20-21 своя тренировка 2: Semi-FFT

A. Lightsabers (hard)

4 seconds, 256 megabytes

There used to be unrest in the Galactic Senate. Several thousand solar systems had declared their intentions to leave the Republic. But fear not! Master Heidi was able to successfully select the Jedi Knights that have restored peace in the galaxy. However, she knows that evil never sleeps and a time may come when she will need to pick another group of Jedi Knights. She wants to be sure she has enough options to do so.

There are n Jedi Knights, each of them with a lightsaber of one of m colors. Given a number k, compute the number of differently colored collections of k lightsabers that some k Jedi Knights might have. Jedi Knights with lightsabers of the same color are indistinguishable (it's not the person, it's the lightsaber color that matters!), and their order does not matter; that is, we consider two collections of Jedi Knights to be different if and only if their vectors of counts of lightsabers of each color (like what you were given in the easy and the medium versions) are different. We count all subsets, not only contiguous subsegments of the input sequence. Output the answer modulo 1009.

Input

The first line of the input contains n ($1 \le n \le 2\cdot 10^5$), m ($1 \le m \le n$) and k ($1 \le k \le n$). The second line contains n integers in the range $\{1, 2, ..., m\}$ representing colors of the lightsabers of subsequent Jedi Knights.

Output

Output one number: the number of differently colored collections of k lightsabers modulo 1009.

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

B. И снова равные строки

4 секунды, 256 мегабайт

Пусть вам даны две строки s и t, состоящие из латинских букв от `a` до `f`. Длины этих строк равны. Разрешено проводить над ними произвольное количество раз операцию вида: выбрать два различных символа c_1 и c_2 и заменить все вхождения символа c_1 в обеих строках на c_2. Определим *расстояние* между строками s и t, как минимальное количество раз, которое нужно применить операцию, чтобы строки стали равными. Например, если s равна `abcd` и t равна `ddcb`, *расстояние* между ними равно 2 — можно заменить все вхождения символа `a` на `b`, превратив s в `bbcd` и заменить все `b` на `d`, строки станут равны `ddcd`.

Вам даны две строки S и T. Для каждой подстроки S, состоящей из $|T|$ символов, определите *расстояние* между этой подстрокой и T.

Входные данные

Первая строка содержит S, вторая — T ($1 \le |T| \le |S| \le 125000$). Обе строки состоят из строчных латинских букв от `a` до `f`.

Выходные данные

Выведите $|S|$ - $|T| + 1$ целых чисел. Число на позиции i должно равняться *расстоянию* между подстрокой S, начинающейся в позиции i, длины $|T|$ и строкой T.

входные данные
```
abcdefa
ddcb
```
выходные данные
```
2 3 3 3
```

C. Basis Change

2 seconds, 512 mebibytes

You are given sequences $\{a_i\}_{i=1}^k$ and $\{b_i\}_{i=1}^k$. Consider all sequences $\{F_i\}_{i=1}^\infty$ which satisfy the following linear recurrence for all $n > k$:

$$F_n = \sum_{i=1}^{k} a_i F_{n-i}.$$

You have to find a sequence $\{c_i\}_{i=1}^k$ such that, for all such $\{F_i\}_{i=1}^\infty$, the following linear recurrence is satisfied for all $n > b_k$:

$$F_n = \sum_{i=1}^{k} c_i F_{n-b_i}.$$

Input

The first line of input contains a single integer k ($1 \le k \le 128$).

The second line of input contains k integers $a_1, \ldots, a_k$ ($1 \le a_i \le 10^9$).

The third line of input contains k integers $b_1, \ldots, b_k$ ($1 \le b_1 < b_2 < \cdots < b_k \le 10^9$).

It is guaranteed that the solution exists and is unique. Moreover, it is guaranteed that sequences a_i and b_i were uniformly randomly chosen among possible ones with some fixed number k for all test cases except the example.

Output

Output k integers $c_1, \ldots, c_k$ on a single line. If $c_k = \frac{P}{Q}$ such that P and Q are coprime, output $(P \cdot Q^{-1}) \bmod (10^9 + 7)$. It is guaranteed that $Q \not\equiv 0 \pmod{10^9 + 7}$.

input
```
2
1 1
1 3
```
output
```
2 1000000006
```

In the example, we have $F_n = F_{n-1} + F_{n-2}$. We can write $F_n - F_{n-1} = (F_{n-1} + F_{n-2}) - (F_{n-2} + F_{n-3})$. Thus $F_n = 2F_{n-1} - F_{n-3}$.

D. Do Use FFT

10 seconds, 1024 megabytes

You are given integer sequences A, B, and C, each of length N. For each $k = 1, 2, \ldots, N$, find the following value modulo 998244353.

$$\sum_{1 \le i \le N} \left(C_i \times \prod_{1 \le j \le k} (A_i + B_j) \right)$$

Input

The first line contains an integer N ($1 \le N \le 250000$).

