

Caderno de Provas



O Poderoso Balão

I Seletiva Interna UDESC 2025

Servidor BOCA:

<https://boca.joinville.udesc.br/>

Organização e Realização:



Patrocínio:

BECOMEX

Informações do ambiente de testes

- Em todas as questões, o limite de memória é de 1 gigabyte.
- Há um limite no tamanho na pilha de 100 megabytes.
- O código-fonte submetido pode ter no máximo 100 kilobytes.

Comandos de compilação utilizados

- C: `gcc -static -O2 <nome_do_arquivo>.c -lm -o <nome_do_programa>`
- C++: `g++ -std=c++20 -static -O2 -lm <nome_do_arquivo>.cpp -o <nome_do_programa>`
- Java: `javac <nomeDoArquivo>.java`
- Kotlin: `kotlinc -J-Xms1024m -J-Xmx1024m -J-Xss100m -include-runtime <nome_do_arquivo>.kt`

Comandos de entrada e saída de dados

- C: `scanf`, `getchar`, `printf`, `putchar`
- C++: as mesmas de C ou os objetos `cout` e `cin`
- Java: `Scanner`, `BufferedReader`, `BufferedWriter` e `System.out.println`
- Python: `read`, `readline`, `readlines`, `input`, `print`, `write`

Template exemplo

A correção é automatizada, portanto siga atentamente as exigências do problema quanto ao formato da entrada e saída de seu programa. Fornecemos um exemplo de problema e um template esperado para a sua resolução:

A entrada consiste em uma linha contendo um inteiro A.

A saída consiste de uma linha contendo o inteiro B, que é igual ao inteiro A multiplicado por 2.

- C:

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    int A;
    scanf("%d", &A);
    int B = A * 2;
    printf("%d\n", B);
    return 0;
}
```

- C++:

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int A;
    cin >> A;
    int B = A * 2;
    cout << B << "\n";
    return 0;
}
```

- Java:

```
import java.io.IOException;
import java.util.Scanner;

public class C {
    public static void main(String[] args) throws IOException {
        Scanner scan = new Scanner(System.in);
        int A;
        A = scan.nextInt();
        int B = A * 2;
        System.out.printf("%d\n", B);
    }
}
```

Obs: o nome do arquivo deve ser LETRA_DO_PROBLEMA.java e a classe deve ser LETRA_DO_PROBLEMA, como no exemplo acima, se o problema solucionado fosse o problema C.

- Python:

```
A = int(input())
B = A * 2;
print(B)
```

Obs: Caso a entrada contenha vários elementos em uma única linha, como por exemplo "uma linha contendo três inteiros A, B e C" usamos o seguinte comando:

```
A, B, C = map(int, input().split())
```

C/C++

- Seu programa deve retornar zero, executando, como último comando, `return 0` ou `exit(0)`.
- É sabido que em alguns casos de problemas com entradas muito grandes, os objetos `cin` podem ser lentos, por conta da sincronização de buffers da biblioteca `stdio`. Caso deseje utilizar `cin` e `cout`, um jeito de se contornar este problema é executando o comando `ios_base::sync_with_stdio(0)`, no início de sua função `main`. Note que, neste caso, o uso de `scanf` e `printf` no mesmo programa é contra-indicado, uma vez que, com buffers separados, comportamentos inadequados podem ocorrer.
- Comando para executar uma solução C/C++: `./<nome_do_programa>`.

Java

- Não declare “**package**” no seu programa Java.
- Note que a convenção para o nome do arquivo fonte deve ser obedecida, o que significa que o nome de sua classe pública deve ser uma letra maiúscula (A, B, C, ...).
- Comando para executar uma solução Java: `java -Xms1024m -Xmx1024m -Xss20m <nomeDoArquivo>`.

Kotlin

- Não declare “**package**” no seu programa Kotlin.
- Comando para executar uma solução Kotlin: `kotlin -J-Xms1024m -J-Xmx1024m -J-Xss100m <nomeDoArquivoKt>`.
- Não é garantido que soluções em Kotlin conseguirão executar dentro do tempo limite alocado.

Python

- O Python está na versão 3.10 e em erro de sintaxe, será retornado **Runtime Error**.
- Comando para executar uma solução Python: `python3 <nome_do_arquivoPy>.py`.
- Não é garantido que soluções em Python conseguirão executar dentro do tempo limite alocado.

Instruções para uso do sistema de submissão BOCA

Tudo no BOCA é feito pela interface web, então antes de fazer qualquer ação certifique-se que:

- O navegador está aberto e na URL do BOCA que se encontra na capa.
- O BOCA está logado com o login do time (com o nome de usuário e senha fornecidos).

Submissão de soluções

Para enviar uma solução para um problema, acesse a aba **Runs**, escolha o problema apropriado, a linguagem utilizada e envie o arquivo fonte.

Resultado da submissão

Para ver o resultado de uma submissão, verifique a aba **Runs**. Os vereditos que você pode receber dos juízes são:

- | | |
|-----------------------------|-----------------------------|
| 1. YES | 6. NO - Contact staff |
| 2. NO - Compilation error | 7. NO - Class name mismatch |
| 3. NO - Runtime error | 8. NO - Wrong language |
| 4. NO - Time limit exceeded | 9. NO - Problem mismatch |
| 5. NO - Wrong answer | |

Os significados de 1, 2, 3 e 4 são auto-explicativos.

