Задача A. Ділення націло на 5

Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

На столі лежить n камінців. За 1 монету ви можете зробити одну з наступних операцій:

- Забрати зі столу один камінець. **Ви не можете виконати цю операцію, якщо на столі нема жодного камінця**.
- Покласти на стіл ще один камінець.

Яку найменшу кількість монет треба витратити, щоб число камінців на столі почало ділитись націло на 5?

Зверніть увагу, що 0 ділиться на будь-яке число, а отже, якщо на столі лишається 0 камінців, то умова задачі виконана.

Формат вхідних даних

Єдиний рядок містить єдине ціле число n $(0 \leqslant n \leqslant 10^9)$ — початкова кількість камінців на столі.

Формат вихідних даних

Виведіть єдине число — мінімальну кількість монет яку треба витратити, щоб число камінців на столі почало ділитись націло на 5.

Приклади

standard input	standard output
0	0
1	1
3	2
228	2
300	0
2021	1

Зауваження

В першому прикладі на столі спочатку лежить 0 камінців. 0 ділиться на 5, тому не потрібно витрачати жодної монети.

В другому прикладі можна заплатити одну монету і забрати один камінець зі столу. Тоді на столі опиниться 0 камінців, а 0 ділиться на 5.

В третьому прикладі можна заплатити одну монету і покласти ще один камінець на стіл (таким чином, на столі буде 4 камінці), а потім заплатити ще одну монету і покласти ще один камінець на стіл, отримуючи таким чином 5 камінців, що ділиться на 5.

Задача B. Медіанне божевілля

Ліміт часу: 3 seconds
Ліміт використання пам'яті: 256 megabytes

Це інтерактивна задача.

Загадана деяка перестановка p чисел від 1 до n, де n **парне**. Ви можете задавати запити наступного вигляду:

- Для даного набору різних індексів непарної довжини $a_1, a_2, \ldots, a_l$, ви можете дізнатись таке число x, що p_x є медіаною елементів $p_{a_1}, p_{a_2}, \ldots, p_{a_l}$.

Відомо, що $p_1 < p_n$. Вгадайте перестановку p за **не більше ніж** $\frac{3n}{2}$ запитів.

Гарантується, що перестановка зафіксована перед початком взаємодії. Іншими словами, **інтерактор не адаптивний**.

Нагадаємо, що медіана непарної кількості чисел визначається наступним чином:

Нехай $b_1, b_2, \ldots, b_{2k+1}$ — це ці числа в порядку зростання (де $2k+1$ — кількість чисел). Тоді медіаною є число b_{k+1}.

Протокол взаємодії

Почніть взаємодію, зчитавши одне ціле число n ($2 \leqslant n \leqslant 1000$, n парне) — довжину перестановки.

Щоб задати питання, необхідно вивести в одному рядку спочатку символ «`?`», потім ціле число l, а потім l цілих чисел a_i ($1 \leqslant l \leqslant n$, $1 \leqslant a_i \leqslant n$, l непарне, всі a_i попарно різні) — індекси, для яких необхідно знайти медіану.

У відповідь програма журі виведе таке число x, що p_x є медіаною елементів $p_{a_1}, p_{a_2}, \ldots, p_{a_l}$.

Коли ви визначили перестановку, то виведіть спочатку символ «`!`», а потім n чисел $p_1, p_2, \ldots, p_n$. Після цього ваша програма має завершити роботу.

Після кожного запиту і виводу відповіді не забудьте вивести перехід рядка і скинути буфер виводу. Для скидання буферу використовуйте:

- `fflush(stdout)` чи `cout.flush()` в C++;
- `System.out.flush()` в Java;
- `flush(output)` в Pascal;
- `stdout.flush()` в Python;

Приклад

standard input	standard output
4	? 3 2 3 4
2	? 3 1 3 4
3	? 3 1 2 4
2	? 3 1 2 3
3	! 1 3 2 4

Зауваження

В прикладі загадана перестановка $p = (1, 3, 2, 4)$.

Перший запит в прикладі — хочемо дізнатись медіану елементів p_2, p_3, p_4, що дорівнюють $3, 2, 4$ відповідно. Медіана цих чисел — 3, тобто p_2, тому інтерактор відповідає числом 2.