The second line contains N integers $A_1, A_2, \ldots, A_N$ ($0 \le A_i < 998244353$).

The third line contains N integers $B_1, B_2, \ldots, B_N$ ($0 \le B_i < 998244353$).

The fourth line contains N integers $C_1, C_2, \ldots, C_N$ ($0 \le C_i < 998244353$).

Output

For each $k = 1, 2, \ldots, N$, print the answer.

input
```
3
1 2 3
4 5 6
7 8 9
```
output
```
146 1050 8694
```

E. Find the LCA

7 seconds, 1024 megabytes

You are given an integer sequence $A_1, A_2, \ldots, A_N$. You'll make a rooted tree with N vertices numbered from 1 through N. The vertex 1 is the root, and for each vertex i $(2 \le i \le N)$, its parent p_i must satisfy $p_i < i$.

You define the score of a rooted tree as follows:

- Let x be the lowest common ancestor of the vertex $N - 1$ and the vertex N. Then, the score is

$$\prod_{v \in (\text{subtree rooted at } x)} A_v$$

Note that we consider x itself is in the subtree rooted at x.

There are $(N-1)!$ ways to make a tree. Find the sum of scores of all possible trees, modulo 998244353.

Input

The first line contains an integer N $(3 \le N \le 250000)$.

The second line contains integers $A_1, A_2, \ldots, A_N$ ($1 \le A_i < 998244353$).

Output

Print the answer.

input
```
3
2 2 2
```
output
```
12
```

input
```
5
1 2 3 4 5
```
output
```
2080
```

F. Cyclic Shift

1 second, 256 megabytes

You are given two strings a and b of the same length and consisting of lowercase English letters. You can pick at most one subsequence of string b and do a cyclic shift on that subsequence exactly once.

For example, if you have a string "`abcdefg`" and you picked the letters at indices 2, 5, and 6 as a subsequence to do a cyclic shift on them, the letter at index 2 will go to index 5, the letter at index 5 will go to index 6, the letter at index 6 will go to index 2, and the string will become "`afcdbeg`".

Your task is to check if it is possible to make string b equivalent to string a using at most one cyclic shift. Can you?

Input

The first line contains an integer T $(1 \le T \le 200)$ specifying the number of test cases.

The first line of each test case contains an integer n $(1 \le n \le 10^5)$ specifying the length of strings a and b. Then two lines follow, giving strings a and b, respectively. Both strings consist only of lowercase English letters.

Output

For each test case, print a single line containing "YES" (without quotes) if it is possible to make string b equivalent to string a using at most one cyclic shift. Otherwise, print "NO" (without quotes).

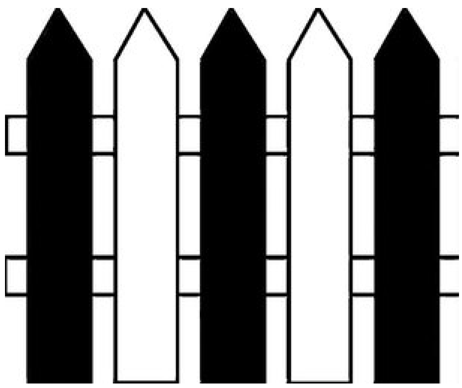
input
```
2
6
abcdef
adcebf
4
abcd
dabd
```
output
```
YES
NO
```

A subsequence is a sequence that can be derived from another sequence by deleting some or no elements without changing the order of the remaining elements. For example, the sequence $\langle A, B, D \rangle$ is a subsequence of $\langle A, B, C, D, E, F \rangle$ obtained after removal of elements C, E, and F.

G. Wooden Fence

1 second, 256 megabytes

Asem has a wooden fence consisting of n boards, in which n is an odd number. Asem wants to paint this fence using two colors; black and white, such that the first board will be painted black, the second board will be painted white, the third board will be painted black, and so on.



Asem has black paint that can paint at most x boards and white paint that can paint at most y boards. Your task is to determine if you can paint the whole fence or not. Can you?

Input

The first line contains an integer T $(1 \le T \le 10^4)$ specifying the number of test cases.

Each test case consists of a line containing three integers n, x, y $(1 \le n, x, y < 10^9)$, in which n is an odd number specifying the number of boards in the fence, x is the maximum number of boards that can be painted in black, and y is the maximum number of boards that can be painted in white.

Output

For each test case, print a single line containing "YES" (without quotes) if you can paint the whole fence. Otherwise, print "NO" (without quotes).

input
```
3
5 3 3
7 2 3
9 6 2
```

output

```
YES
NO
NO
```

H. Multiplication Dilemma

1 second, 256 megabytes

Multiplication operation is not always easy! For example, it is hard to calculate 27×20 using your mind, but it is easier to find the answer using the following methods: 30×20 - 3×20. It turns out that people can calculate the multiplication of two special numbers very easily.

