

Задача А. Тест

Обмеження використання часу: 1 секунда
Обмеження використання пам'яті: 256 мегабайтів

Ви складаєте тест з інформатики. Відомо, що на тесті є a завдань, кожне з яких оцінюється в один бал.

Ви дали відповідь на b завдань.

Для того, щоб скласти тест, потрібно дати правильну відповідь на як мінімум c завдань.

Яку максимальну кількість помилок ви могли зробити у тих b завданнях, щоб й досі скласти тест?

Формат вхідних даних

Перший рядок містить одне ціле число a ($1 \leq a \leq 50$) — кількість завдань.

Другий рядок містить одне ціле число b ($1 \leq b \leq a$) — кількість завдань, на які ви дали відповідь.

Третій рядок містить одне ціле число c ($1 \leq c \leq b$) — мінімальна кількість завдань, на які потрібно дати правильні відповіді, щоб скласти тест.

Гарантується, що відповідь завжди невід'ємна.

Формат вихідних даних

Виведіть одне ціле число — відповідь на задачу.

Приклад

стандартний ввід	стандартний вивід
10 8 5	3

Задача В. Середнє арифметичне

Обмеження використання часу: 1 секунда
Обмеження використання пам'яті: 256 мегабайтів

У Козака Вуса є два числа a та b ($a \leq b$). Він знайшов середнє арифметичне цих чисел (нехай це буде число c). Тобто, $c = \frac{a+b}{2}$.

Вам дано число a (тобто менше з тих двох чисел), а також дано число c . Знайдіть b .

Формат вхідних даних

Перший рядок містить одне ціле число a ($1 \leq a \leq 100$).

Другий рядок містить одне ціле число c ($a \leq c \leq 100$).

Формат вихідних даних

Виведіть одне ціле число — b .

Приклад

стандартний ввід	стандартний вивід
3 7	11

Задача С. Гітара

Обмеження використання часу: 1 секунда
Обмеження використання пам'яті: 256 мегабайтів

Пан Т хоче вивчити нову пісню на 3-струнній гітарі. Він завантажив табулатури для неї — послідовність нот для кожної струни, які потрібно грати протягом пісні.

Однак іноді пісні передбачають якісь божевільні позиції пальців на гітарному рифі, наприклад, розміщення одного пальця на 1-й ноті, а іншого на 11-й, що вимагає неймовірної розтяжки! Пан Т, наприклад, може одночасно дотягуватися лише до нот, які **не далі ніж на 3 одиниці одна від одної**.

Пісня триває n секунд, а табулатури представлені 3 масивами s_1, s_2, s_3 , де $s_{i,j}$ представляє ноту, яку потрібно грати на i -й струні в j -й секунді. Якщо $s_{i,j} = -1$, то i -та струна просто не грається в момент j (тому пану Т не потрібно розтягувати пальці, щоб дотягнутися до неї); якщо $s_{i,j} = 0$, то i -та струна грається "відкрито" — тобто пану Т також не потрібно її торкатися. Якщо $s_{i,j} > 0$, то пан Т повинен розмістити свій палець на цій струні точно на $s_{i,j}$ -й ноті в момент j .

Тепер йому цікаво, чи може він зіграти всю пісню, тобто чи не існує жодного моменту часу, коли йому доведеться тримати дві ноти, які занадто далеко одна від одної. Якщо так, виведіть перший момент часу, коли це трапляється.

Формат вхідних даних

Перший рядок вхідних даних містить єдине ціле число n ($1 \leq n \leq 1000$) — довжина пісні.

Наступні 3 рядки містять по n цілих чисел $s_{i,1}, s_{i,2}, \dots, s_{i,n}$ ($-1 \leq s_{i,j} \leq 19$).

Формат вихідних даних

Виведіть єдиний токен — «Yes», якщо пан Т може зіграти всю пісню, і «No», якщо не може. Якщо відповідь «No», у другому рядку виведіть єдине ціле число t , що позначає перший момент часу, в якому він зазнає невдачі.