- Sobre 1 e 5:
 - Se a saída da solução do time é exatamente igual à saída dos juízes, a resposta é “YES”;
 - Caso contrário, o veredito é “Wrong Answer”.
- Sobre 6: Usado em circunstâncias inesperadas. Neste caso, utilize o menu “Clarifications” e forneça o número da run para maiores esclarecimentos.
- Sobre 7: Apenas para submissões Java, veredito quando o time submete uma solução com nome da classe principal diferente do especificado, de forma que a execução falha. Não é usado no caso de submissões C/C++ ou Python.
- Sobre 8: Principalmente para Python3, mas pode ser usado também para outras linguagens.
- Sobre 9: Identificação errada de problema ao submeter a solução. Note que nem sempre é possível distinguir entre os vereditos 3, 7, 8 e 9. Por exemplo, quando um time submete um arquivo `B.java` no qual uma classe pública `A` é definida como solução para o problema `A`, o veredito pode ser “Class name mismatch” ou “Problem mismatch”.

Esclarecimentos

Para solicitar esclarecimentos sobre o enunciado de um problema, acesse a aba **Clarifications**, escolha o problema apropriado, escreva sua dúvida e submeta.

Placar

Para ver o placar local você deve acessar a aba **Score**.

O Poderoso Balão

I Seletiva Interna UDESC 2025

07 de Junho de 2025

Conteúdo

Problema A: Absolute Cinema	7
Problema B: Magic Library	9
Problema C: Clichés	11
Problema D: Inside the Matrix	13
Problema E: Enigma	15
Problema F: Frodo & Sam	16
Problema G: Godfather	18
Problema H: Cinematic Hierarchy	20
Problema I: Lost Language	22
Problema J: John Wick	23
Problema K: Kill Bill Vol. 1	24
Problema L: Las Tortuguitas	26
Problema M: Movieguessr	28
Problema N: Ni	30
Problema O: The Martian	31

Problema A: Absolute Cinema

Arquivo: A. [c|cpp|java|kt|py]
 Limite de tempo: 1 second

After the unforgettable episode with the El Café coffee shop, Max and his brother Min decided to take a well-deserved vacation — this time, away from coffee and closer to the big screen. Their love for cinema, especially the films of Martin Scorsese, led them to create a new tradition: **continuous Scorsese marathons**.

During their month of rest, the brothers organized N of the director's films in chronological order, numbering them from 1 to N . Each film received a score from film critics; the score of the i -th film is a_i .

Being as competitive and creative as always, Max and Min decided to evaluate the quality of each “continuous marathon” in a curious way:

- Max, unsurprisingly, cares about the maximum.
- Min, of course, is interested in the minimum.
- For them, the quality of a marathon is defined as the product of the maximum and the minimum values in the sequence of scores of the films watched in that marathon.

A continuous marathon is any contiguous subsequence of the original list of films — it can be a single film or all N films.

Naturally, Max and Min became curious: what is the total sum of the qualities of all possible continuous marathons?

Since this value can be very large, they decided to only care about the result modulo $10^9 + 7$.

Entrada

The first line contains an integer N ($1 \leq N \leq 10^5$), representing the number of films in the list organized by Max and Min. The second line contains N integers $a_1, a_2, \dots, a_N$ ($1 \leq a_i \leq 10^9$), representing the scores given by film critics to each film.

Saída

Print a single integer, which is the remainder of the division of the total sum of the qualities of all possible continuous marathons in Max and Min's list by $10^9 + 7$.

Exemplos

standard input	standard output
3 2 1 3	22
5 5 9 1 2 3	230

Notas

Explanation of example 1: For each contiguous subsequence (marathon), we compute the quality as the product of the maximum and minimum score values among the films in that marathon:

$$2 \rightarrow \text{Max} = 2, \text{Min} = 2, \text{Quality} = 2 \cdot 2 = 4$$

$$2,1 \rightarrow \text{Max} = 2, \text{Min} = 1, \text{Quality} = 2 \cdot 1 = 2$$

$$2,1,3 \rightarrow \text{Max} = 3, \text{Min} = 1, \text{Quality} = 3 \cdot 1 = 3$$

$$1 \rightarrow \text{Max} = 1, \text{Min} = 1, \text{Quality} = 1 \cdot 1 = 1$$

$$1,3 \rightarrow \text{Max} = 3, \text{Min} = 1, \text{Quality} = 3 \cdot 1 = 3$$

$$3 \rightarrow \text{Max} = 3, \text{Min} = 3, \text{Quality} = 3 \cdot 3 = 9$$

The sum of all qualities is $4 + 2 + 3 + 1 + 3 + 9 = 22$.

Problema B: Magic Library

Arquivo: B. [c|cpp|java|kt|py]
 Limite de tempo: 2 seconds

At the School of Magic and Witchcraft of Brutewarts, there is a library containing millennia-old magical knowledge. In the library, there are several bookshelves, each dedicated to a different book category. The librarian, Karina McGonagall, is very meticulous about the indexing of books on the shelves, and her favorite category is the one about the Three Little Pigs.

In a given category, she likes to keep the books sorted by their titles in lexicographic order—that is, the same order as they would appear in a dictionary. For instance, the word “abacate” comes before “abacaxi” in the dictionary, since when we look at the first position where they differ, the letter “t” comes before the letter “x” in the alphabet.