Другий запит в прикладі — хочемо дізнатись медіану елементів p_1, p_3, p_4, що дорівнюють $1, 2, 4$ відповідно. Медіана цих чисел — 2, тобто p_3, тому інтерактор відповідає числом 3.

Третій запит в прикладі — хочемо дізнатись медіану елементів p_1, p_2, p_4, що дорівнюють $1, 3, 4$ відповідно. Медіана цих чисел — 3, тобто p_2, тому інтерактор відповідає числом 2.

Четвертий запит в прикладі — хочемо дізнатись медіану елементів p_1, p_2, p_3, що дорівнюють $1, 3, 2$ відповідно. Медіана цих чисел — 2, тобто p_3, тому інтерактор відповідає числом 3.

Задача C. Абсолютно неадекватні операції

Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Дано масив з n цілих чисел $[a_1, a_2, \ldots, a_n]$. За одну операцію можна зробити наступне:

- Вибрати деяке i, для якого $2 \leqslant i \leqslant n-1$.
- Нехай $a_i = x$. Тоді ми додаємо x до a_{i-1} та a_{i+1}, а a_i замінюємо на $-x$.

Наприклад, якщо масив мав вигляд $[1, 2, -1, 3]$, то ми можемо обрати $i = 3$, виконати операцію, і отримати масив $[1, 1, 1, 2]$.

Ви можете застосовувати дану операцію до масиву довільну кількість разів. Скільки різних масивів ви можете отримати? Якщо ви можете отримати нескінченну кількість масивів, виведіть -1, інакше виведіть число цих масивів за модулем $10^9 + 7$.

Два масиви вважаються різними, якщо вони відрізняються принаймні в одній позиції.

Формат вхідних даних

Перший рядок вхідних даних містить єдине ціле число n ($3 \leqslant n \leqslant 10^5$) — довжину масиву.

Другий рядок містить n цілих чисел $a_1, a_2, \ldots, a_n$ ($-10^9 \leqslant a_i \leqslant 10^9$) — елементи масиву.

Формат вихідних даних

Якщо ви можете отримати нескінченну кількість масивів, виведіть -1, інакше виведіть число цих масивів за модулем $10^9 + 7$.

Приклади

standard input	standard output
3 18 9 2021	2
5 0 0 0 0 0	1
6 1 -1 1 -1 1 -1	10

Зауваження

В першому прикладі, ми можемо застосувати операцію лише для $i = 2$, що дає два різні масиви: $[18, 9, 2021]$ та $[27, -9, 2030]$.

В другому прикладі, яку б операцію ми не застосували, масив лишатиметься рівним $[0, 0, 0, 0, 0]$.

Задача D. Паліндромна лихоманка

Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Вам дано рядок s з латинських маленьких літер. Знайдіть довільний рядок t з латинських маленьких літер довжини не більше $2 \cdot 10^5$, для якого:

- Рядок $s + t$ є паліндромом.
- Рядок $t + s$ є паліндромом.

Можна показати, що при обмеженнях даної задачі такий рядок завжди знайдеться.

Нагадаємо, що паліндром — це рядок, що зліва направо читається так само як справа наліво. Наприклад, `abcba` є паліндромом, а `oppagangnamstyle` ні.

Нагадаємо, що позначення $s + t$ позначає конкатенацію рядків s та t. Наприклад, `energynot` + `over` = `energynotover`.

Формат вхідних даних

Єдиний рядок вхідних даних містить рядок s з латинських маленьких літер довжини не більше 10^5.

Формат вихідних даних

Виведіть довільний **непустий** рядок t з латинських маленьких літер довжини не більше $2 \cdot 10^5$, що задовольняє умовам задачі.

Зверніть увагу, що t не може бути пустим, навіть якщо пустий рядок задовольняє умові задачі.

Приклади

standard input	standard output
abbaabba	abba
zyzz	zzyz

Зауваження

В першому прикладі, $s + t = t + s =$ `abbaabbaabba`, що є паліндромом.

В другому прикладі, $s + t =$ `zyzzzzyz`, а $t + s =$ `zzyzzyzz`. Обидва слова є паліндромами.

