

Caderno de Provas



Maratona nas Estrelas

I Seletiva Interna UDESC 2024

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**.

Maratona nas Estrelas

I Seletiva Interna UDESC 2024

18 de Maio de 2024

Conteúdo

Problema A: Stellar Year	7
Problema B: Bit Tennis 2	8
Problema C: Song	10
Problema D: Course Deviation	12
Problema E: El Café	14
Problema F: Frogs or Toads?	16
Problema G: Gargantua	18
Problema H: Higgs	19
Problema I: Itwise Bor	21
Problema J: Jugando Fuerte	23
Problema K: Kosmos	26
Problema L: Lango Mocos	27
Problema M: Giant Worms	29
Problema N: Shield Navigation	32
Problema O: Osmos V	34

Problema A: Stellar Year

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

Limite de tempo: 1 second

Guizinho has just bought a new pair of glasses and can finally see the stars! Very excited about this, he started noting down patterns he observed in the night sky.

After years of observation, Guizinho recorded in his notebook N numbers $a_1, a_2, \dots, a_N$, representing that star i appears in Earth's sky every a_i years. Gui says that this number represents the period of a star.

The year 2024 is a very lucky year for him because, while observing the sky, he noticed that all N stars whose periods he recorded are visible at the same time! Guizinho named the years in which all these stars appear in the sky simultaneously as "stellar years."

After admiring the sky long enough, he started thinking about the future of star observation: Knowing that the current year is a stellar year, which stars will be visible in Earth's sky after X more years?

Entrada

The first line of the input consists of two integers N ($1 \leq N \leq 3 \cdot 10^5$) and X ($0 \leq X \leq 10^9$), the number of stars Guizinho recorded in his notebook and how many years into the future he wants to consider, respectively.

The next line consists of N integers, $a_1, a_2, \dots, a_N$ ($1 \leq a_i \leq 10^9$), the period of each star.

Saída

The first line of the output must contain an integer K ($0 \leq K \leq N$), the number of stars that will be visible in the sky after X years.

The next line must contain K integers $i_1, i_2, \dots, i_K$, the stars that will be visible in the sky after X years. The stars can be shown in any order.

Exemplos

standard input	standard output
1 4 2	1 1
5 3 3 4 1 9 81	2 1 3
3 2 11 1 3	1 2
4 0 4 3 92 7	4 1 2 3 4

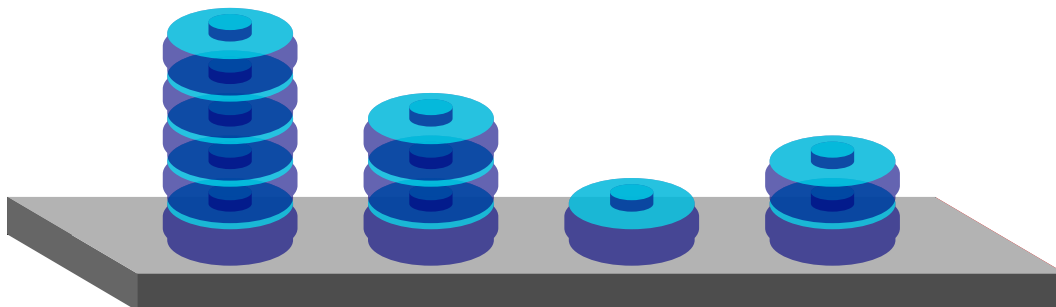
Problema B: Bit Tennis 2

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

Giovana and Julia, after becoming the champions of the Bit Tennis doubles tournament on Earth, decided to travel to another planet in search of more competition. During the trip, they remembered they had a holographic piece board that allows them to play various games. Since they were already bored of all the games (including Bit Tennis), they decided to invent a new game: Bit Tennis 2.

The rules of the game are as follows:

- The game starts with N stacks of holographic pieces, where the i -th stack contains a_i pieces;
- Julia starts the game, and after that, the turns alternate between Giovana and Julia;
- On each turn, the player must choose a stack and remove any number of pieces that is a power of 2;
- The player who cannot make a move loses the game.



For the game with stack sizes $[5, 1, 3, 2]$, Julia, by starting, has a strategy that ensures Giovana cannot win the game.

They both realized that the game was very difficult for the second player, so they added a new rule: before the game begins, Giovana must perform the following operation exactly X times:

- Choose a stack and double the number of pieces in it. That is, if Giovana chooses stack i , which has a_i pieces, it now has $2 \cdot a_i$ pieces.

Both players, just like when they invented Bit Tennis, quickly learned to play optimally and noticed that the outcome of the game seems to be determined even before the first move is made. Curious about this fact, they asked for your help. Given N , X , and the size of each stack, they ask you to determine who will be the winner.

Entrada

The first line of the input contains two numbers N ($1 \leq N \leq 10^5$) and X ($0 \leq X \leq 10^9$), the number of stacks and the number of operations Giovana must perform, respectively.

The second line contains N values a_i ($1 \leq a_i \leq 10^9$), the number of holographic pieces in each stack.