After a lot of effort, Karina finally finished organizing her most ambitious bookshelf: the shelf of all finite strings containing letters from “a” to “z”. Although there are some books with repeated titles, books titled with every imaginable finite string are already there, perfectly sorted lexicographically.

On this shelf, she noticed that magic students almost always read one of the first books, located near the beginning, and it was rare for anyone to want to reach the end.

Karina’s favorite book title is S , and she would like to reposition it in a place where it will be more frequently read. Since she cannot change the ordering, she only wants to **slightly modify** the title so that, when reinserted in its correct position, it will be as close to the beginning as possible. To avoid harming comprehension, she will make the following change **at most once**:

- Choose two positions i, j ($1 \leq i, j \leq N$) and swap the characters S_i and S_j , that is, S_i goes to position j and vice versa.

As she is already very tired after grading Transfiguration exams, the poor librarian asks for your help.

Given the title of the book, what should its new title be in order to meet Karina’s request?

Entrada

The input consists of two lines. The first contains an integer N ($1 \leq N \leq 10^5$), the length of Karina’s favorite book title. The second line contains the title S , a word consisting only of lowercase letters.

Saída

Print the new book title after fulfilling Karina’s request.

Exemplos

standard input	standard output
8 discreta	aiscreted
11 harrypotter	ahrrypotter
10 crepusculo	ccepusrulo
3 abc	abc

Problema C: Clichés

Arquivo: C. [c|cpp|java|kt|py]

Limite de tempo: 1.5 seconds

Queliton Tarantino is a legendary figure in the world of cinema; his films are known for their sharp dialogue, spectacular plot twists, and above all, for using every possible cliché — always with that ironic touch.

Recently, Queliton has been working on the script for his new masterpiece: a film encoded as a word S , composed only of lowercase letters. Like any impactful screenwriter, he wants to make sure his film is full of classic cinematic clichés — each represented by a word.

But there's a twist: Queliton wants to make one bold move — reversing, at most once, a continuous segment of the film. He knows this trick can turn an ordinary scene into an iconic moment, creating new occurrences of the clichés and, who knows, securing his tenth Palme d'Or.

Your task is to help him determine the maximum number of cliché occurrences he can insert into the film after reversing, at most once, any continuous segment of the script.

Each occurrence counts, even if the same cliché appears multiple times, including overlapping with other occurrences or with itself.

Entrada

The first line contains the word S ($1 \leq |S| \leq 2000$), the current script of the film.

The second line contains an integer M ($1 \leq M \leq 2000$) — the number of cinematic clichés.

Each of the next M lines contains a word P_i ($1 \leq |P_i| \leq 2000$), representing a cliché.

The total sum of the lengths of all strings P_i does not exceed 2000.

Saída

Print a single integer: the maximum number of cliché occurrences Queliton can achieve after reversing, at most once, any continuous segment of S .

Exemplos

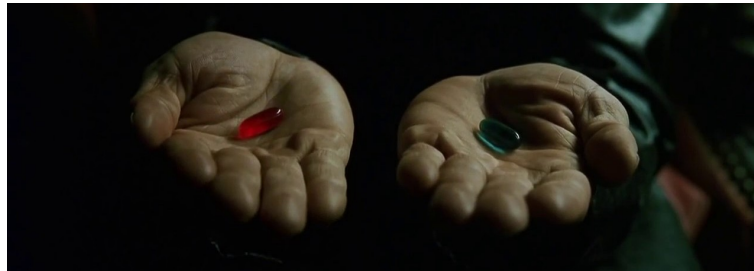
standard input	standard output
abbaca 3 cab bbac aa	2
abcda 2 da bc	2
aaaaaabaa 1 abaaa	1
aabababb 4 ab bab abab baba	12

Problema D: Inside the Matrix

Arquivo: D. [c|cpp|java|kt|py]

Limite de tempo: 1 second

Lucas Thomas Anderson used to work in an office and lived a normal life, except for the fact that he was a hacker in his spare time and discovered he is trapped inside an $N \times M$ matrix! He is currently at cell $(1, 1)$ and, to escape the matrix, he needs to reach Morpheus, who is located at cell (N, M) .



However, Agent Gui also wants to find Morpheus to thwart Thomas's plans before he escapes the matrix. To do this, Gui alters reality, creating a challenge centered around the bitwise OR operation, because Gui knows that although Thomas is an experienced hacker, he is more familiar with other operations.

In this reality altered by Agent Gui, the time anyone takes to walk through a path in the matrix is the bitwise OR of the values in the cells traversed. Also, since Gui is an AI-powered program, he can easily traverse the matrix and find the fastest route to wherever he wants to go.

Now, Agent Gui is at cell (X, Y) and wants to reach Morpheus before Thomas. Gui is not in the same position as Thomas or Morpheus. Both Gui and Thomas can move from one cell to another if the two share an edge (i.e., up, down, left, or right).

The fate of humanity is in your hands! Given Agent Gui's position and the value of each cell, write a program that tells Thomas the minimum time he can take to reach Morpheus and also the time it would take Gui.

Entrada

The first line of input contains two integers N and M ($3 \leq N \cdot M \leq 10^5$), the dimensions of the matrix.

The second line contains two integers X and Y , Agent Gui's position in the matrix.

Next, N lines follow. The i -th of these lines ($1 \leq i \leq N$) contains M values, and the j -th value in it represents the value in cell (i, j) of the matrix ($0 \leq V_{i,j} \leq 2 \cdot 10^9$).