Задача E. Шлях додому

Ліміт часу: 8 seconds
Ліміт використання пам'яті: 256 megabytes

n міст розташовані на прямій в порядку $1, 2, \ldots, n$, відстань між містами i та $i+1$ рівна D_i для кожного i від 1 до $n-1$.

Вам потрібно порахувати кількість маршрутів, для яких виконуються наступні умови:

- В маршруті рівно k міст
- Всі міста в маршруті попарно різні
- Сумарна довжина маршруту ділиться на m.

Сумарна довжина маршруту дорівнює сумі відстаней між містами, що йдуть в маршруті підряд. Відстань між містами i, j визначається за такою формулою:

- $D_i + \ldots + D_{j-1}$, якщо $i < j$.
- $D_j + \ldots + D_{i-1}$, якщо $j < i$.

Наприклад, якщо $n = 4$ і $D = [3, 5, 7]$, то довжина маршруту $[3, 1, 4]$ дорівнює $8 + 15 = 23$.

Виведіть кількість маршрутів за модулем $10^9 + 7$.

Формат вхідних даних

У першому рядку містяться три цілі числа $n, m, k (2 \leqslant n \leqslant 80, 1 \leqslant m \leqslant 80, 2 \leqslant k \leqslant n)$ — кількість міст на прямій, число, на яке повинна ділитися сумарна довжина маршруту, і кількість міст в маршруті.

У наступному рядку містяться $n-1$ цілих чисел $D_1, D_2, \ldots, D_{n-1}$ $(1 \leqslant D_i \leqslant m)$ — відстані між сусідніми містами.

Формат вихідних даних

Виведіть кількість маршрутів за модулем $10^9 + 7$.

Приклади

standard input	standard output
4 5 3 1 2 3	4
15 17 6 1 2 3 4 5 6 7 8 9 10 11 12 13 14	210690

Зауваження

В першому прикладі існує 4 такі маршрути:

- $2 \to 3 \to 4$: довжина рівна $2 + 3 = 5$.
- $4 \to 3 \to 2$: довжина рівна $3 + 2 = 5$.
- $1 \to 3 \to 2$: довжина рівна $(1 + 2) + 2 = 5$.
- $2 \to 3 \to 1$: довжина рівна $2 + (1 + 2) = 5$.

Задача F. ПерестановочкА

Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Для перестановки p чисел від 1 до n, визначимо $f(p)$ наступним чином: для кожної пари чисел (i, j) з $1 \leqslant i \leqslant j \leqslant n$, порахуємо пару (min, max), де min — найменше число серед чисел $a_i, a_{i+1}, \ldots, a_j$, а max — найбільше з них. Тоді $f(p)$ рівна кількості різних пар серед всіх $\frac{n(n+1)}{2}$ пар.

Наприклад розглянемо перестановку $(1, 3, 2)$.

- Для пари $(1, 1)$, $(min, max) = (1, 1)$
- Для пари $(1, 2)$, $(min, max) = (1, 3)$
- Для пари $(1, 3)$, $(min, max) = (1, 3)$
- Для пари $(2, 2)$, $(min, max) = (3, 3)$
- Для пари $(2, 3)$, $(min, max) = (2, 3)$
- Для пари $(3, 3)$, $(min, max) = (2, 2)$

Всього 5 різних пар, тому $f((1, 3, 2)) = 5$.

Знайдіть $f(p)$ для даної вам перестановки $(p_1, p_2, \ldots, p_n)$.

Формат вхідних даних

Перший рядок вхідних даних містить єдине число n $(1 \leqslant n \leqslant 2 \cdot 10^5)$.

Другий рядок вхідних даних містить n цілих чисел $p_1, p_2, \ldots, p_n$ ($1 \leqslant p_i \leqslant n$, p_i **попарно різні**) — перестановку довжини n.

Формат вихідних даних

Виведіть $f(p)$.

Приклади

standard input	standard output
3 1 3 2	5
8 1 2 3 4 5 6 7 8	36
8 1 8 2 7 3 6 4 5	15

Зауваження

Перший приклад розібрано в умові.