Saída

Print “Giovana” or “Julia”, the name of the winner of the game.

Exemplos

standard input	standard output
4 2 5 3 1 2	Julia
1 10 32	Julia
2 1 4 2	Giovana

Problema C: Song

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

Lívia is the captain of a spaceship that was attacked by space pirates, and, as a last resort for survival, she had to eject herself from the ship.

Now, Lívia finds herself floating aimlessly through space, and the only things she has with her are her cellphone and a pair of headphones. Luckily, she managed to send an SOS before leaving the ship and knows that, although it may take a few days, help is on the way. To pass the time, she decided to listen to some songs by her favorite artist, Saylor Twift.

Lívia created her own musical notation for the melodies she listens to. She considers that a song contains at most 26 distinct musical notes, and, when listening to a melody, she represents each musical note by one of the letters of the Latin alphabet (from “a” to “z”). Therefore, a melody is represented by a sequence of letters.

Additionally, Lívia wrote down N pairs of notes (u, v) to represent that note v sounds good after note u . Now, she wants to compile a playlist with all of Saylor Twift’s beautiful songs, so she asked for your help. Lívia considers beautiful only the melodies where, for every note played (except the first), it sounds good after the previous note.

Lívia emphasizes that a certain note v sounding good after note u does not mean that note u would also sound good after note v . Given a melody by Saylor Twift, determine whether Lívia will consider it beautiful or not.

Entrada

The first line of the input contains a string S ($1 \leq |S| \leq 10^5$), representing Saylor Twift’s song.

The second line contains an integer N ($0 \leq N \leq 676$), the number of pairs of musical notes that sound good together.

Each of the next N lines contains two Latin alphabet letters, u and v , representing that note u sounds good before note v .

Saída

The output must contain “SIM” if Lívia will consider the melody beautiful, or “NAO” if she will not.

Exemplos

standard input	standard output
abac 3 a b b a a c	SIM
abac 3 a b c a b a	NAO
taylor swift 10 l o a y y l i f o r r s s w w i t a f t	SIM
cde 2 c d c e	NAO

Problema D: Course Deviation

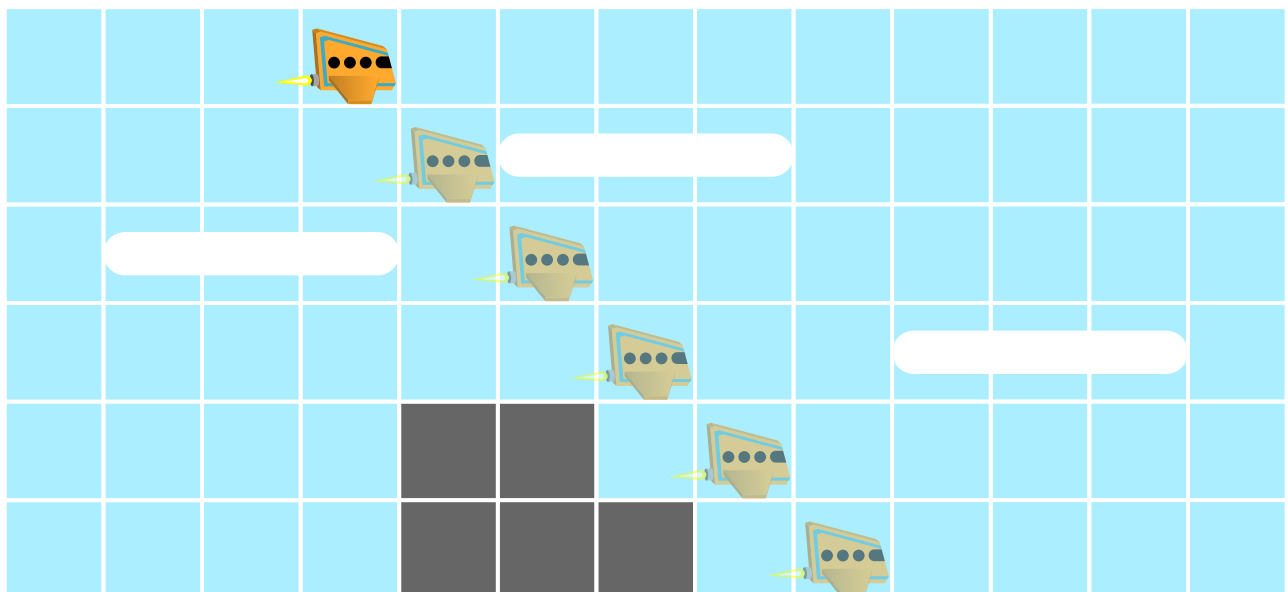
Arquivo: D. [c|cpp|java|kt|py]

Limite de tempo: 1 second

Commander Rosso, renowned for his precise and safe landings, finds himself in a critical situation in the middle of the void of space during yet another mission. An unexpected solar storm interrupts his exploration mission, damaging the navigation systems of his ship, the Geometric Voyager, and forcing an emergency landing on the remote planet JOI-47.