It is guaranteed that the values of the cells where Thomas, Gui, and Morpheus are located are 0.

Saída

Print two values: the time Thomas and Gui take to reach Morpheus.

Exemplos

standard input	standard output
3 3 1 3 0 2 0 1 5 6 1 1 0	1 3
3 4 3 1 0 2 4 8 4 8 2 0 0 2 3 0	6 3
3 4 3 1 0 2 3 8 4 8 2 0 0 2 4 0	3 6
3 3 1 2 0 0 1 1 1 1 1 1 0	1 1

Problema E: Enigma

Arquivo: E. [c|cpp|java|kt|py]
 Limite de tempo: 1 second

Leo Turing is a scientist with incredible discoveries, one of the most renowned in Computer Science; his most influential work is his study on encryption in search engines.

At this very moment, he is trying to break the encryption code to uncover the enemy's search history during the Second World Hackathon and has encountered a new challenge.

Turing wants to find two unknown values A and B , which are crucial for decoding the enemy's searches. However, the only known values are X and Y , which represent the following:

- $X = A + B$
- $Y = |A - B|$

To find these values, he asked for your help. Given X and Y , print two values A and B that satisfy the conditions above. Turing guarantees that A and B exist and are uniquely determined for the inputs he will provide to your program.

Entrada

The input contains two lines. The first line contains the number X ($0 \leq X \leq 10^9$), and the second line contains the number Y ($0 \leq Y \leq 10^9$).

Saída

Print two values A and B that satisfy the problem conditions. You may print them in any order.

Exemplos

standard input	standard output
10 2	4 6
3 7	-2 5
10 10	0 10

Problema F: Frodo & Sam

Arquivo: F. [c|cpp|java|kt|py]

Limite de tempo: 1 second

Frodo and Sam, the **great** hobbits of Middle-earth, are on an adventure to destroy the One Ring. Their journey takes them to the treacherous Mines of Moria, where a Balrog — one of Morgoth’s most feared creatures — dwells.



As they descend into the depths of Moria, something extraordinary happens — **this is where the plot twists!** The Balrog awakens and casts a powerful spell on Frodo and Sam, transporting them to another plane of reality.

Frodo and Sam are taken to the top of a mountain, where there are two paths made of magical rocks. Frodo stands to the left of these paths, and Sam to the right. The paths can be represented by two permutations P and Q of length N , where the type of the i -th rock from left to right is P_i on the first path and Q_i on the second.

To escape this illusory world and continue their journey to Mordor, Frodo and Sam must destroy some of these rocks so that the two permutations P and Q of rock types become identical, but they **cannot destroy all the rocks on both paths**.

Frodo and Sam begin destroying the rocks simultaneously. For each of the two paths, Frodo may destroy as many rocks as he wants starting from the left side of both paths, and Sam can do the same starting from the right. More formally, Frodo will choose two integers (a, b) and destroy the rocks $P_1, P_2, \dots, P_a$ and $Q_1, Q_2, \dots, Q_b$, while Sam will choose two integers (c, d) and destroy the rocks $P_N, P_{N-1}, \dots, P_c$ and $Q_N, Q_{N-1}, \dots, Q_d$. Note that Frodo and/or Sam may choose not to destroy any rocks.

Given that Frodo takes F time to destroy a rock, Sam takes S time to destroy a rock, and the time to move between any two rocks is zero, what is the minimum time they need to spend destroying rocks to escape the Balrog’s magical realm?

Entrada

The first line of input consists of three integers N ($1 \leq N \leq 10^5$), F , and S ($1 \leq F, S \leq 10^9$) — the number of rocks separating Frodo and Sam, the time Frodo takes to destroy a rock, and the time Sam takes to destroy a rock, respectively.

The second line contains a permutation of N integers $P_1, P_2, \dots, P_N$ ($1 \leq P_i \leq N$) where P_i represents the type of the i -th rock on the first path from left to right.

The third line contains a permutation of N integers $Q_1, Q_2, \dots, Q_N$ ($1 \leq Q_i \leq N$) where Q_i represents the type of the i -th rock on the second path from left to right.

Saída

The output should contain a single integer — the answer to the problem.

Exemplos

standard input	standard output
3 1 2 1 2 3 3 1 2	2
1 1 2 1 1	0
8 24 16 7 4 8 1 2 3 6 5 3 2 7 4 5 8 1 6	144
7 9 6 7 5 1 6 2 3 4 3 2 5 1 6 4 7	30

Notas

In the third case, Frodo and Sam can remove rocks so that $P = [2]$ and $Q = [2]$. Thus, Frodo must destroy 5 rocks in total, taking 120 units of time, and Sam must destroy 9 rocks in total, taking 144 units of time.

In the fourth case, Frodo and Sam can remove rocks so that $P = [5, 1, 6]$ and $Q = [5, 1, 6]$. Thus, Frodo must destroy 3 rocks in total, taking 27 units of time, and Sam must destroy 5 rocks in total, taking 30 units of time.

Problema G: Godfather

Arquivo: G. [c|cpp|java|kt|py]

Limite de tempo: 2 seconds

Joãozão and Eric, two renowned screenwriters, are working together on the movie *Godfather*. When they reached the stage of creating the climax of the story, they realized they needed a truly insane scene to surprise the audience.