Задача G. ПерестановочкИ

Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Для перестановки p чисел від 1 до n, визначимо $f(p)$ наступним чином: для кожної пари чисел (i, j) з $1 \leqslant i \leqslant j \leqslant n$, порахуємо пару (min, max), де min — найменше число серед чисел $a_i, a_{i+1}, \ldots, a_j$, а max — найбільше з них. Тоді $f(p)$ рівна кількості різних пар серед всіх $\frac{n(n+1)}{2}$ пар.

Наприклад розглянемо перестановку $(1, 3, 2)$.

- Для пари $(1, 1)$, $(min, max) = (1, 1)$
- Для пари $(1, 2)$, $(min, max) = (1, 3)$
- Для пари $(1, 3)$, $(min, max) = (1, 3)$
- Для пари $(2, 2)$, $(min, max) = (3, 3)$
- Для пари $(2, 3)$, $(min, max) = (2, 3)$
- Для пари $(3, 3)$, $(min, max) = (2, 2)$

Всього 5 різних пар, тому $f((1, 3, 2)) = 5$.

Знайдіть суму $f(p)$ по всім перестановкам p довжини n, за модулем $10^9 + 7$.

Формат вхідних даних

Єдиний рядок вхідних даних містить єдине число n $(1 \leqslant n \leqslant 2 \cdot 10^5)$.

Формат вихідних даних

Виведіть суму $f(p)$ по всім перестановкам p довжини n, за модулем $10^9 + 7$.

Приклади

standard input	standard output
1	1
2	6
3	32
228	384127128

Задача H. Вишуканий максимум

Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Дано n **попарно різних чисел** $a_1, a_2, \ldots, a_n$. Знайдіть максимальне можливе значення виразу $\frac{a_i a_j}{|a_i - a_j|}$ по $1 \leqslant i < j \leqslant n$.

Формат вхідних даних

Перший рядок містить єдине ціле число n $(2 \leqslant n \leqslant 2 \cdot 10^5)$ — кількість чисел.

Другий рядок містить n **попарно різних** цілих чисел $a_1, a_2, \ldots, a_n$ $(1 \leqslant a_i \leqslant 10^9)$.

Формат вихідних даних

Виведіть єдине число — максимальне можливе значення виразу $\frac{a_i a_j}{|a_i - a_j|}$ по $1 \leqslant i < j \leqslant n$.

Ваш відповідь буде вважатися правильною, якщо її абсолютна або відносна помилка не перевищує 10^{-6}.

Формально, нехай ваш відповідь дорівнює a, а відповідь журі дорівнює b. Ваша відповідь буде зарахована, якщо і тільки якщо $\frac{|a-b|}{\max(1, |b|)} \leqslant 10^{-6}$.

Приклад

standard input	standard output
3 10 3 7	23.3333333333

Зауваження

В прикладі, $\frac{3 \cdot 7}{4} = 5.25$, $\frac{3 \cdot 10}{7} = 4.2857\ldots$, $\frac{7 \cdot 10}{3} = 23.3333\ldots$.

Задача I. Надзвичайно оригінальна задача про кістякове дерево

Ліміт часу: 3 seconds
Ліміт використання пам'яті: 256 megabytes

В Україні n міст і 0 доріг. До 30-ої річниці незалежності пора б це виправити.

Ви хочете побудувати $n-1$ дорогу між містами таким чином, щоб ними з кожного міста можна було дістатись до будь-якого іншого. Вартість прокладання дороги між містами i та j рівна $(a_i + a_j) \bmod M$ мільйонів гривень з бюджетних коштів, де M — улюблене число чинного Президента України.

Яку найменшу кількість мільйонів гривень з бюджетних коштів потрібно витратити для виконання цього плану?

Формат вхідних даних

Перший рядок містить два цілих числа n, M $(1 \leqslant n \leqslant 200000, 1 \leqslant M \leqslant 10^9)$ — число міст та улюблене число чинного Президента України відповідно.

Другий рядок містить n цілих чисел $a_1, a_2, \ldots, a_n$ $(0 \leqslant a_i < M)$.