The terrain of JOI-47 is notoriously uneven: there is a chain of kilometer-high mountains and only a small strip of land where the ship can land, located beyond the mountains.

Upon entering the planet's dense atmosphere, Rosso must carefully adjust the initial descent altitude of the Geometric Voyager, as the ship's automated altitude control system causes it to descend at a constant rate of one kilometer per second. Each second of descent corresponds to one kilometer of forward movement toward the landing strip. Since the available landing strip is short, Rosso wants to choose the **smallest possible initial altitude that avoids all the mountains**. The ship avoids a mountain if its altitude is strictly greater than the mountain's height when passing over it.



In the example above, the smallest initial altitude required to pass over the mountains is 5; if the initial altitude were lower, the ship would collide with the first or second mountain. The mountains have heights $[2, 2, 1]$ and the ship will have altitudes $[4, 3, 2]$ while passing over the mountains. Note that the height of the Geometric Voyager is measured by the number of squares below it, and when the ship is landed, its height is zero.

You are Rosso's co-pilot and he wants to test whether you are fit to pilot the Geometric Voyager alone. Mistakes are not an option, as a poorly calculated initial altitude could cause the ship to crash into the mountains or fail to land in time. Given the heights of the N mountains, report the ideal height to enter the landing course.

Entrada

The first line of input consists of an integer N ($1 \leq N \leq 10^5$), the number of mountains

between Rosso's ship and the landing strip.

The second line contains N integers $h_1, h_2, \dots, h_N$ ($1 \leq h_i \leq 10^9$), where h_i represents the height of the i -th mountain.

Saída

The output must contain a single integer, representing the smallest initial altitude (in kilometers) required to enter the landing course without colliding with any mountain.

Exemplos

standard input	standard output
3 2 2 1	5
1 10	12
5 3 4 2 3 1	8

Problema E: El Café

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

Limite de tempo: 1 second

Max and his older brother Min had an unbreakable bond throughout their lives. However, a few years ago, Min moved to the Sombrero Galaxy, where he opened a coffee shop, El Café, and became very successful.

Following his family's passion for coffee, Max applied for a job at his brother's coffee shop. The first stage of the selection process is a simulation of a normal working day, which would be straightforward if not for the unconventional way the café serves its customers.

At El Café, there is only one type of drink, the *vainilla helada*, which consists of a combination of ALL (yes, all) the available ingredients in the store at that moment. Furthermore, each ingredient must be equally distributed among the ordered drinks, based on the quantity of each. Thus, when a customer places an order at El Café, they only specify the number of *vainillas heladas* they want, and, if possible, the employee must prepare the order. If it is not possible, the customer should be informed that the requested quantity of drinks cannot be made at that time.

For example, if the current list of ingredient quantities is $[24, 12, 3]$ and a customer orders 3 *vainillas heladas*, Max should inform them that he can fulfill the order, as he can prepare three drinks, each using ingredient quantities $[8, 4, 1]$. On the other hand, if the customer had requested 2 drinks, Max would have had to inform them that it is not possible at the moment.

During the shift, new ingredients may arrive, and some of the newest ingredients may expire, forcing Max to throw them away (expiration works differently in the Sombrero Galaxy).

To increase his chances of getting the job, Max decides to call you (since he knows your programming skills) and asks you to write a program to help him train for the selection process.

The program must handle 3 types of requests:

- Type 1: indicates that a new ingredient has arrived in quantity G . Note that each arriving ingredient is different from the others.
- Type 2: indicates that the last K ingredients, i.e., the K newest ingredients, have expired and must be discarded.
- Type 3: simulates a possible order of X *vainillas heladas* from a customer. The program should only inform whether it is possible to fulfill such an order.

Given the number N of different ingredients at the beginning of the shift, the quantity a_i of each ingredient, and the Q requests, write a program to help Max pass the test.

Entrada

The first line consists of two integers N and Q ($1 \leq N, Q \leq 10^5$), representing the number of ingredients at the start of the shift and the number of requests, respectively.

The second line contains N integers $a_1, a_2, \dots, a_N$ ($1 \leq a_i \leq 10^9$), where a_i represents the quantity of the i -th ingredient in order from oldest to newest.

Then, Q lines follow, each representing a request.

- 1 G : a type 1 request indicating that a new ingredient has arrived in quantity G ($1 \leq G \leq 10^9$).
- 2 K : a type 2 request indicating that the last K ($1 \leq K \leq 10^5$) newest ingredients have expired and should be discarded. It is guaranteed that there will be at least $K + 1$ different ingredients for this type of request.
- 3 X : a type 3 request asking whether it is possible to serve X ($1 \leq X \leq 10^9$) *vainillas heladas* at that moment.

Saída

For each request of type 3, print a single line. If it is possible to fulfill the customer's order, print "SIM", otherwise print "NAO".

Exemplos

standard input	standard output
5 5 12 12 6 4 2 2 3 3 4 1 6 1 3 3 2	SIM NAO
3 3 20 20 4 3 5 1 10 3 2	NAO SIM

Problema F: Frogs or Toads?