Приклади

стандартний ввід	стандартний вивід
4 0 1 0 3 1 1 1 0 2 2 0 0	Yes
2 0 0 6 1 7 5	No 2
1 0 7 -1	Yes

Примітка

У першому прикладі пан Т ніколи не повинен дотягуватися до нот на відстані більше ніж 1 в

будь-який момент часу.

У другому прикладі, під час 2-ї секунди пан Т повинен грати 2-гу струну на ноті 1, одночасно граючи 3-ю струну на 5-й ноті, при цьому відстань між ними становить 4, що занадто далеко для пана Т.

У третьому прикладі пан Т не повинен грати 3-ю струну і не потрібно торкатися 1-ї струни, тому він може розмістити свій єдиний палець на струні 2 на будь-якій ноті.

Задача D. EVO проти STI

Обмеження використання часу: 1 секунда
Обмеження використання пам'яті: 256 мегабайтів

Два улюблені гонщики пана Т — EVO Guy та STI Guy змагатимуться у ралійному чемпіонаті.

n гоночних трас доступні для чемпіонату. Організатори мають ідеальне середовище для симуляції, тому вже заздалегідь знають, скільки часу кожен гонщик витратить на кожній з трас — EVO Guy знадобиться e_i секунд, щоб завершити одне коло на трасі i , а STI Guy — s_i секунд відповідно.

Після проведення всіх гонок, рахунок кожного гонщика розраховується як загальна кількість секунд, які він витратив на всіх трасах, на яких брав участь. Той, у кого менший рахунок, виграє чемпіонат; якщо рахунки рівні, оголошується нічия.

Звичайно, організатори не можуть провести більше однієї гонки на одній трасі, і вони можуть вирішити не проводити гонки на деяких трасах взагалі. Однак вони отримують гроші від спонсорів за кожну проведену гонку, тому їх цікавить проведення якомога більшої кількості гонок.

Тепер організатори цікавляться, якщо вони хочуть, щоб певний гонщик виграв чемпіонат, яка буде найбільша кількість гонок, яку вони можуть провести? Їх цікавлять обидва числа, тому вони просять вас визначити їх — R_E , найбільша кількість проведених гонок, щоб переміг EVO Guy, та R_S , найбільша кількість проведених гонок, щоб переміг STI Guy. Якщо незалежно від вибраних трас гонщик не може виграти чемпіонат, виведіть 0 замість відповідного числа.

Формат вхідних даних

Перший рядок вхідних даних містить одне ціле число n ($3 \leq n \leq 2000$) — кількість доступних гоночних трас.

Другий рядок містить n цілих чисел $e_1, e_2, \dots, e_n$ ($1 \leq e_i \leq 7200$) — кількість секунд, які EVO Guy витрачає на завершення i -ї траси.

Третій рядок містить n цілих чисел $s_1, s_2, \dots, s_n$ ($1 \leq s_i \leq 7200$) — кількість секунд, які STI Guy витрачає на завершення i -ї траси.

Формат вихідних даних

Виведіть два цілі числа R_E та R_S — найбільша кількість гонок, проведених з перемогою EVO Guy або STI Guy у чемпіонаті відповідно, або 0, якщо відповідний гонщик не може виграти чемпіонат, незалежно від вибраних трас.

Приклади

стандартний ввід	стандартний вивід
5 12 13 11 13 15 12 10 10 16 16	4 4
5 60 63 37 61 90 61 59 33 70 90	5 4
3 100 100 100 101 101 100	3 0

Примітка

У першому прикладі організатори можуть провести гонки на трасах 1, 3, 4, 5, і рахунок EVO Guу буде $12 + 11 + 13 + 15 = 51$, тоді як рахунок STI Guу буде $12 + 10 + 16 + 16 = 54$, тому EVO Guу виграє з 4 проведеними гонками. Водночас, якщо вони проведуть гонки на трасах 1, 2, 3, 5, то рахунок EVO Guу буде $12 + 13 + 11 + 15 = 51$, тоді як рахунок STI Guу буде $12 + 10 + 10 + 16 = 48$. Відповідно, STI Guу виграє з 4 проведеними гонками.

