

Задача А. Пиріжки

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

У Козака Вуса є рівно n пиріжків та m друзів.

Відомо, що кожен його друг хоче, щоб йому дали рівно три пиріжки.

Козак Вус не підводить своїх друзів, тому він віддасть кожному те, що він хоче.

Визначте, скільки пиріжків залишиться у Козака Вуса, якщо кожному другу він віддасть три пиріжки. Відомо, що у Козака достатньо пиріжків.

Формат вхідних даних

Перший рядок містить одне ціле число n ($3 \leq n \leq 100$) — кількість пиріжків.

Другий рядок містить одне ціле число m ($1 \leq m \leq 100$) — кількість друзів Козака Вуса.

Гарантується, що у Вуса буде достатньо пиріжків, щоб роздати всім його друзям.

Формат вихідних даних

Виведіть одне ціле число — кількість пиріжків, які залишаться у Козака Вуса після того, як він їх роздасть своїм друзям.

Приклад

стандартний ввід	стандартний вивід
10 2	4

Примітка

У Козака два друга, кожному з них він віддасть три пиріжки. Сумарно він віддасть шість, тому залишаться лише чотири пиріжки.

Задача В. Анфіса і квіти

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

Мурзик одну з квіткових клумб зробив у вигляді шахової дошки розмірами $m \times n$, у кожній клітинці якої росте якась квітка. Інколи на цю клумбу він виводить на прогулянку Анфісу (так, не дивуйтеся, вони дійсно друзі). Анфіса, починаючи завжди з верхнього лівого кута, переміщується по клумбі до правого нижнього і збирає квіти, причому таким чином, щоб щоразу проходити по новому маршруту, а Мурзик на виході вручає їй шматочок сиру.

Порахуйте, яка найбільша кількість шматочків сиру дістанеться Анфісі, якщо вона весь час намагатиметься зберегти якнайбільше квітів. При кожному своєму переміщенні Анфіса обов'язково має зібрати як мінімум одну квітку.

Формат вхідних даних

Перший рядок містить одне ціле число n ($1 \leq n \leq 1000$).

Другий рядок містить одне ціле число m ($1 \leq m \leq 1000$).

Формат вихідних даних

Виведіть одне ціле число — відповідь на задачу.

Приклад

стандартний ввід	стандартний вивід
3 4	7

Задача С. НМТ

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

Ви склали національний мультипредметний тест (НМТ) і отримали чотири бали p_1, p_2, p_3, p_4 з чотирьох предметів.

Університет оцінює абітурієнтів, використовуючи ваги w_1, w_2, w_3, w_4 для цих предметів. Ваш загальний бал для вступу становить

$$\text{score} = p_1w_1 + p_2w_2 + p_3w_3 + p_4w_4.$$

Університет прийме вас, якщо і тільки якщо ваш бал буде не менше S (тобто $\text{score} \geq S$).

Визначте, чи будете ви зараховані.

Формат вхідних даних

Перший рядок містить чотири цілі числа p_1, p_2, p_3, p_4 ($0 \leq p_i \leq 200$) — ваші бали з чотирьох предметів.

Другий рядок містить чотири цілі числа w_1, w_2, w_3, w_4 ($0 \leq w_i \leq 5$) — ваги, які використовує університет.

Третій рядок містить одне ціле число S ($0 \leq S \leq 5000$) — мінімальний бал, необхідний для вступу.

Формат вихідних даних

Виведіть YES, якщо ви будете зараховані (ваш загальний бал не менше S). В іншому випадку виведіть NO.

Приклади

стандартний ввід	стандартний вивід
100 120 80 90 1 1 1 1 350	YES
200 200 200 200 1 2 3 4 2000	YES
100 100 100 100 1 1 1 1 500	NO

Примітка

У першому прикладі бал становить $100 + 120 + 80 + 90 = 390$, що не менше 350, тому відповідь YES.

У другому прикладі бал становить

$$200 \cdot 1 + 200 \cdot 2 + 200 \cdot 3 + 200 \cdot 4 = 2000,$$

що дорівнює $S = 2000$, тому правильна відповідь YES.

У третьому прикладі бал становить

$$100 \cdot 1 + 100 \cdot 1 + 100 \cdot 1 + 100 \cdot 1 = 400,$$

що менше, ніж $S = 500$, тому правильна відповідь NO.

Задача D. Змагання

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

Є N команд, запрошених на змагання. Під час виступу спонсора кожна команда повинна бути присутня більшу частину часу. Якщо команда відсутня більше ніж X безперервних хвилин у будь-який момент, ця команда буде дискваліфікована.

Виступ спонсора починається о 0. Спочатку, о 0, всі команди присутні.