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

Eric has just bought a new game for his NES, called Battletoads. The objective of the game is to defeat the Dark Queen and her army of space mutants.

After spending a long time trying to beat the game, he realized that it was very difficult to kill the Dark Queen because his character had no weapons and needed to defeat her using only punches.

Frustrated by not being able to defeat the game's final boss, Eric decided to develop a new game, which he named Battlefrogs. In this game, the final boss is called Light Queen and the playable characters are always equipped with a laser weapon. The weapon works as follows:

Let E be the weapon's energy.

- If $E > 0$, Eric can fire a laser beam and E decreases by one unit. If Eric fires the beam at a space mutant, the mutant will die, and if he fires the beam at the Light Queen, her life decreases by one unit.
- If $E \leq 0$, Eric can still fire a laser beam and E decreases by one unit. However, if Eric fires the beam at a space mutant, the mutant will die, but if he fires the beam at the Light Queen, her life **will not be affected**.

Before reaching the Light Queen's level, Eric must pass through N levels numbered from 1 to N , starting from level 1. Each level contains s_i space mutants on the way and an energy bank with power e_i that charges his weapon by e_i units. Additionally, at the start of each level, there is an alternative route with a wormhole guarded by X space mutants that allows Eric to skip k levels ahead.

If Eric uses the wormhole at level i , he immediately jumps to level $i + k$, or, if the $(i + k)$ -th level does not exist, to the Light Queen's level. When using the wormhole, Eric does not need to kill any of the s_i mutants, but he will not be able to charge his weapon by e_i units and is required to kill the X mutants guarding the wormhole. If he decides not to use the wormhole at level i , he must defeat the s_i space mutants, will be able to increase his weapon's energy by e_i units, and will proceed to level $(i + 1)$, or to the Light Queen's level if the $(i + 1)$ -th level does not exist.

Knowing that the Light Queen dies when her life reaches 0 and that Eric's weapon starts with 0 energy, he is now thinking about balancing the game and needs to decide how much life to assign to the final boss. He asked you: what is the maximum amount of life the Light Queen can have so that it is still possible to defeat her?

Entrada

The first line of the input consists of three integers N ($1 \leq N \leq 2 \cdot 10^5$), the number of levels in Battlefrogs, k ($1 \leq k \leq N$), how many levels ahead each wormhole takes Eric, and X ($0 \leq X \leq 10^9$), the number of space mutants guarding each wormhole.

Then, there are N lines, where the i -th line contains two integers s_i and e_i , representing the number of space mutants and the power of the energy bank at level i , respectively.

Saída

The output must contain a single integer, the maximum life the Light Queen can have such that it is possible to defeat her.

Exemplos

standard input	standard output
4 2 1 1 2 4 6 5 2 3 11	8
5 3 20 3 4 2 2 5 6 3 1 8 5	0

Notas

If Eric's weapon cannot reach the Light Queen with energy greater than zero, the Light Queen's life must be zero.

Problema G: Gargantua

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

Léo and Ema are two astronauts living on Earth. One day, Ema was called on a mission to planet Miller, which orbits a black hole named Gargantua.

Because it orbits a black hole, time on Miller passes X times slower than on Earth.

Knowing that Léo was A years old before Ema traveled, and that Ema stayed Y years (according to Miller's time) on Miller, what will be Léo's age when Ema returns to Earth?

Entrada

The input consists of three lines, each containing one of the integers A, X, Y ($0 \leq A, X, Y \leq 100$) — Léo's age before Ema traveled to Miller, the factor by which time passes slower on Miller compared to Earth, and the number of years Ema stayed on Miller, respectively.

Saída

The output must contain a single integer, Léo's age when Ema returns to Earth.

Exemplos

standard input	standard output
20 3 4	32
31 17 3	82
1 2 3	7

Problema H: Higgs

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

Risa is the greatest thief in the GianParme Galaxy, and she is preparing for her biggest heist yet. The target is on planet Amorim, named after its fearless leader Matheus Amorim, who ruled the planet for many years until his tragic passing.

Risa's goal is to steal the planet's greatest relic, the crown that Matheus wore during his reign. The artifact is stored in a museum on planet Amorim. The first step of her plan is figuring out how to reach this planet.

The GianParme Galaxy can be modeled by N planets numbered from 1 to N and M connections between them. Each connection is represented by two planets (u, v) and a "Higgs level" h . The connections are designed such that it's possible to travel from any planet to any other planet using the galaxy's connections.

The Higgs level is a form of security in the galaxy; all residents of GianParme have their own Higgs level, and to use a connection between two planets, their Higgs level must be **greater than or equal to** the Higgs level of that connection.

Risa is currently on planet number 1, and planet Amorim is planet number N . Since Risa is a well-known thief, her Higgs level is 0, and to complete her journey to planet Amorim, she has two options:

- Bribe the guard of a connection to use it regardless of her Higgs level.
- Forge her identity to have any Higgs level she desires.

