are going to report the number of substrings $S[i..j]$ of a given string S, so that $S[i..j]$ has a valid conversion.

Input

The input consists of a string S.

- $1 \leq |S| \leq 1000000$.

Output

Output a number indicating the number of substrings that has a valid conversion.

Examples

standard input	standard output
aabaab	4
abcabcabc	0
aabbcc	6

Problem H. Mouse Trap

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

Recently, the mouse problem in Bobo's house is getting more serious. Dozens of mice can be seen running inside the living room, the kitchen, and even Bobo's room! Hence, he decided to build a magical mouse trap. The mouse trap is a **simple** polygon with vertices choosen from the N base points. If a mouse enters the polygon, it will be disintegrated by the force field, so Bobo is basically safe there.

Now Bobo wants to maximize the area of the mouse trap, so he will have larger living space. The optimal solution is clearly the convex hull of the N base points. However, Bobo hates convex polygons, so he will choose the **concave hull** — the maximum area non-convex simple polygon that encloses all N base points (All N base points should be inside or on the border of the polygon). Can you help him calculate its area?

A polygon is simple if no two non-adjacent edges intersect, and adjacent edges intersect at exactly the common vertex. A simple polygon is non-convex if at least one of its interior angles is strictly larger than $180°$.

Input

The first line contains an integer N indicating the number of base points. For the following N lines, the i-th line contains 2 integers x, y separated by space representing the coordinates of the i-th base point.

- $1 \le N \le 500\,000$
- $|x_i|, |y_i| \le 10^9$
- No two points coincide.

Output

The output should contain an integer $2A$, where A is the area of the concave hull. Since the coordinates of base points are integers, $2A$ is guaranteed to be an integer, too. If the concave hull doesn't exist, please output -1.

Examples

standard input	standard output
3 0 0 0 1 1 1	-1
4 0 0 1 2 1 -2 -2 0	8
7 0 0 4 0 4 3 3 3 2 2 1 3 0 3	23
5 0 0 1 0 2 0 0 2 1 2	-1

Problem I. LCM and GCD

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

Give you two positive integers N and M.

Please calculate $(\Pi_{X \in S} lcm(X_1, X_2, ..., X_n)^{gcd(X_1, X_2, ..., X_n)})$ mod $10^9 + 7$, where S is the set contains all sequence which the length is N and all number in it are postive integers between 1 and M.

Input

The first line contains an integer T indicating the total number of test cases.

For each test case, there is only one line contain two number N, M.

- $1 \leq T \leq 10^3$
- $1 \leq N \leq 10^9$
- $1 \leq M \leq 2 \times 10^5$
- the summation of M in all test cases is not larger than 2×10^5

Output

For each test case, output an integer in one line indicating the answer of this test case.

Example

standard input	standard output
2 1 3 5 5	108 574847666

Problem J. String Game

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

There is a game about four strings.

Give you four string $S1$, $T1$, $S2$, $T2$.

Both length of $S1$ and $T1$ are N and both length of $S2$ and $T2$ are M.

We denote the i-th character of a string s as $s[i]$.

the target of this game is change the string $S1$ to $T1$.

The game contains two phase.

In the first phase, you can do following operation as many time as you want:

- Choose two distinct integers i and j $(0 \leq i, j < N)$ and swap $S1[i]$ and $S1[j]$.

Then in the second phase, you can do following operation as many time as you want. But this time, you want to minimize the time you do it.

- Choose two integer i and j $(0 \leq i < N$ and $0 \leq j < M)$ satisfying $S1[i]$ equal to $T2[j]$ and swap $S1[i]$ and $S2[j]$.

If it's impossible to change $S1$ to $T1$, output -1. Otherwise, output the minimum times you need for the operations in the second phase.

Input

There are four lines. Each line contains a string. They are $S1$, $T1$, $S2$, and $T2$ in this order.

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq M \leq 2 \times 10^5$
- all string are consist of lowercase English letters.

Output

Output an integer in one line indicating the answer.