Вам дано Q подій. Кожна подія описується двома цілими числами `time` та `idx`:

- `idx` — індекс команди ($1 \leq \text{idx} \leq N$),
- `time` — час події (в хвилинах, $0 \leq \text{time} \leq 10^9$).

Коли відбувається подія (`time`, `idx`):

- якщо команда `idx` наразі присутня, вона йде (стає відсутньою);
- якщо команда `idx` наразі відсутня, вона повертається (стає присутньою).

Між подіями стан кожної команди (присутня чи відсутня) не змінюється.

Нехай T — максимальний час серед усіх подій. Виступ спонсора закінчується о T . Для кожної команди розгляньте всі її безперервні інтервали відсутності протягом часового сегмента $[0, T]$. Якщо існує такий інтервал, довжина якого є **суворо більшою** за X , то ця команда дискваліфікована. Відсутність протягом точно X безперервних хвилин дозволена.

Визначте, які команди дискваліфіковані.

Формат вхідних даних

Перший рядок містить три цілі числа N , Q та X ($1 \leq N \leq 200000$, $0 \leq Q \leq 200000$, $1 \leq X \leq 10^9$).

Кожен з наступних Q рядків містить два цілі числа `time` та `idx` ($0 \leq \text{time} \leq 10^9$, $1 \leq \text{idx} \leq N$) — одна подія.

Події подані в довільному порядку. Для кожної команди жодні дві події не мають однакового часу.

Формат вихідних даних

У першому рядку виведіть кількість дискваліфікованих команд.

У другому рядку виведіть індекси всіх дискваліфікованих команд у зростаючому порядку, розділені пробілами.

Приклад

стандартний ввід	стандартний вивід
3 4 4	2
2 1	1 2
7 1	
3 2	
8 2	

Примітка

У прикладі, $N = 3$, $X = 4$. Усі команди присутні о 0. Події в хронологічному порядку:

- час 2, команда 1 йде;
- час 3, команда 2 йде;
- час 7, команда 1 повертається;
- час 8, команда 2 повертається.

Останній час події $T = 8$, тому ми розглядаємо інтервал $[0, 8]$.

- Команда 1 відсутня з часу 2 до часу 7 (тривалість 5). Оскільки $5 > X = 4$, команда 1 дискваліфікована.
- Команда 2 відсутня з часу 3 до часу 8 (тривалість 5). Це також більше ніж 4, тому команда 2 дискваліфікована.
- Команда 3 ніколи не йде, тому вона завжди присутня і не дискваліфікована.

Отже, дискваліфікованими командами будуть 1 і 2.

Задача Е. Видалення тестів

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

Зверніть увагу, що у цій задачі вам, можливо, буде корисно використовувати Python 3.10 (PyPy) замість Python 3.13.

Містер ІР підготував задачу для майбутнього змагання. Він реалізував n рішень і створив m тестів.

Однак сьогодні він дізнався, що сервер буде повільнішим, ніж очікувалося, і тепер йому потрібно видалити деякі тести, щоб уникнути довгих черг на оцінювання під час змагання. Він повинен залишити принаймні один і не більше ніж t з m ($t < m$) початкових тестів.

Для цього він вирішив придумати якісь критерії для ранжування тестів, щоб зберегти якість тестового набору. Для кожного рішення він виміряв його продуктивність на тестовому наборі і позначив $a_{i,j}$ як витрати часу i -го рішення на j -му тесті.

Оригінальні тести були створені таким чином, що для будь-яких двох тестів j і k одне й те саме рішення ніколи не витрачає однакою кількість часу. Іншими словами, для будь-якого $1 \leq i \leq n$ і будь-яких $1 \leq j, k \leq m$, якщо $j \neq k$, то $a_{i,j} \neq a_{i,k}$.

Природно, містер ІР хоче видалити тести, на яких кожне з рішень працює швидко, тому він придумав наступну стратегію — спочатку він визначає додатний цілий параметр d , а потім для кожного рішення i він позначає як "недійсні" всі тести j , для яких існує принаймні d інших тестів, на яких це рішення витрачає більше часу, ніж на тесті j . Після цього він видаляє всі тести, які позначені як "недійсні" для кожного з n рішень. Звичайно, принаймні один тест повинен залишитися після цього.

Оскільки не завжди можливо вибрати таке значення d , не видаливши **всі тести**, ви повинні вивести -1 , коли такого d не існує.

Допоможіть містеру ІР знайти найбільше значення d , для якого він видалить достатньо тестів, щоб в кінці залишилося не більше t з них, або встановіть, що такого значення немає.

Формат вхідних даних

Перший рядок містить три цілі числа n, m, t ($1 \leq n \leq 1500$, $1 \leq t < m \leq 1500$) — кількість рішень, кількість тестів і максимальна кількість тестів, які можуть залишитися після видалення.