Joãozão wants to make the scene as crazy as possible, trying to **maximize** the madness level of the story. On the other hand, Eric wants to maintain balance and prevent exaggeration, trying to **minimize** this level. Thus, the film's director — Bras — decided to use their differing ideas to reach a point of creative balance.

The creative process takes place over N **days**. Bras uses a binary string S to assist him. On the i -th day, Bras chooses who between Joãozão and Eric will make the change to the plot according to this string:

- If $S_i = 1$, Joãozão makes the change.
- If $S_i = 0$, Eric makes the change.

To start the process, Bras chose two plot twists: a **main plot** with madness level A , and an **auxiliary plot** with madness level B .

Whoever is chosen to make the change on day i will begin the day with two values X and Y , the current madness levels of the main and auxiliary plots, respectively. Then, this person creates a new main plot with value $X + Y$, and selects one of X or Y to continue as the auxiliary plot's value. Note that there is always exactly one main plot and one auxiliary plot.

Bras is still unsure how to choose who changes the plot on each day, so he wants to simulate possible scenarios. Given the initial S , he will perform Q operations, each of which inverts the choices on the days in an interval $[L, R]$. That is, for all i such that $L \leq i \leq R$, the value of S_i is flipped: if it was 0, it becomes 1; if it was 1, it becomes 0.

Your task is to perform this simulation; after each of the Q changes, determine what the final madness level of the main plot will be, knowing that Joãozão wants to maximize the final madness and Eric wants to minimize it. Since the madness level can be very large, Bras will be satisfied knowing just the result modulo 998244353.

Entrada

The first line of the input contains three integers N ($1 \leq N \leq 2 \cdot 10^5$), A , and B ($1 \leq A, B < 10^6$) — the number of days to develop the plot, and the initial madness levels of the main and auxiliary plots, respectively.

The second line of the input contains the binary string S of length N , which Bras uses to determine who will make the change on each day.

The third line contains an integer Q ($1 \leq Q \leq 2 \cdot 10^5$), the number of changes Bras will request.

The next Q lines each contain two integers L and R ($1 \leq L \leq R \leq N$), meaning that for every i such that $L \leq i \leq R$, if Joãozão was previously making the change on day i , now Eric will do it, and vice versa.

Saída

The output must contain Q lines; the i -th line must contain the madness level of the main plot twist after the first i updates, modulo 998244353.

Exemplos

standard input	standard output
8 4 5 11101011 8 1 8 1 6 4 7 4 6 1 8 2 6 8 8 5 7	131 162 226 213 131 153 153 170
1 8 2 1 5 1 1 1 1 1 1 1 1 1 1	10 10 10 10 10
6 3 19 111111 3 1 5 2 5 4 5	37 191 122

Problema H: Cinematic Hierarchy

Arquivo: H. [c|cpp|java|kt|py]
 Limite de tempo: 2 seconds

BRUTE, a group originally known for their performance in competitive programming contests, decided to change direction and venture into the world of cinema. Risa, a renowned director from the group, was tasked with selecting some actors from N candidates for the first film produced by BRUTE.

To do this, Risa contacted all the actors and collected their conditions for participating in the project. Each actor, in addition to demanding a monetary value, indicated names of other **more famous** actors they would like to work with, requiring that these also be included in the cast. Each actor is identified by a number, where the i -th least famous actor is represented by the number i , meaning that the most famous actor is represented by the number N , the next most famous by $N - 1$, and so on.

After organizing all the requirements, Risa noticed two things:

- Each actor appears in **at most one** other actor's requirement.
- There is **exactly one** actor who is not required by any other, and this actor is the least famous among all.

Knowing that the i -th actor generates a profit of a_i if hired (negative values represent a loss), what is the maximum possible profit that BRUTE can obtain if Risa optimally chooses which actors to hire? Note that Risa may choose to hire no actors at all.

Entrada

The first line of the input contains an integer N ($2 \leq N \leq 5 \cdot 10^5$), representing the number of candidates Risa has available.

The second line contains N integers $a_1, a_2, \dots, a_N$ ($-10^9 \leq a_i \leq 10^9$), where a_i represents the profit the i -th actor will generate if hired for the film.

The third and last line contains $N - 1$ integers $s_2, s_3, \dots, s_N$ ($1 \leq s_i < i$), indicating that actor s_i required the participation of actor i in the cast.

Saída

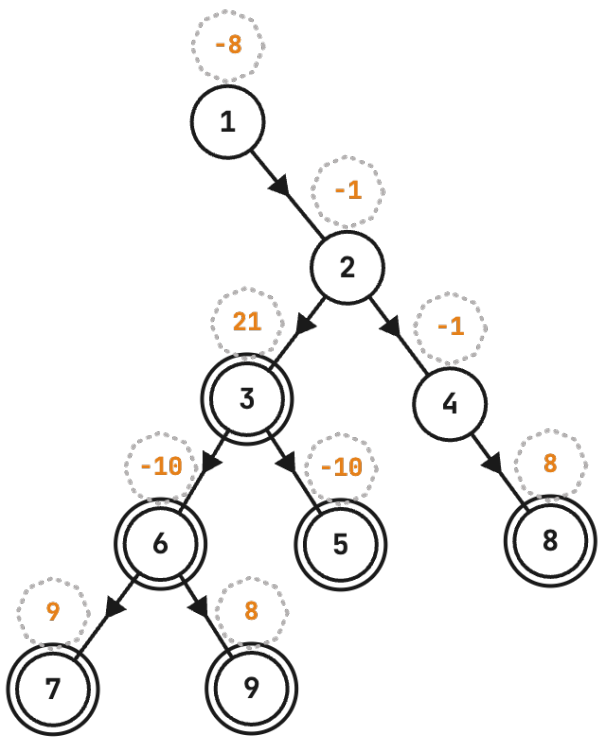
The output must contain exactly one line, the maximum profit BRUTE can obtain if Risa optimally chooses which actors to hire.