However, forging an identity is a very difficult task, so she needs to know: What is the minimal Higgs level she must forge in order to complete her journey to planet Amorim, knowing she has enough money to bribe K guards?

Entrada

The first line of the input consists of three integers N ($2 \leq N \leq 3 \cdot 10^5$), M ($1 \leq M \leq 3 \cdot 10^5$), and K ($0 \leq K \leq M$), representing the number of planets, the number of connections, and the number of guards Risa can bribe, respectively.

The next M lines each contain three integers u, v ($1 \leq u, v \leq N$) and h ($1 \leq h \leq 10^9$), indicating that there is a connection between planets u and v with Higgs level h .

Saída

The output must contain a single integer, the minimal Higgs level Risa needs to forge to complete her journey.

Exemplos

standard input	standard output
4 4 1 1 2 3 2 4 4 1 3 3 3 4 1	1
6 6 2 1 2 1 2 3 3 3 4 10 4 6 4 3 5 2 5 6 4	2
5 6 2 1 2 3 1 3 2 2 3 1 3 4 5 4 5 3 2 4 4	2

Problema I: Itwise Bor

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

Limite de tempo: 1 second

João is visiting a planet far from Earth, the alien planet PEI-X40, governed by Nei, a peaceful leader who warmly welcomed João's arrival.

The inhabitants of this planet use binary representation to express numbers: for example, the number 6 on Earth is represented as "110" there. In this representation, they refer to each digit as an "it", upon which they define operations called "itwise". Among these, the main ones are "band" and "bor".

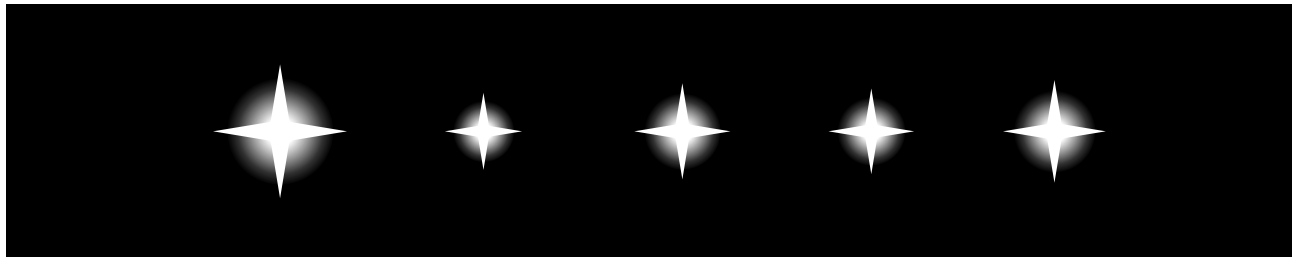
The bor operation is particularly interesting to an astronomer like João because Nei has decreed that the beauty of a constellation is defined as the itwise bor of the brightness values of all the stars within it.

Curious, João studied a bit about bor and learned that it is represented by the symbol "|" and works as follows:

For each position i in the binary representation of X and Y , if X has the i -th it equal to 1 and/or Y has the i -th it equal to 1, then $X | Y$ has the i -th it equal to 1. Otherwise, $X | Y$ has the i -th it equal to 0.

For example, in binary, $101 | 010 = 111$. This is equivalent to $5 | 2 = 7$ in the Earth decimal system.

While gazing at the sky of PEI-X40, João noticed a peculiar constellation: there were N stars in the sky, one next to the other. He then recorded the brightness of each star $b_1, b_2, \dots, b_N$, in the order they appeared.



After jotting down some mathematical expressions, he realized that this constellation can be divided into several continuous constellations in such a way that the structure of the stars being next to each other is preserved.

Now, João wants to divide the constellation into several others so that the sum of the beauties of all these constellations is maximized, and, among all the ways that yield the maximum sum, he wants the division with the smallest number of constellations. Since the inhabitants of PEI-X40 didn't train him well enough for this, he asked for your help.

Entrada

The input consists of two lines.

The first line contains an integer N ($1 \leq N \leq 3 \cdot 10^5$), the number of stars in the original constellation.

The next line contains N integers $b_1, b_2, \dots, b_N$ ($0 \leq b_i < 2^{30}$), the brightness values of the stars in the order they appear in the sky.

Saída

The output must contain two integers S and K , the maximum possible sum of beauty and the minimal number of groups required to achieve this sum, respectively.

Exemplos

standard input	standard output
5 4 1 2 1 3	11 3
3 1 2 3	6 2
4 7 7 7 7	28 4
5 1 3 4 8 2	18 3

Notas

In the first example, João can divide the constellation into three groups: $[4, 1]$, $[2, 1]$, and $[3]$ to obtain the maximum sum of $5 + 3 + 3 = 11$.

In the second example, João can divide the constellation into two groups: $[1, 2]$ and $[3]$ to obtain the maximum sum of $3 + 3 = 6$.

Problema J: Jugando Fuerte

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

The billionaire explorer Granza, founder of the major space travel company Space-Y, enjoys exploring the Universe aboard his ship, the StarPuma-GTE.