A number is called special if it contains exactly one non-zero digit at the beginning (i.e. the most significant digit), followed by a non-negative number of zeros. For example, 30, 7, 5000 are special numbers, while 0, 11, 8070 are not.

In this problem, you are given two numbers a and b. Your task is to calculate the multiplication of a and b ($a \times b$), by writing the multiplication expression as a summation or subtraction of multiplication of special numbers. Can you?

Input

The first line contains an integer T $(1 \le T \le 10^4)$ specifying the number of test cases.

Each test case consists of a single line containing two integers a and b $(-10^9 \le a, b \le 10^9, a \ne 0, b \ne 0)$, as described in the problem statement above.

Output

For each test case, print a single line containing the multiplication expression of a and b as a summation or subtraction of multiplication of special numbers. All special numbers must be between -10^9 and 10^9 (inclusive). If there are multiple solutions, print any of them. It is guaranteed that an answer always exists for the given input.

The multiplication expression must be printed in exactly one line. A single term must be printed as $z\#x\#w$ in which z and w are both special numbers, # represents a single space, and x represents the multiplication operation. Two consecutive terms must be printed as $z\#x\#w\#o\#z\#x\#w$ in which z and w are both special numbers, # represents a single space, x represents the multiplication operation, and o represents either the addition operation $+$ or the subtraction operation $-$. (**Check the sample output for more clarification**).

input

```
2
55 20
70 17
```

output

```
60 x 20 - 5 x 20
-3 x 70 + 70 x 20
```

I. Beautiful Substrings

1 second, 256 megabytes

You are given two strings a and b consisting of lowercase English letters. A beautiful substring is defined as a substring of any length of string b such that the first and last letters of it are the same as the first and last letters of any substring of length k of string a.

Your task is to count the number of beautiful substrings. Can you?

Input

The first line contains an integer T $(1 \le T \le 100)$ specifying the number of test cases.

The first line of each test case contains three integers n, m, and k $(1 \le n, m \le 10^5, 1 \le k \le n)$, in which n is the length of string a, m is the length of string b, and k is the described variable in the statement.

Then two lines follow, the first line contains a string a of length n and the second line contains a string b of length m. Both strings consist only of lowercase English letters.

Output

For each test case, print a single line containing the number of beautiful substrings

input

```
2
4 5 3
acbd
abcbd
3 3 1
kkd
dkd
```

output

```
3
4
```

A substring of a string s is a sequence $s_l, s_{l+1}, \ldots, s_r$ for some integers (l, r) such that $(1 \le l \le r \le n)$, in which n is the length of the string s.

J. Stupid Submissions

1.5 seconds, 256 megabytes

Abed is so motivated this year, his goal is qualifying for the 2019 ACM International Collegiate Programming Contest (ACM ICPC). Therefore, he always trains using Codeforces online judge. If you do not know Codeforces, the following rules can help you to understand how it works:

- Each problem has a set of n tests numbered from 1 to n. The first k tests are called sample tests. These tests are visible in the problem statement and a user can see them all.
- When Abed gets a wrong answer on test x $(x \le n)$, he can see all the tests up to test number x. If Abed gets accepted, he can see all the tests.
- Each test can be either small or big. Abed can see the entire test only if it is small. If the test is big, Abed can see it partially. All the sample tests are small.

For example, let us consider a problem contains 6 tests, in which the first two tests are the sample tests. If Abed got a wrong answer on test 4, a wrong answer on test 3, and a wrong answer on test 5, he can see all tests from 1 to 5. If test 3 is small and the tests from 4 to 6 are big, Abed can see the first three tests fully, but he cannot fully see the remaining tests.

Unfortunately, Abed usually gets a lot of wrong answers. A submission made by Abed is called stupid if he got a wrong answer on a small test that he can see it fully. You are given a list of submissions made be Abed, your task is to count how many stupid submission Abed has made. Can you?

Input

The first line contains an integer T $(1 \le T \le 100)$ specifying the number of test cases.

The first line of each test case contains three integers n, m, and k $(1 \le k \le n \le 10^4, 1 \le m \le 10^4)$, in which n is the number of tests in the problem, m is the number of submissions Abed has made, and k is the number of sample tests. Then a line follow contains n characters $t_1, \ldots, t_n$, in which t_i is 'S' if the i^{th} test is small, and 'B' if it is big.

Then m lines follow, giving the list of submissions Abed has made, such that the i^{th} line will either contain a single character 'A' if Abed got accepted on the i^{th} submission, or contain a character 'W' and an integer x $(1 \le x \le n)$ giving that Abed got a wrong answer on test case x.

Output

For each test case, print a single line containing the number of stupid submissions Abed has made.

input	output
2 5 4 2 S S B S B W 3 W 4 W 4 A 4 3 2 S S B B W 1 W 2 W 3	1 2

Codeforces (c) Copyright 2010-2021 Михаил Мирзаянов
Соревнования по программированию 2.0