Exemplos

standard input	standard output
9 -8 -1 21 -1 -10 -10 9 8 8 1 2 2 3 3 6 4 6	26
4 -10 -8 -9 -10 1 1 2	0
7 1 -5 1 9 2 0 -7 1 2 1 4 4 2	12

Notas

In the first case, Risa can hire actors 3, 5, 6, 7, 8, and 9, to obtain a profit of $21 - 10 - 10 + 9 + 8 + 8 = 26$.



Above is an illustration of the first case; the numbers inside the nodes represent the actors, while the dotted circle above them shows their generated profit a_i . Each directed edge from i to j means actor i required actor j . The nodes highlighted with a surrounding circle represent a possible optimal set of hired actors.

Problema I: Lost Language

Arquivo: I. [c|cpp|java|kt|py]
 Limite de tempo: 1 second

Rondônia Jones is an archaeologist and explorer known for his extensive knowledge of ancient civilizations and their languages.

During one of his expeditions, he discovered a stone from the Brutense people (dated 998244353 B.C.) containing a short introductory text followed by N mysterious inscriptions: all of them were words made up of Latin alphabet letters (from “a” to “z”). The text explained, in a long-forgotten language, that all these words were non-decreasing. This led Jones to expect that they would look like “abcdef” or “aacgmz” — non-decreasing words, assuming the standard Latin alphabetical order. However, contrary to expectations, the letters appeared to follow a strange order unfamiliar to him.

Now, Jones wants to know whether there exists some ordering of the Latin alphabet letters that makes the inscribed words obey the rule stated in the text. If so, he asks you to print one such possible order of the alphabet, as it would be crucial for his studies of the Brutense people.

Entrada

The first line of input contains an integer N ($1 \leq N \leq 10^5$).

The next N lines follow. The i -th of these contains a word s_i .

It is guaranteed that the sum of all $|s_i|$ is less than or equal to 10^6 .

Saída

Print one line with “SIM” (YES) or “NAO” (NO), depending on whether a valid answer exists.

If the answer is YES, print an additional line with a permutation of the 26 Latin alphabet letters that satisfies the requirements. If multiple answers are possible, print any one.

Exemplos

standard input	standard output
3 abc def ghi	SIM abcdefghijklmnopqrstuvwxyz
3 acb zcb ghi	SIM abcdefghijklmnopqrstuvwxyzcb
4 brute bruta bruto trem	NAO

Problema J: John Wick

Arquivo: J. [c|cpp|java|kt|py]
 Limite de tempo: 1 second

John Wick Fröhlich has finally managed to escape the mysterious labyrinth, but as he moved away, he realized he had dropped his navigation compass — a vital item — exactly in the central corridor, a narrow passage that is slowly closing. Now, he needs to be away for **exactly** K seconds to retrieve the artifact and return before the passage becomes impassable.

The corridor is bounded by two walls, represented by the character “#”. Between them, there are indestructible stones (character “o”) and empty spaces (character “.”), which were already part of the labyrinth.

Each second, the **left** wall advances one position to the **right**, immediately pushing any stone (o) that is in the space ahead.

Pushed stones can, in turn, push other stones in a chain reaction, but they are **never** destroyed or overlapped. The walls are also indestructible and cannot be overlapped.

Before setting out, John Fröhlich wants to know what the configuration of the corridor will look like **after exactly** K seconds of wall movement, to ensure he can return safely.

Entrada

The first line contains two integers N and K ($3 \leq N \leq 500$, $0 \leq K \leq 500$) — the initial length of the corridor and the amount of time in seconds John Fröhlich will be away.

The second line contains a string of length N , composed of:

- Exactly two “#” characters at the ends (the moving walls),
- A combination of “o” (stones) and “.” (empty spaces) in between.

Saída

Print a single line with the string of length N representing the state of the corridor **after** K seconds of movement.

Exemplos

standard input	standard output
8 3 #o..o.o#	...#ooo#
6 3 #....#	...#.#
6 2 #oooo#	#oooo#
13 4 #...o..o.ooo#	...#o.o.ooo#

Problema K: Kill Bill Vol. 1

Arquivo: K. [c|cpp|java|kt|py]

Limite de tempo: 1 second

When the Bride asked Hanzo, the greatest and finest katana forger the world has ever seen, to craft a new weapon, the master was hesitant, as it had been decades since his last creation and the process was rather complicated.



Nevertheless, sympathizing with the Bride's story and mission, Hanzo decided to call you — the smartest person he knows — and explain step-by-step his forging method:

- Hanzo initially builds the katana by attaching a Japanese steel blade to a hilt and sharpens the blade in the conventional way;
- After that, the master divides the blade into N parts and assigns a value to each part based on its sharpness level;
- Now, Hanzo aims to minimize the number of distinct sharpness values on the blade by choosing two distinct parts of the blade that have the same sharpness level and modifying all parts between them to also have that same level.