During one of his trips through the Sombrero Galaxy, Granza discovered a planet called Guadalajara. This planet was famous not only for its breathtaking landscapes but also for its enormous casinos and a peculiar card game known as At-Poker.

At-Poker was considered the national pastime of Guadalajara, involving skill, luck, and strategy. On this planet, the casinos are luminous temples where fortunes are made and lost in a matter of rounds. Granza, always a man of challenges, couldn't resist the temptation to participate. As they say in Guadalajara, Granza "jugó fuerte" — he bet high — and unfortunately, his Earth poker skills were of little use against the complex strategies of At-Poker, and he soon found himself without a large part of his fortune.

In At-Poker, N players numbered from 1 to N are arranged around a circular table in the order 1, 2, 3... N , so that for every player i from 1 to $N - 1$, they are to the left of player $i + 1$, and player N is to the left of player 1.

In the game of At-Poker, each player receives a deck of cards represented by a string of lowercase letters and a number G_i . The official rulebook specifies the following:

Rules of At-Poker

1. **Forming the Hand:** Each player i forms their hand by concatenating the decks of the G_i players to their left, plus their own deck. This is done in a circular manner, where N is the total number of players. For example, if $N = 5$ and $G_i = 3$ for player $i = 3$, then player 3's hand is formed by concatenating the decks of players 5, 1, 2, and 3, in that order.
2. **Wildcards and Scores:** There are several strings placed on the table, which are called wildcards. Each wildcard string j has an associated value X_j .
3. **Calculating the Score:** A player's score is determined by the highest value among the wildcards that occur in their hand as a contiguous subsequence, where the occurrence of the wildcard must end within the player's original deck (i.e., it cannot be entirely within the decks of players to their left). If no wildcard occurs in the player's hand, their score is zero.

Determined not to leave Guadalajara as a loser, Granza set out to learn the strategies of At-Poker. He knew that the only way to return to Earth having recovered his losses would be if he could guarantee a win. To do so, he needed a system that could predict, based on the initial hands of each player, how many points each would score at the end of a round.

Entrada

The first line of input contains an integer N ($1 \leq N \leq 10^5$), representing the number of players at the table.

The next N lines describe each player. The i -th line contains a string s_i ($1 \leq |s_i| \leq 10^5$) of lowercase letters (from "a" to "z") and an integer G_i ($0 \leq G_i \leq N - 1$). The string s_i

represents player i 's deck, and the integer G_i indicates how many decks of players to their left are used to form their hand.

The following line contains an integer M ($1 \leq M \leq 10^5$), indicating the number of wildcards available on the table.

The next M lines list the wildcards. The i -th line contains a string t_i ($1 \leq |t_i| \leq 10^5$) and an integer X_i ($1 \leq X_i \leq 10^9$). The string t_i is the pattern to look for in the players' hands, and the integer X_i is the score assigned to this pattern if it is found in a player's hand, ending within their original deck.

It is guaranteed that the sum of the lengths of the strings s_i and the sum of the lengths of the strings t_i do not exceed 10^5 .

Saída

The output must consist of a single line containing N integers separated by spaces, where the i -th value is the score of player i in the game.

Exemplos

standard input	standard output
3 abd 0 dcbca 0 dbbca 0 3 ab 3 dc 7 c 5	3 7 5
5 adui 0 aba 0 gbcaffdf 0 abfh 0 abbfcafd 0 8 bfc 5 fd 4 ab 7 daduia 16 bagbc 12 bca 1 i 2 fh 10	2 7 4 10 7
5 adui 2 aba 2 gbcaffdf 1 abfh 0 abbfcafd 3 8 bfc 5 fd 4 ab 7 daduia 16 bagbc 12 bca 1 i 2 fh 10	2 16 12 10 7

Problema K: Kosmos

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

Limite de tempo: 1 second

Machado, a renowned researcher at the BRUTE Space Agency, invented a device he believes can reveal important patterns for the trajectory of planetary orbits. This device, named “Kosmos”, produced a revolutionary numeric sequence, but since it was programmed in pure C, a memory leak occurred which led to the device’s self-destruction. After discovering this, Machado immediately started programming everything in Rust. Now, shaken by the loss of his invention, the only thing he remembers is that the sequence produced by the device was defined by the following recursive formula:

$$\begin{aligned} F(0) &= 1 \\ F(1) &= 2 \\ F(n) &= F(n-1) \cdot F(n-2), \text{ if } n \geq 2 \end{aligned}$$

Kosmos had enough computational power to compute any term of this sequence from 1 to 10^{18} . Since the value of the term can be enormous, Machado is only interested in the remainder of this number modulo 998244353. However, without Kosmos, Machado has no idea how to compute the N -th term of this sequence if N is that large. Therefore, he has asked for your help.

Entrada

The input consists of a single line with an integer N ($0 \leq N \leq 10^{18}$).

Saída

Print the N -th term of the sequence generated by Kosmos. Since this value can be very large, print it modulo 998244353.

