

OLIMPIADA DE INFORMATICĂ ÎN ECHIPE – ETAPA NAȚIONALĂ

martie 2023

Base12

Input file: standard input
Output file: standard output
Time limit: 0.3 seconds
Memory limit: 128 megabytes

RANDy is now very interested in winning the grand prize at the new lotteries opened in his country. Because he has relations everywhere, he got a list of the winning numbers from all the N lotteries available. The winning numbers are expressed in base 12 (the numbers can begin with digit 0).

However, the lotteries could not agree on which letters (upper case English alphabet letters) should be used for 10 and 11, so each made its own choice. Even though he has such a great advantage, RANDy still doesn't know the rules to win the lottery. He thinks that in order to win the prize from a lottery he needs to guess only one digit from the winning number.

He doesn't know how to write, so help him learn the minimum number of digits in order to win (in his conception) at each lottery.

Input

The first line of the input contains a single number, N ($1 \leq N \leq 2.5 * 10^5$), representing the number of lotteries. Each of the next N lines contains a number written in base 12, representing the winning numbers.

The sum of lengths of all the numbers is at most $4 * 10^6$.

For test worth 13 points, all the numbers will be written in base 10.

For test worth 21 more points, all numbers will be written in base 12, but only with digits 10 and 11.

For test worth 27 more points, $1 \leq N \leq 5 * 10^3$.

Output

The only line of the output contains a single integer, K , representing the minimum number of digits RANDy has to learn how to write in order for him to believe he can win every lottery.

Example

standard input	standard output
4 AABAA B112 DAA 1777768CD	2

Note

RANDy can learn the following two digits: 1 and A. He can write "1" for lotteries 2 and 4 and "A" for lotteries 1 and 3. Thus he thinks that he wins at all of them.

OLIMPIADA DE INFORMATICĂ ÎN ECHIPE – ETAPA NAȚIONALĂ

martie 2023

Interval XOR 2

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

You became fond of the problem Interval XOR from the third round, so we decided to give you a more challenging version.

You are given an array of length n , which has positive integers, and q queries.

For each query, we will give the values l and r and we are required to solve the problem Interval XOR for the values in the range $[l, r]$ in the array. In other words, you will need to find the maximum XOR we can get by removing exactly one value from that interval.

Input

The first line of the input will contain n and q ($1 \leq n, q \leq 2 * 10^5$), the number of values in the array and the number of query operations your program will have to perform.

The next line of the array will have n values, the starting values of the array ($0 \leq v_i < 2^{30}$).

The next q lines of the array will contain 2 values each, respecting the format given earlier in the statement ($1 \leq l, r \leq n$).

For tests worth 14 points, $1 \leq n, q \leq 2 * 10^3$.

For tests worth 24 more points, $l = 1$ for all the queries.

Output

The output will contain a line for each operation given in the input.

Example

standard input	standard output
5 4	7
4 5 2 6 7	7
1 4	5
1 3	7
3 5	
1 5	

OLIMPIADA DE INFORMATICĂ ÎN ECHIBE – ETAPA NAȚIONALĂ

martie 2023

Love Journal

Input file: standard input
Output file: standard output
Time limit: 0.5 seconds
Memory limit: 64 megabytes

After Hayley realized that Essex is in fact not a continent, but a county, she decided to start a love journal with one of the fellow Love Islanders. Since they are not remarkably smart by nature, they forgot to add the spaces in between their daily entries in their journal and they now need your help to figure out how did their journal look in fact.

More formally, we are given a string of length n which contains only lowercase letters, digits and the character '.' (the dot).

We define a journal as a sequence of substrings which begin with a certain date (the format of the date is dd.mm), for example 01.01 or 16.11 are valid dates, contain a non-empty text and they end either at the end of the string, or when a new daily entry is added, which is the day after the day in which the previous entry started (for example, 15.03 is after 14.03 and 01.01 is after 31.12). The journal entry doesn't have to end the first time we find a new date in it.

Since the answer can be big, Hayley wants it printed modulo 666013 (she can't read big numbers).

Input

The input will contain on the first line n , the length of the string ($1 \leq n \leq 10^6$).

The second line will contain the string s , ($|s| = n$).

For tests worth 20 points, ($1 \leq n \leq 50$).

For tests worth 30 more points, ($1 \leq n \leq 5000$).

February is always considered to have 28 days in length.

Any pattern of type dd.mm will be followed by at least one letter, even if it can't be part of any valid date according to the journal chronology.

The string will start with a valid date.

Output

The output will contain on the first line the answer requested by Hayley.

Example

standard input	standard output
35 01.01ily02.01ily02.01xxox03.01acasa	5

Note

The journals we can form are

[01.01ily02.01ily02.01xxox03.01acasa]

[01.01ily; 02.01ily02.01xxox03.01acasa]

[01.01ily; 02.01ily02.01xxox; 03.01acasa]

[01.01ily02.01ily; 02.01xxox03.01acasa]

[01.01ily02.01ily; 02.01xxox; 03.01acasa]

OLIMPIADA DE INFORMATICĂ ÎN ECHIPE – ETAPA NAȚIONALĂ

martie 2023

Phone book

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 128 megabytes

Your friend, Ursu, gave a list of N words only containing lowercase letters of the English alphabet. The list represents Ursu's phone book. Unfortunately, Ursu's phone book is quite long. That is why he would like you to help him rename all his contacts. Since both of you are quite lazy, you want to minimize the time spent renaming all the contacts. Now, Ursu asks you to find for each of the N words the length of its shortest substring that does not appear in any of the other strings. If no such substring exists, Ursu would like to know that he cannot rename that contact, so, the answer will be considered to be -1 .

