Bottle Necks

Problem Description

There are N bottles. ith bottle has A[i] radius. Once a bottle is enclosed inside another bottle, it ceases to be visible. Minimize the number of visible bottles.

You can put ith bottle into jth bottle if following condition is fulfilled:

1) ith bottle itself is not enclosed in another bottle.

2) jth bottle does not enclose any other bottle.

3) Radius of bottle i is smaller than bottle j (i.e. A[i] < A[j]).

Constraints

1 <= N <= 100000.

1 <= A[i] <= 10^18.

Input Format

First line contains a single integer N denoting the number of bottles.

Second line contains N space separated integers, ith integer denoting the radius of Ith bottle.

(1 <= i <= N).

Output

Minimum number of visible bottles.

Test Case

Explanation

Example 1

Input

8

1 1 2 3 4 5 5 4

Output

2

Explanation

1st bottle can be kept in 3 rd one 1-->2 , which makes following bottles visible [1,2,3,4,5,5,4]

similarly after following operations, the following will be the corresponding visible bottles

Operation ? Visible Bottles

2 ? 3 [1,3,4,5,5,4]

3 ? 4 [1,4,5,5,4]

4 ? 5 [1,5,5,4]

1 ? 4 [5,5,4]

4 ? 5 [5,5]

finally there are 2 bottles which are visible. Hence, the answer is 2

## Similar Char

### Problem Description

Tahir and Mamta are woking in a project in TCS. Tahir being a problem solver came up with an interesting problem for his friend Mamta.

Problem consists of a string of length N and contains only small case alphabets.

It will be followed by Q queries, in which each query will contain an integer P (1<=P<=N) denoting a position within the string.

Mamta's task is to find the alphabet present at that location and determine the number of occurrence of same alphabet preceding the given location P.

Mamta is busy with her office work. Therefore, she asked you to help her.

### Constraints

1 <= N <= 500000

S consisting of small case alphabets

1 <= Q <= 10000

1 <= P <= N

### Input Format

First line contains an integer N, denoting the length of string.

Second line contains string S itself consists of small case alphabets only ('a' - 'z').

Third line contains an integer Q denoting number of queries that will be asked.

Next Q lines contains an integer P (1 <= P <= N) for which you need to find the number occurrence of character present at the Pth location preceeding P.

### Output

For each query, print an integer denoting the answer on single line.

### Test Case

### Explanation

Example 1

Input

9

abacsddaa

2

9

3

Output

3

1

Explanation

Here Q = 2

For P=9, character at 9th location is 'a'. Number of occurrences of 'a' before P i.e., 9 is three.

Similarly for P=3, 3rd character is 'a'. Number of occurrences of 'a' before P. i.e., 3 is one.

## Prime Face

### Problem Description

Accept a number N up to 5 digits long in the positional numeral system formed by symbols 0, 1, ... 9, A, ..., Z. Also, accept another symbol S other than zero. Separate N and S with a space. Considering N to be represented in the least base possible between 2 and 36, identify the smallest prime number greater than or equal to N that contains at least one occurrence of S in it in base S + 1. (Refer example section for a better understanding). Prime number should be identified with respect to Base 10 i.e. a regular prime number.

### Constraints

1. Length of N <= 5

2. Max Base = 36

3. Face values for symbols:  
Symbol => Value in base 10  
0 => 0  
1 => 1  
2 => 2  
….  
9 => 9  
A => 10  
B => 11   
….  
Z => 35

### Input Format

One line containing two integers, N and S separated with space.

### Output

Print the smallest prime number greater than or equal to N that contains at least one occurrence of S in it, in base S + 1.

### Test Case

### Explanation

Example 1

Input

10 B

Output

B

Explanation

The least possible base for N is 2 and its value in that base is 2. We want the smallest prime number in base 12 (1 more than the face value of B, 11) that contains symbol B and is greater than or equal to 2. The first few numbers in ascending order in base 12 containing face value B are B (value 11), 1B (value 1 \* 12 + 11 = 23), 2B (value 2 \* 12 + 11 = 35): of these the smallest number that is prime is 11, which is greater than N. Hence, the output is B.

Example 2

Input

ZZ Z

Output

11Z

Explanation

The least possible base for N is 36 and its value in that base is 35 \* 36 ^1 + 35 = 1295. The first few numbers in ascending order in base 36 (1 more than the face value of Z, 35) containing face value Z and greater than N are 10Z (1 \* 36^2 + 0\*36^1 + 35 = 1331, non-prime), 11Z (1 \* 36^2 + 1 \* 36^1 + 35 = 1367, a prime). Hence, the output is 11Z.

## Lazy Student

### Problem Description

There is a test of Algorithms. Teacher provides a question bank consisting of N questions and guarantees all the questions in the test will be from this question bank. Due to lack of time and his laziness, Codu could only practice M questions. There are T questions in a question paper selected randomly. Passing criteria is solving at least 1 of the T problems. Codu can't solve the question he didn't practice. What is the probability that Codu will pass the test?

### Constraints

0 < T <= 10000

0 < N, T <= 1000

0 <= M <= 1000

M,T <= N

### Input Format

First line contains single integer T denoting the number of test cases.

First line of each test case contains 3 integers separated by space denoting N, T, and M.

### Output

For each test case, print a single integer.

If probability is p/q where p & q are co-prime, print (p\*mulInv(q)) modulo 1000000007, where mulInv(x) is multiplicative inverse of x under modulo 1000000007.

### Test Case

### Explanation

Example 1

Input

1

4 2 1

Output

500000004

Explanation

The probability is ½. So output is 500000004.

## Death Battle

### Problem Description

In a crossover fantasy universe, Houin Kyoma is up in a battle against a powerful monster Nomu that can kill him in a single blow. However being a brilliant scientist Kyoma found a way to pause time for exactly M seconds. Each second, Kyoma attacks Nomu with certain power, which will reduce his health points by that exact power. Initially Nomu has H Health Points. Nomu dies when his Health Points reach 0. Normally Kyoma performs Normal Attack with power A. Besides from Kyoma’s brilliance, luck plays a major role in events of this universe. Kyoma’s Luck L is defined as probability of performing a super attack. A super attack increases power of Normal Attack by C. Given this information calculate and print the probability that Kyoma kills Nomu and survives. If Kyoma dies print “RIP”.

### Constraints

0 < T <= 50

1 <= A, H, C, L1, L2 <= 1000

1 <= M <= 20.

L1<=L2

### Input Format

First line is integer T denoting number of test cases.

Each test case consist of single line with space separated numbers A H L1 L2 M C. Where luck L is defined as L1/L2. Other numbers are, as described above.

### Output

Print probability that Kyoma kills Nomu in form P1/P2 where P1<=P2 and gcd(P1,P2)=1. If impossible, print “RIP” without quotes.

### Test Case

### Explanation

Example 1

Input

2

10 33 7 10 3 2

10 999 7 10 3 2

Output

98/125

RIP