Exemplos

standard input	standard output
0	1
1	2
5	32
123456789123456789	433257388
998244353	470934745

Problema L: Lango Mocos

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

Dudu is a great astronaut working for BRUTE — Brazilian Universal Technological Expeditions — and he has been given the difficult task of traveling to the planet “LK-4D4” to collect some precious rocks classified as “lango mocos.”

After years in cryogenic sleep, Dudu finally arrived on this planet and began studying the lango mocos found there. After much research, he discovered that lango mocos contain chemical components never before seen by humans and that certain types of these rocks are toxic when in contact with certain other types.

He classified the lango mocos into N types and wrote down M pairs in his notebook. A pair (u, v) written in Dudu’s notebook means that lango moco type u is toxic when in contact with lango moco type v (and vice versa).

Fascinated by the lango mocos, Dudu decided he wants to bring all types of this precious rock back to Earth. Since the return trip is very long, Dudu must store the lango mocos in special bags that protect them from the dangers of space, and obviously, he doesn’t want to place two lango mocos that are toxic to each other in the same bag. Each bag can hold an infinite quantity of lango mocos, as they are made from an expandable material developed by BRUTE.

More formally, if a bag contains types $a_1, a_2, \dots, a_k$, there must be no pair (i, j) ($1 \leq i, j \leq k$) such that a_i is toxic to a_j .

Unfortunately, Dudu only has two special bags, and now he doesn’t know how to separate the lango mocos.

To solve this problem, Dudu sent a message to you, one of BRUTE’s best programmers, asking you to separate all the lango mocos into 2 bags, or tell him that the mission is impossible. If there is more than one way to make this separation, you can choose any of them.

Entrada

The first line of input contains two integers N ($1 \leq N \leq 10^5$) and M ($0 \leq M \leq \min(10^5, (N \cdot (N-1))/2)$), the number of lango moco types and the number of pairs Dudu wrote down in his notebook.

The next M lines each contain two integers u and v , representing that lango moco type u is toxic with lango moco type v (and vice versa).

Saída

The first line of output must consist of the word “POSSIVEL” (without quotes) if it is possible to make the separation, or “IMPOSSIVEL” (without quotes) if it is impossible.

If the answer is possible, you must print 3 additional lines:

The first of these lines must contain two integers K ($K \geq 0$) and H ($H \geq 0$) such that $K + H = N$, representing the number of lango mocos in the first and second bags, respectively.

The second line must contain K integers $a_1, \dots, a_K$, the types of lango mocos in the first bag.

The third line must contain H integers $b_1, \dots, b_H$, the types of lango mocos in the second bag.

The lango mocos in each bag can be printed in any order.

Exemplos

standard input	standard output
2 1 1 2	POSSIVEL 1 1 1 2
1 0	POSSIVEL 1 0 1
6 5 1 2 2 3 3 1 4 5 5 6	IMPOSSIVEL
5 4 1 2 1 3 1 4 3 5	POSSIVEL 2 3 1 5 2 3 4

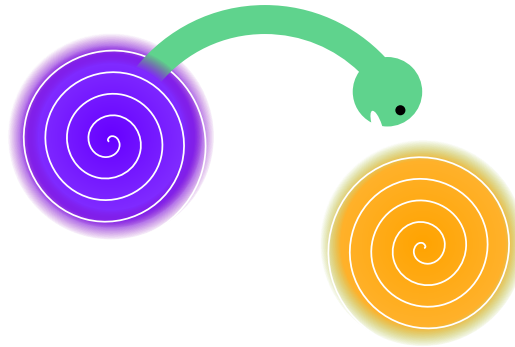
Problema M: Giant Worms

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

Limite de tempo: 1 second

Joãozinho is an astronomer who has just started his master's degree at UFMG, studying the “worms” of the multiverse.

The multiverse is the set of all universes that exist, and Joãozinho is researching a hierarchical way to represent it. In his studies, each universe in the multiverse is assigned an integer u as its identifier, and in each universe, there are several giant worms that allow travel to other universes.



The first thing Joãozinho did was to write down all the facts he knows about the multiverse. Here is everything he recorded:

- If a universe u has a giant worm that allows travel to universe v , then it is physically impossible for universe v to have a worm that allows travel back to universe u .
- The multiverse can be modeled as a directed tree, where the nodes of the tree are universes and an edge from universe u to v represents a giant worm that allows travel from u to v .
- There exists exactly one universe from which all other universes can be reached. Joãozinho labeled this universe with the number 1.
- If universe u has a worm that allows travel to universe v , then the number of stars in universe u is **strictly greater** than the number of stars in universe v .

Joãozinho's master's advisor, Karina, gave him a task to simulate any multiverse that obeys the rules of the real multiverse, and this simulation must be able to answer very specific questions about any multiverse.

After weeks of hard work, Joãozinho is almost done, but one “sub-task” remains, and he can't solve it. It is defined as follows:

Given K universes $a_1, a_2, \dots, a_K$, calculate the following sum:

$$\sum_{i=1}^K \sum_{j=i}^K f(i, j)$$

Where $f(i, j)$ is the number representing the universe with the smallest number of stars from which it is possible to travel to universes $a_i, a_{i+1}, \dots, a_{j-1}, a_j$.