More formally, let a_i be the sharpness value of the i -th part of Hanzo's blade. The operation is defined as follows:

1. Choose two indices i and j ($1 \leq i < j \leq N$) such that $a_i = a_j$;
2. For every k ($i < k \leq j$), assign $a_k := a_i$.

Hanzo stated that the sharpness of the blade is the most important part of forging a great katana. However, it is also by far the most difficult step, and he feels he's not always making the most of the blade. Therefore, help him minimize the number of distinct sharpness values by developing a program that receives the initial sharpness levels of the blade and performs Hanzo's operation any number of times.

Entrada

The input consists of two lines. The first line contains a single integer N ($1 \leq N \leq 2 \cdot 10^5$), the number of parts into which Hanzo divided the blade. The second line contains N numbers ($1 \leq a_i \leq 10^9$), the sharpness values of each part of the blade.

Saída

Print a single integer, the minimum number of distinct sharpness values on the blade after applying the process described by Master Hanzo.

Exemplos

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

Problema L: Las Tortuguitas

Arquivo: L. [c|cpp|java|kt|py]
 Limite de tempo: 1 second

Dudu is on a magical journey inside the Fantastic Chocolate Factory and discovered a secret room full of machines that produce extremely rare candies called “Tortuguitas”. This room contains N machines numbered from 1 to N , and each machine produces exactly K Tortuguitas, numbered from 1 to K .

A machine is represented by the rarity of the Tortuguitas it produces. More formally, the i -th machine produces K Tortuguitas with rarities $r_{i,1}, r_{i,2}, \dots, r_{i,K}$. Let X be the rarity value of the simplest Tortugueta, and the rarity of a Tortugueta can be calculated as follows.

- $r_{i,j} = j \cdot X$ for all $(1 \leq j \leq K)$, when $i = 1$.
- $r_{i,j} = \sum_{t=1}^j r_{i-1,t}$, when $i > 1$.

Dudu was amazed by these magical machines and asked for your help to answer some questions about them.

He will give you Q questions of the following type: In the i -th machine, what is the sum of the rarities of the Tortuguitas numbered greater than or equal to L and less than or equal to R ?

More formally, Dudu will ask for the result of $\sum_{j=L}^R r_{i,j}$.

Since this result can be absurdly large, he only wants you to return the remainder of this sum when divided by 998244353.

Entrada

The first line of input contains 4 integers N , K , Q , and X ($1 \leq N, K, Q \leq 3 \cdot 10^5$) ($1 \leq X < 998244353$) — the number of machines, the number of Tortuguitas each machine produces, the number of questions Dudu asked, and the rarity of the simplest Tortugueta, respectively.

The next Q lines each contain 3 integers i , L , and R ($1 \leq i \leq N$) ($1 \leq L \leq R \leq K$), the machine number and the range for which Dudu wants to know the sum.

Saída

The output should contain Q lines, each with the result of the sum modulo 998244353.

Exemplos

standard input	standard output
3 10 3 5 1 4 7 2 4 7 3 4 7	110 370 975
50 100 2 7 4 1 10 50 3 100	14014 574802200
1 1 1 1 1 1 1	1

Problema M: Movieguessr

Arquivo: M. [c|cpp|java|kt|py]
 Limite de tempo: 1 second

The year is 62442 and Machado and Marcel were playing their favorite game, **Movieguessr**. Movieguessr is a famous game that teleports its players into the universe of a movie, where they have a limited time to figure out which work they are in.

When Machado and Marcel thought it was just another round of the game, the unbelievable happened — the Movieguessr servers crashed, and they became trapped inside one of the most iconic schools from ancient movies, **Hogwarts**.

The two were on the famous magical staircase of Hogwarts, but in different positions on it, more specifically:

- The staircase is represented by N steps numbered from 1 to N and a list a to represent them. The first step is at ground level ($a_1 = 0$), and the rest can be above ($a_i = 1$) or below ($a_i = -1$) the previous step: going up a step increases a person's height by **one** unit; going down decreases it by **one** unit.
- Marcel is on the K -th step.
- Machado found a magic wand that allows him to teleport to any empty step on the staircase — that is, any step except the one Marcel is occupying.

Machado wants to use this magic to find his friend. Knowing that Machado can only see Marcel on the staircase **if and only if** his height is **greater than or equal to** Marcel's, and considering that Machado is A units tall and Marcel is B , your mission is to help Machado: is there a step where Machado will be able to see Marcel? If so, to which step should he teleport in order to finally locate Marcel?

Entrada

The first line of the input contains 3 integers N , A , and B ($2 \leq N \leq 2 \cdot 10^5$) ($1 \leq A, B \leq 10^9$), the number of steps in the staircase, Machado's height, and Marcel's height, respectively.

The second line of the input contains N integers $a_1, a_2, \dots, a_N$, where a_i is 1 if the i -th step is above the previous one and -1 if it is below; also, $a_1 = 0$.

The third line of the input contains 1 integer K ($1 \leq K \leq N$), the step Marcel is occupying.

Saída

The first line of the output should contain "SIM" (without quotes) if there is a step where Machado can find Marcel, or "NAO" (without quotes) otherwise. If the answer is "SIM", the second line of the output should contain a number, the position Machado should teleport to in order to see Marcel; if more than one such step exists, any of them may be printed.