Якщо всі 5 гонок будуть проведені, рахунки будуть $12 + 13 + 11 + 13 + 15 = 64$ та $12 + 10 + 10 + 16 + 16 = 64$ без переможця.

У третьому прикладі на будь-якій трасі STI Guу витрачає більше часу на завершення кола, ніж EVO Guу, тому він не може виграти незалежно від вибраних трас.

Задача Е. Шифр

Обмеження використання часу: 1 секунда
Обмеження використання пам'яті: 256 мегабайтів

Учасники UJOI бояться голови UJOI, тому вони вигадали шифр, щоб приховати свої повідомлення від нього. Шифр простий — вони генерують 26 **різних** чисел і призначають рівно одне для кожної англійської літери. Тепер, коли вони спілкуються, вони використовують його для кодування своїх повідомлень.

Голова UJOI може таємно переглядати їхні повідомлення, але не знає шифру. Йому цікаво дізнатися, чи з'являється його ім'я в повідомленнях; наприклад, якби його ім'я було "ILLIA", один із способів, як його можна зашифрувати за допомогою шифру, це [1, 2, 2, 1, 3] (тут 1 означає "I", 2 для "L", 3 для "A"), тому, якщо він бачить цю підпоследовність у повідомленні, він знає, що це може бути його ім'я.

Дано його ім'я та зашифроване повідомлення, він хоче дізнатися, чи можливо, що його ім'я з'являється в повідомленні. Іншими словами, чи існує послідовність з 26 цілих чисел, така що, якщо k -та літера англійського алфавіту в його імені буде замінена на k -те з цих цілих чисел, воно з'явиться як підрядок у повідомленні.

Крім того, якщо так, виведіть послідовність з 26 цілих чисел, що представляють шифр.

Формат вхідних даних

Перший рядок вхідних даних містить два цілі числа n ($1 \leq n \leq 10^3$) та m ($1 \leq m \leq 10$) — довжина повідомлення та довжина імені.

Другий рядок містить n цілих чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 26$) — повідомлення, закодоване за допомогою шифру. Гарантується, що не більше 26 унікальних чисел.

Третій рядок містить рядок T з m великих англійських літер, ім'я голови UJOI.

Формат вихідних даних

У першому рядку виведіть один рядок — "Yes" або "No", відповідаючи на питання, чи існує такий шифр, що ім'я, закодоване ним, з'являється в повідомленні.

Крім того, якщо відповідь "Yes", виведіть 26 цілих чисел $c_1, c_2, c_3, \dots, c_{26}$ ($1 \leq c_i \leq 26$), де c_k представляє ціле число, що кодує k -ту літеру латинського алфавіту.

Для кожного $1 \leq i < j \leq 26$ повинно виконуватись $c_i \neq c_j$.

Приклади

стандартний ввід	стандартний вивід
10 5 19 9 7 13 1 1 14 20 15 14 ANTON	Yes 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
5 5 1 2 2 1 3 ILLIA	Yes 3 4 5 6 7 8 9 10 1 11 12 2 13 14 15 16 17 18 19 20 21
7 6 7 7 7 7 7 7 7 AKSHAD	No

Примітка

У першому прикладі кодування імені “ANTON” за допомогою шифру $[1, 2, 3, \dots, 26]$ дає $[1, 14, 20, 15, 14]$, що з’являється як підрядок у повідомленні починаючи з позиції 2.

У другому прикладі кодування імені “ILLIA” за допомогою шифру з виходу дає $[1, 2, 2, 1, 3]$, що є самим повідомленням.

У третьому прикладі легко побачити, що не існує шифру, який би перетворив “AKSHAD” на 6 однакових цілих чисел.

Задача F. Пінгвіни

Обмеження використання часу: 1 секунда
Обмеження використання пам'яті: 256 мегабайтів

Містер ІІ має плюшевого пінгвіна, тому пінгвіни з Байтеланду попросили його про допомогу. Є n родин пінгвінів; i -та родина має $1 \leq a_i \leq 3$ пінгвінів.