Joãozinho is out of ideas on how to solve this, so he asked for your help to write a program that receives Q such queries and answers them.

Entrada

The first line of input contains two integers N ($1 \leq N \leq 10^5$) and Q ($1 \leq Q \leq 10^5$), the number of universes in the simulated multiverse and the number of queries, respectively.

The next $N - 1$ lines each contain two integers u and v ($1 \leq u, v \leq N$), indicating that there is a giant worm in universe u that allows travel to universe v .

It is guaranteed that the given multiverse satisfies all the facts and conventions Joãozinho recorded.

The next Q lines each contain an integer K ($1 \leq K \leq N$), followed by K integers $a_1, \dots, a_K$, the universes in the query. It is guaranteed that all universes in a query are distinct, and also **the sum of K over all queries does not exceed $3 \cdot 10^5$.**

Saída

The output must consist of Q lines, each containing a single integer — the answer to the corresponding query.

Exemplos

standard input	standard output
5 2 1 2 1 3 3 4 3 5 1 2 2 4 5	2 12
4 1 1 2 1 3 1 4 3 2 3 4	12
1 1 1 1	1

Notas

Explanation for the first example:

In the first query, only one universe is given, so the universe with the fewest stars that can reach universe 2 is universe 2 itself. In the second query, the answer is $f(1, 1) + f(1, 2) + f(2, 2) = 4 + 3 + 5 = 12$.

Problema N: Shield Navigation

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

Enzo is a big fan of the game Lemmings, and his favorite parts of the game are protecting the Lemmings from danger and when they build ramps to travel safely from one place to another. Since Enzo finds Lemmings a bit outdated, he decided to create a futuristic version of the game. He named his game “Almeidings 3077” and made it in 2D, viewed from above.

In Almeidings 3077, there are several creatures — the Almeidings — that must travel from a starting point (x_i, y_i) to an endpoint (x_f, y_f) safely. To parallel the ramp-building mechanics of the original game, Enzo created the concept of Protective Shields. A Protective Shield is a structure built by the Almeidings that allows them to pass safely through dangerous areas.

Enzo defined the following rules for his game:

- The game is in 2D, and the space can be represented by an $N \times M$ matrix.
- A Protective Shield has the shape of a cross and is centered at (x, y) . A Protective Shield centered at (x, y) protects all the cells in row x and all the cells in column y .
- An Almeiding can occupy cell (x, y) if and only if that cell is protected by a Protective Shield.
- The Almeidings can move in 4 directions (right, left, up, down):
 - From (x, y) to $(x + 1, y)$.
 - From (x, y) to $(x - 1, y)$.
 - From (x, y) to $(x, y + 1)$.
 - From (x, y) to $(x, y - 1)$.

After deciding that the goal of the game is to make the Almeidings conquer the space, Enzo decided to simulate it with your help. He will ask you Q queries, and each query can be one of two types:

- Type 1: Make an Almeiding build a Protective Shield centered at (x, y) .
- Type 2: If there is an Almeiding at (x_i, y_i) , is it possible for it to reach (x_f, y_f) ?

Your task is to answer all type 2 queries correctly.

Entrada

The first line of input contains three integers N , M ($1 \leq N \cdot M \leq 10^6$), the dimensions of the matrix representing the game, followed by Q ($1 \leq Q \leq 2 \cdot 10^5$), the number of queries Enzo will make.

The next Q lines represent the queries, which can be of two types:

- 1 x y : Type 1, build a Protective Shield centered at (x, y) ($1 \leq x \leq N, 1 \leq y \leq M$).

- 2 $x_i y_i x_f y_f$: Type 2, check if an Almeiding can move from (x_i, y_i) to (x_f, y_f) ($1 \leq x_i, x_f \leq N$), ($1 \leq y_i, y_f \leq M$).

Saída

For each query of type 2, print a line containing “SIM” or “NAO”, the answer to the query.

Exemplos

standard input	standard output
3 3 3 1 2 2 2 2 1 2 3 2 1 1 3 3	SIM NAO
3 3 4 2 1 1 3 3 1 1 1 2 1 1 3 3 2 3 1 1 3	NAO NAO SIM

Problema O: Osmos V

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

Limite de tempo: 1 second

Malu is on a journey to the distant planet Osmos V aboard her spaceship, the HB-20000. Traveling at a speed close to the speed of light, Malu cannot determine her exact position in the universe. However, she knows that the total duration of the trip is exactly X years and that Y years have already passed.

How many years are left until Malu reaches her destination?

Entrada

The input consists of two lines.

The first line contains the integer X ($1 \leq X \leq 10^9$), representing the total duration of the trip in years.

The second line contains an integer Y ($0 \leq Y \leq X$), representing how many years have already passed.

Saída

The output must contain a single integer, representing the number of years remaining until Malu reaches her destination.

Exemplos

standard input	standard output
10 5	5
1 1	0
100000 4312	95688