Кожен з наступних n рядків містить m різних цілих чисел: $a_{i,1}, a_{i,2}, \dots, a_{i,m}$ ($1 \leq a_{i,j} \leq 10^6$), де $a_{i,j}$ є часом, витраченим рішенням i на тесті j . Для кожного фіксованого i всі $a_{i,j}$ парно різні.

Формат вихідних даних

Виведіть єдине ціле число — максимальне значення d , таке що після того, як містер ІР видалить всі тести, які позначені як "недійсні" для кожного рішення, кількість залишених тестів буде не більше t , або -1 , якщо такого значення не існує.

Система оцінювання

У кожному блоці тести оцінюються незалежно.

1. (20 балів): $n, m \leq 50$;
2. (20 балів): $n, m \leq 400$;
3. (60 балів): без додаткових обмежень.

Приклади

стандартний ввід	стандартний вивід
3 6 4 5 1 4 6 3 2 10 3 4 1 2 5 8 9 1 7 3 4	2
4 5 3 3 10 6 1 8 5 2 7 9 1 4 3 8 6 2 9 1 5 4 7	-1

Примітка

У першому тестовому випадку, з $d = 2$, для першого рішення тести 2, 3, 5, 6 позначені як "недійсні". Для другого та третього рішень "недійсні" тести це 2, 3, 4, 5 та 3, 4, 5, 6 відповідно.

Ми можемо бачити, що лише тести 3 і 5 є "недійсними" для кожного з 3 рішень, тому лише ці два тести будуть видалені.

Можна перевірити, що для більших значень d принаймні 5 тестів залишаються після видалення.

Задача F. Банк

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

Містер IP витратив багато грошей цього місяця — він заплатив n разів своєю банківською картою, і i -та оплата коштувала йому точно 2^{a_i} грн.

Тепер його банківський додаток згрупував його витрати за категоріями (наприклад, ресторани, продукти, таксі, ігри тощо). Всього є k категорій.

Однак його бос буде дуже сердитий, якщо дізнається, що містер IP витратив занадто багато грошей на будь-яку конкретну категорію. На щастя, ви знайшли вразливість у додатку, яка дозволяє вам редагувати категорію будь-якої оплати.

Тепер ви хочете використати це, щоб допомогти містеру IP зробити найбільшу суму витрат для однієї категорії якомога меншою серед усіх категорій. Іншими словами, якщо після внесення змін сума витрат для i -ї категорії дорівнює s_i , ви хочете мінімізувати значення $\max_{i=1}^k s_i$.

Містер IP боїться свого боса і просить вас допомогти йому. Він хоче, щоб ви вивели n цілих чисел $c_1, c_2, \dots, c_n$, де $1 \leq c_i \leq k$ позначає номер категорії, до якої ви призначили i -ту оплату.

Формат вхідних даних

Перший рядок містить два цілі числа n і k ($1 \leq k \leq n \leq 2 \cdot 10^5$) — кількість оплат та кількість категорій.

Другий рядок містить n цілих чисел $a_1, a_2, \dots, a_n$ ($0 \leq a_i < n$).

Формат вихідних даних

Виведіть n цілих чисел — значення $c_1, c_2, \dots, c_n$ ($1 \leq c_i \leq k$), що дають призначення категорій для оплат, яке мінімізує найбільшу суму витрат для однієї й тієї ж категорії.

Система оцінювання

У кожному блоці тести оцінюються незалежно.

- (40 балів): $a_i < 30$;
- (60 балів): без додаткових обмежень.

Приклади

стандартний ввід	стандартний вивід
4 2 3 2 1 0	1 2 2 2
8 3 1 1 1 1 1 1 1 1	2 1 3 2 1 3 2 1
7 3 1 1 1 5 1 1 1	2 2 2 1 2 2 2

Примітка

У першому тесті витрати становлять $\{2^3, 2^2, 2^1, 2^0\} = \{8, 4, 2, 1\}$ грн. Розділивши їх на дві групи $\{8\}$ і $\{4, 2, 1\}$, ми можемо досягти найбільших витрат для однієї групи, що становитиме 8. Ми можемо показати, що зробити краще неможливо.

У другому тесті один зі способів досягнення відповіді полягає в розподілі витрат на $\{2^1, 2^1, 2^1\}$, $\{2^1, 2^1, 2^1\}$, $\{2^1, 2^1\}$, при цьому найбільша сума витрат для однієї категорії становитиме $2 + 2 + 2 = 6$. Ми можемо показати, що зробити краще неможливо.

У третьому тесті, незалежно від того, як ми групуємо витрати, одна категорія завжди міститиме елемент $2^5 = 32$, отже, одне можливе призначення це $\{2^5\}$, $\{2^1, 2^1, 2^1, 2^1, 2^1, 2^1\}$, $\{\}$, що досягає найменшої можливої відповіді 32.