Input

The first line of the input contains the number N ($1 \leq N \leq 2 * 10^5$). The next N lines will each contain a string S .

The total length of all the strings (noted S) will be at most 10^6 .

For tests worth 13 points, $1 \leq S \leq 100$.

For tests worth 54 more points, $1 \leq S \leq 2 * 10^4$.

Output

The program will print N lines, the i -th of them representing the length of the shortest substring of the i -th string not appearing in any other string.

Example

standard input	standard output
3	2
abdf	2
abf	1
abde	

OLIMPIADA DE INFORMATICĂ ÎN ECHIBE – ETAPA NAȚIONALĂ

martie 2023

Squirrel

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 1024 megabytes

A squirrel discovered a deposit of peanuts. The deposit has a weirder shape, as it is made of 3 rows of rooms, each of them having 3 rooms. The deposit has in total 9 rooms situated on the 3 x 3 grid. We know the number of peanuts present in each of the 9 rooms from the matrix.

As the squirrel doesn't like to see too many peanuts in a room or too few peanuts in another room, it will proceed as follows: from day 1, the squirrel will take all the rooms in which a maximal number of peanuts exists and moves exactly one peanut to every adjacent room.

The squirrel is now asking you the following question:

"How many peanuts will there be in each room after n days?"

We need to answer to t such questions the squirrel will give to us.

Input

The first line of the input contains t , representing the number of tests we will need to solve ($1 \leq t \leq 100$).

Each testcase will contain 4 lines.

The first line of the test will contain n ($1 \leq n \leq 10^{18}$), the number of days corresponding to each step.

The following 3 lines of the test will contain the grid, each of the lines having 3 columns, representing the initial values of the matrix ($10 \leq a_{ij} \leq 10^{18}$).

For tests worth 14 points, ($1 \leq n \leq 100, 10 \leq a_{i,j} \leq 1000$).

For tests worth 7 more points, ($1 \leq n \leq 10^5$).

For tests worth 5 more points, ($10 \leq a_{i,j} \leq 15$).

For tests worth 14 more points, ($10 \leq a_{i,j} \leq 10^4$).

Output

Each test will have the 3 x 3 grid corresponding to the answer for that particular test case.

Example

standard input	standard output
4	16 17 14
2	14 13 16
15 14 18	13 16 15
16 15 13	13 13 13
11 18 14	12 13 15
3	15 13 13
11 14 13	12 14 14
13 12 15	14 14 11
14 16 12	13 12 14
4	12 12 13
16 12 14	11 14 11
13 14 12	13 13 11
11 14 12	
1	
11 12 13	
14 13 11	
12 13 11	

OLIMPIADA DE INFORMATICĂ ÎN ECHIBE – ETAPA NAȚIONALĂ

martie 2023

String Stack Removal

Input file: standard input
Output file: standard output
Time limit: 0.75 seconds
Memory limit: 256 megabytes

El Bandito Inofensivo gives a set of words S and a stack, you are asked to perform the following operations:

- operation of type 0: add the character c on top of the stack
- operation of type 1: pop the character from the top of the stack
- operation of type 2: insert the word W in S
- operation of type 3: erase the word W from S

We say that a word S of length N is on top of the stack if the suffix (we see the stack as a string, where the last element of the string is the top of the stack) of length N of the stack is equal to the string S . If there is a word from the set S at the top of the stack, it is removed instantly.

Initially, both the stack and the set of words are empty. You are given Q operations of the types above and you are asked to print the character on top of the stack after each of the Q operations. The character on top of the empty stack is considered to be the character '?'.

Input

First line of the input contains the number Q ($1 \leq Q \leq 1.5 \cdot 10^5$). The next Q lines describe the Q operations. Each of the lines contain a number op , representing the type of operation we have to perform:

- if $op = 0$, then the line also contains the character c . This line describes an operation of type 0.
- if $op = 1$, the line describes an operation on type 1. This operation will never show up when the stack is empty.
- if $op = 2$, then the line also contains the string W . This line describes an operation of type 2.
- if $op = 3$, then the line also contains the string W . This line describes an operation of type 3. This operation will never involve a word which is not in S .

For tests worth 13 points, $1 \leq Q \leq 100$.

For tests worth 38 more points, there is no operation of type 1 and all the operations of type 0 are performed after all the operations of type 2 and 3.

The sum of lengths of all words inserted in S is at most 10^5 .

The number of operations of type 0 is at most 10^5 .

It is guaranteed that, at any moment of time, S does not contain two words such that one is a suffix of the other.

Output

The only line of the output will contain a string S of length Q , where the i -th character from S represents the character on top of the stack after the i -th operation. If after the i -th operation the stack is empty, then the i -th character in S will be the character '?'.

Example

standard input	standard output
10 0 a 0 b 2 bbd 0 b 0 b 1 2 ab 0 d 3 ab 0 b	abbbbbbaab

Note

The evolution of the stack and the set of string after each operation:

After step 1: "a"

After step 2: "ab"

After step 3: "ab", ["bbd"]

After step 4: "abb", ["bbd"]

After step 5: "abbb", ["bbd"]

After step 6: "abb", ["bbd"]

After step 7: "abb", ["bbd", "ab"]

After step 8: "abbd" -> "a", ["bbd", "ab"]

After step 9: "a", ["bbd"]

After step 10: "ab", ["bbd"]