Вони хочуть побудувати k міст і просять Містера ІІ розділити їх на k команд для цього. Звичайно, всі пінгвіни з однієї родини повинні бути в одній команді, а команди повинні мати якомога схожі розміри. Іншими словами, ви повинні розділити родини на k команд так, щоб різниця між найбільшою та найменшою командами була мінімізована.

Формат входніх даних

Перший рядок містить два цілі числа n та k ($2 \leq k \leq n \leq 2 \cdot 10^5$) — кількість родин та кількість команд.

Другий рядок містить n цілих чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 3$) — кількість пінгвінів у i -тій родині.

Формат вихідних даних

Виведіть n цілих чисел — $t_1, t_2, \dots, t_n$ з $1 \leq t_i \leq k$, що позначають номер команди i -ї родини так, щоб всі розміри команд були якомога ближчими.

Приклади

стандартний ввід	стандартний вивід
6 3 1 2 3 1 2 3	2 1 2 3 1 3
5 2 1 2 3 2 3	2 1 1 2 2
5 3 3 3 3 3 2	3 1 2 3 2

Примітка

У першому прикладі розміри команд 1, 2, 3 становлять $(2 + 2)$, $(1 + 3)$ та $(1 + 3)$ відповідно; отже, всі розміри команд рівні, і таким чином розподіл є оптимальним.

У другому прикладі розміри становлять $(2 + 3) = 5$ та $(1 + 2 + 3) = 6$ з різницею 1. Можна показати, що неможливо сформувати дві команди з різницею розміру, що дорівнює 0.

У третьому прикладі розміри команд становлять (3) , $(3 + 2) = 5$, $(3 + 3) = 6$ з різницею 3. Можна показати, що неможливо сформувати команди з різницею розміру менше ніж 3.

Задача G. Множення запитів

Обмеження використання часу: 1 секунда
Обмеження використання пам'яті: 256 мегабайтів

Містер IP має два масиви a та b , обидва довжини n , і ціле число Y . Спочатку $b_i = 0$ для всіх $1 \leq i \leq n$.

Тепер він хоче обробити q оновлень наступного вигляду:

Для i -го оновлення задаються три цілі числа l_i, r_i, x_i . Потім виконується наступне:

- для кожного $l_i \leq j \leq r_i$, виконуємо $a_j := a_j \cdot x$.
- після цього, для кожного $1 \leq k \leq n$ (зверніть увагу, що для всіх k , а не лише між l та r), якщо $a_k \geq Y$, додаємо 1 до b_k .

Тепер, після виконання всіх оновлень, містер IP хоче дізнатися значення b_i для кожного $1 \leq i \leq n$.

Формат вхідних даних

Перший рядок містить три цілі числа n ($2 \leq n \leq 2 \cdot 10^5$), q ($1 \leq q \leq 2 \cdot 10^5$), Y ($2 \leq Y \leq 10^6$) — кількість елементів у масивах a та b , кількість оновлень та параметр Y .

Другий рядок містить n цілих чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i < Y$).

Кожен з наступних q рядків містить три цілі числа l_i, r_i, x_i ($1 \leq l_i \leq r_i \leq n$, $1 \leq x_i < Y$).

Формат вихідних даних

Виведіть n цілих чисел в одному рядку — значення $b_1, b_2, \dots, b_n$ після обробки всіх оновлень.

Приклад

стандартний ввід	стандартний вивід
5 5 5 1 1 1 1 1 2 4 3 1 3 2 1 5 1 1 5 2 1 5 2	1 4 4 2 0

Примітка

Після першого оновлення масив a виглядає наступним чином: $[1, 3, 3, 3, 1]$. Масив $b = [0, 0, 0, 0, 0]$.

Після другого оновлення, $a = [2, 6, 6, 3, 1]$. Масив $b = [0, 1, 1, 0, 0]$.

Після третього оновлення, $a = [2, 6, 6, 3, 1]$. Масив $b = [0, 2, 2, 0, 0]$.

Після четвертого оновлення, $a = [4, 12, 12, 6, 2]$. Масив $b = [0, 3, 3, 1, 0]$.

Після п'ятого оновлення, $a = [8, 24, 24, 12, 4]$. Масив $b = [1, 4, 4, 2, 0]$.