Задача G. Матриця

Обмеження використання часу: 3 секунди
Обмеження використання пам'яті: 256 мегабайтів

У вас є квадратна бінарна матриця a розміром $n \times n$. Спочатку всі її елементи дорівнюють нулю. Вам дано q оновлень. Кожне оновлення має наступний формат:

- $x \ y$ — встановити $a_{x,y} := 1$. Якщо ця клітинка вже була 1, вона залишиться 1.

Координати x та y індексуються з 1.

Після початкової матриці, що складається лише з нулів (перед обробкою будь-яких оновлень), і після кожного з q оновлень, вам потрібно знайти довжину сторони найбільшої квадратної підматриці, що складається лише з 1.

Формат вхідних даних

Перший рядок містить два цілі числа n та q ($1 \leq n \leq 500$, $1 \leq q \leq 250000$) — розмір матриці та кількість оновлень.

Кожен з наступних q рядків містить два цілі числа x та y ($1 \leq x, y \leq n$) і описує оновлення $x \ y$ — встановити $a_{x,y} := 1$.

Формат вихідних даних

Виведіть $q + 1$ рядків.

У першому рядку виведіть довжину сторони найбільшої квадратної підматриці з усіх одиниць у початковій матриці (яка складається лише з нулів).

Потім, після кожного оновлення, виведіть довжину сторони найбільшої квадратної підматриці з усіх одиниць у поточній матриці.

Система оцінювання

У кожному блоці тести оцінюються незалежно.

1. (25 балів): $n \leq 10$;
2. (33 балів): $n \leq 100$;
3. (42 балів): без додаткових обмежень.

Приклад

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

Примітка

У прикладі матриця після кожного запиту виглядає так:

- Після 1 1:

1	0	0
0	0	0
0	0	0

- Після 1 2:

1	1	0
0	0	0
0	0	0

Найбільший квадрат з усіх одиниць має довжину сторони 1.

- Після 2 2:

1	1	0
0	1	0
0	0	0

- Після 2 1:

1	1	0
1	1	0
0	0	0

Тепер найбільший квадрат з усіх одиниць має довжину сторони 2 (верхній лівий кут у $(1,1)$).

Задача Н. Мін-Макс-Множення!

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

Містер ІР знайшов цілий масив a довжини n . Він вільний на вихідних, тому знайшов собі заняття:
Для кожного підмасиву $a[l..r]$ ($1 \leq l \leq r \leq n$) визначте

$$\text{mn}(l, r) = \min(a_l, a_{l+1}, \dots, a_r),$$

$$\text{mx}(l, r) = \max(a_l, a_{l+1}, \dots, a_r).$$

Тепер його цікавить значення

$$S = \sum_{l=1}^n \sum_{r=l}^n \text{mn}(l, r) \cdot \text{mx}(l, r).$$

Оскільки воно може бути дуже великим, він хоче знати лише його залишок при діленні на $10^9 + 7$ (іншими словами, $S \bmod (10^9 + 7)$). Допоможіть йому обчислити це!

Формат вхідних даних

Перший рядок містить єдине ціле число n ($1 \leq n \leq 5 \cdot 10^5$) — довжина масиву.

Другий рядок містить n цілих чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^6$).

Формат вихідних даних

Виведіть єдине ціле число — значення залишку S за модулем $10^9 + 7$.

Система оцінювання

У кожному блоці тести оцінюються незалежно.

- (20 балів): $n \leq 2000$;
- (10 балів): $a_i \leq a_{i+1}$ для всіх $1 \leq i \leq n - 1$;
- (20 балів): серед елементів a не більше 2 різних значень;
- (30 балів): $n \leq 10^5$;
- (20 балів): без додаткових обмежень.

Приклади

стандартний ввід	стандартний вивід
3 1 2 3	25
4 2 1 3 2	38

Примітка

У першому прикладі всі підмасиви та їх внески:

- $[1]$: $\min = 1$, $\max = 1$, внесок $1 \cdot 1 = 1$
- $[1, 2]$: $\min = 1$, $\max = 2$, внесок $1 \cdot 2 = 2$
- $[1, 2, 3]$: $\min = 1$, $\max = 3$, внесок $1 \cdot 3 = 3$
- $[2]$: $\min = 2$, $\max = 2$, внесок $2 \cdot 2 = 4$
- $[2, 3]$: $\min = 2$, $\max = 3$, внесок $2 \cdot 3 = 6$
- $[3]$: $\min = 3$, $\max = 3$, внесок $3 \cdot 3 = 9$

Сума всіх внесків дорівнює $1 + 2 + 3 + 4 + 6 + 9 = 25$.