Example

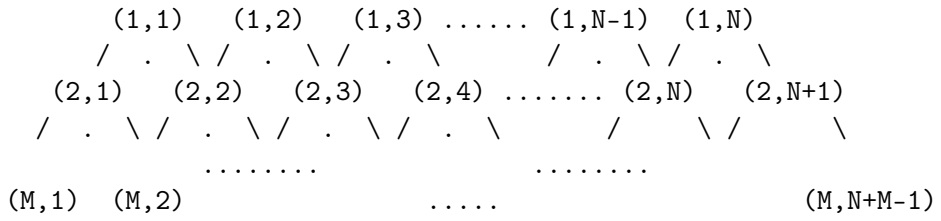
standard input	standard output
ae ed bcd abc	3
a b ab ab	-1

Problem K. Dropping Marbles

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

One day, Hanhan went to the night market as he had no other plans. There, he encountered Bobo the merchant, who's hosting a game stand. The rules are quite simple. On a pinball board, Bobo places several marbles on the top of the board initially. Then Hanhan makes a guess on the final positions of the marbles after dropping to the bottom. If Hanhan's guess is correct, he wins the jackpot and receives endless fortune to spend in the rest of his life.

To be more specific, the pinball board Bobo will play with looks like this.



Where (x, y) represents the y-th possible position in x-th row, and the single dots represent the pin. In the begining of each game, Bobo places K marbles in the first row. That is, the marbles are at position $(1, a_1), (1, a_2), \cdots, (1, a_K)$.

Before the game starts, Hanhan will choose K distinct integer $b_1, b_2, \cdots, b_K$ in $[1, N + M - 1]$ and bet that the marbles will fall to position $(M, b_1), (M, b_2), \cdots (M, b_K)$ at last.

After the game starts, the marbles would start falling at the same speed. That is, the marbles initially in row 1 will start to fall to row 2 and then 3, 4 ... Also, because of the pin, the marble would have equal probabilities to fall to the left or to the right. So, a marble at position $(1, 2)$ will fall to $(2, 2)$ or $(2, 3)$ both with probability 0.5. Eventually, all the marbles will fall to M-th row and stop. However, as the marbles fall, if any two of them collide with each other at any moment, one of the marbles bounces off the board and Hanhan will have no chance of winning.

Just like all other games in the night market, the chances of Hanhan winning the the jackpot is very slim. Even though, Hanhan would like to know what is the probability of him actually winning the jackpot given the initial placement of marbles and his guess. Of course, Hanhan is now seeking for your help. Can you find out the probability of Hanhan winning the jackpot?

Input

The first line contains an integer T indicating the total number of test cases. Each test case begins with a line containing 3 integers N, M, K separated by a space. In the second line, there are K integers a_i. In the third line, there are K integers b_i.

- $1 \le T \le 2000$
- $1 \le N, M \le 60$
- $1 \le K \le 25$
- a_i, b_i are given in ascending order

Output

Output an integer x representing the probability. Let the probability be $\frac{P}{Q}$, x should be $P \times Q^{-1} \mod 10^9 + 7$.

Examples

standard input	standard output
3 2 2 2 1 2 2 3 2 3 2 1 2 1 4 4 3 2 1 3 1 6	250000002 562500004 0

Problem L. Coin Tossing

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

Coin tossing is the practice of throwing a coin in the air and checking which side is showing when it lands to choose between two alternatives.

HH is bored and wants to play 10000 coin tossing games with you. At the beginning of each game, you need to guess the result (head or tail), and then HH would toss the coin. If you can successfully predict at least 95% of them, HH will be impressed by your luckiness and give you an "Accepted".

Here is the top secret about HH which might help you: He is controlled by skynet. By reverse engineering we found that he will toss coins according to the following C++ snippet.

```
#include <random>
struct HH {
    std::minstd_rand rnd;
    HH(int s) : rnd(s) {}

    int toss() {
        return rnd() % 2;  // 0 is head, 1 is tail
    }
};
```

Note that s is a secret number in range $[1, 2147483646]$, and the i-th number generated by `minstd_rand` would be $s \times 48271^i \bmod 2147483647$. For example, if $s = 7122$, the first 10 tosses would be `0100101001`.

Interaction Protocol

Your program should output each guess as "`0`" or "`1`" in one line. After making each guess, you need to make sure to flush standard output (like `fflush(stdout)` in C/C++). After each guess, there will be a result of toss to be read from standard input. The result would be a line with a single character "0" or "1".

After playing 10000 times, your program should exit normally without any redundant output.