Exemplos

standard input	standard output
5 10 8 0 1 -1 -1 1 2	SIM 1
11 2 4 0 1 1 -1 -1 -1 1 1 -1 1 -1 10	NAO
8 19 16 0 -1 1 1 1 1 1 1 8	SIM 5

Problema N: Ni

Arquivo: N. [c|cpp|java|kt|py]

Limite de tempo: 1 second

On a regular day walking through the forest, Luiza encounters a curious group of people: the Knights who say “Ni”.

Luiza was stunned by how many times they said the word “Ni”, but she noticed that they said “Ni” exactly K times.

Write a program that prints exactly what they said.

Entrada

The only line of input contains an integer K ($1 \leq K \leq 5$), the number of times “Ni” was said.

Saída

Print the solution to the problem. Each occurrence of “Ni” must be separated by a whitespace, and you may use any capitalization.

Exemplos

standard input	standard output
5	Ni Ni Ni Ni Ni
3	Ni Ni Ni

Notas

Problema O: The Martian

Arquivo: 0. [c|cpp|java|kt|py]

Limite de tempo: 1 second

Upon arriving on Mars, astronaut Mark Watney and his team discovered that the planet was not quite as they had expected. Its terrain was filled with large ravines—gaps in the ground—that made it very difficult for the team to traverse the red planet.

Fortunately, the astronauts brought along a set of ladders of various lengths; these ladders allowed them to cross the ravines when laid out horizontally, as long as their extended length was greater than or equal to the size of the gap.

The team has a schedule indicating one coordinate of interest to visit each day, and for these visits, they choose one ladder from their set to help cross the ravines. After reaching the point of interest, the astronauts must return via the same path they came from (back to the area where they reside), and the used ladder is then discarded due to wear and tear, making it unavailable for future days.

Given this, Mark was tasked with developing a program that, when provided with a map showing regions separated by ravines, the lengths of the ravines, the set of ladders brought by the team, and the cities the team wants to visit each day, maximizes the number of regions of interest visited during the entire mission. Mark, being a great biologist but not much of a programmer, called you for help.

The map of Mars is given as a graph with N regions numbered from 1 to N and M ravines numbered from 1 to M connecting these regions, where the i -th ravine has a length of w_i meters. It is also a fact that if it were possible to use a ladder of infinite length, one could travel from any region to any other region using only the ravines.

Additionally, the mission will last D days, where the region to be visited on the i -th day is d_i , and the team has exactly D ladders numbered from 1 to D , where the length of the i -th ladder is e_i . Note that a ladder can be used on any day, but only once; and during a day, the chosen ladder can be reused as many times as necessary.

Knowing that the team resides in region 1, write a program to help Mark's team with their mission!

Entrada

The first line of the input consists of two numbers N and M ($2 \leq N \leq 2 \cdot 10^5$) ($1 \leq M \leq 2 \cdot 10^5$), the number of regions and ravines on Mars, respectively.

The next M lines each contain three integers u_i , v_i , and w_i , ($1 \leq u_i, v_i \leq N$) ($1 \leq w_i \leq 10^9$), indicating the existence of a ravine of length w_i between regions u_i and v_i . It is guaranteed that there is only one ravine connecting any two regions, and no ravine connects a region to itself.

Then, a line with a single integer D ($1 \leq D \leq N$) is given, representing the number of days in the mission; followed by two lines with D integers each.

The first of these lines contains $d_1, d_2, \dots, d_D$ ($1 \leq d_i \leq N$), indicating that the i -th region to be visited is region number d_i .

The next line contains $e_1, e_2, \dots, e_D$ ($1 \leq e_i \leq 10^9$), indicating that the i -th ladder is e_i meters long.

Saída

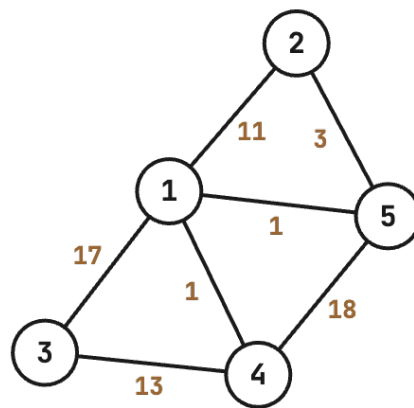
Print a single line with the maximum number of regions that Mark and his team can visit.

Exemplos

standard input	standard output
<pre> 6 10 5 3 4 3 6 3 1 6 2 4 1 7 2 1 15 6 4 19 3 2 10 5 1 7 4 2 17 5 2 5 6 1 2 3 4 5 6 3 4 3 5 5 5 </pre>	<pre> 5 </pre>
<pre> 5 7 2 5 3 5 4 18 5 1 1 4 1 1 4 3 13 3 1 17 2 1 11 3 5 3 2 1 11 1 </pre>	<pre> 2 </pre>
<pre> 5 6 2 1 13 4 2 1 5 2 1 5 1 2 3 2 11 4 1 5 2 2 3 1 1 </pre>	<pre> 0 </pre>

Notas

Below is the map for the second example:



We can use one of the ladders of length two to reach node 5 on the first day; then, we can skip the second day's target and use the ladder of length 11 to reach node 2 on the third day (through the path 1 – 5 – 2). Note that there may be more than one way to achieve the optimal answer.