Формат вихідних даних

Виведіть найменшу кількість мільйонів гривень з бюджетних коштів, які потрібно витратити, щоб побудувати $n-1$ дорогу, щоб ними з кожного міста можна було дістатись до будь-якого іншого.

Приклади

standard input	standard output
3 350 42 69 300	130
12 12 3 7 5 2 9 7 6 7 8 7 2 1	14
6 11 3 2 2 2 2 8	17
1 998244353 7788	0

Зауваження

В першому прикладі ми можемо прокласти 3 дороги з наступними вартостями:

- Між містами 1, 2: $(42 + 69) \bmod 350 = 111$
- Між містами 1, 3: $(42 + 300) \bmod 350 = 342$
- Між містами 2, 3: $(69 + 300) \bmod 350 = 19$

Найвигідніше обрати дороги $(1, 2)$ та $(2, 3)$, з сумарною вартістю $111 + 19 = 130$.

Задача J. Коли немає чим зайнятись, а вдома лише перестановка

Ліміт часу: 2 seconds
Ліміт використання пам'яті: 256 megabytes

Дана перестановка $(p_1, p_2, \ldots, p_n)$ чисел від 1 до n.
З нею ви можете робити наступну операцію:

- Ви можете переставити місцями два **сусідні** елементи p. Ця операція займає рівно одну секунду.

За одну ітерацію ви робите наступне:

- Розглянемо перестановку $(q_1, q_2, \ldots, q_n)$, що знаходиться **прямо перед** $(p_1, p_2, \ldots, p_n)$ в лексикографічному порядку, тобто q лексикографічно менша за p і не існує жодної перестановки "між ними". Наприклад, для $p = (2, 3, 1)$ $q = (2, 1, 3)$, для $p = (3, 1, 2, 4)$ $q = (2, 4, 3, 1)$, а для $p = (1, 4, 2, 3)$ $q = (1, 3, 4, 2)$.
- Перетворимо перестановку p в перестановку q, за мінімально можливу кількість операцій. Наприклад, щоб перетворити $(1, 4, 2, 3)$ в $(1, 3, 4, 2)$, потрібно мінімум 2 операції: $(1, 4, 2, 3) \to (1, 4, 3, 2) \to (1, 3, 4, 2)$.

Ви застосовуєте до отриманої перестановки p дану ітерацію, поки вона не стане тотожною перестановкою (тобто рівною $(1, 2, 3, \ldots, n)$). Скільки часу це займе? Оскільки це число може бути дуже великим, виведіть його за модулем $10^9 + 7$.

Перестановка p вважається лексикографічно меншою за перестановку q, якщо існує такий індекс i, що $p_j = q_j$ для всіх $j < i$, а також $p_i < q_i$.

Формат вхідних даних

Перший рядок містить єдине число n ($1 \leqslant n \leqslant 2 \cdot 10^5$)— довжину перестановки.

Другий рядок містить n цілих чисел $p_1, p_2, \ldots, p_n$ ($1 \leqslant p_i \leqslant n$, всі числа попарно різні) — елементи перестановки.

Формат вихідних даних

Виведіть кількість секунд, яка пройде, поки процес не завершиться, за модулем $10^9 + 7$.

Приклади

standard input	standard output
4 1 4 2 3	6
6 6 5 4 3 2 1	1423

Зауваження

В першому прикладі маємо:
$(1, 4, 2, 3) \to (1, 4, 3, 2) \to (1, 3, 4, 2)$: 2 секунди.
$(1, 3, 4, 2) \to (1, 3, 2, 4)$: 1 секунда.
$(1, 3, 2, 4) \to (1, 2, 3, 4) \to (1, 2, 4, 3)$: 2 секунди.
$(1, 2, 4, 3) \to (1, 2, 3, 4)$: 1 секунда.
Всього 6 секунд.

Задача K. Чергове розчарування: задача на парування

Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Дано непарне число n. Для масиву з n чисел $[a_1, a_2, \ldots, a_n]$, будемо визначати його **вартість**, як максимальну вагу досконалого парування в графі на $n+1$ вершинах, в якому вага ребра (i, j) для $i < j$ визначається як $max(a_i, a_{i+1}, \ldots, a_{j-1})$.

Наприклад, для масиву $[1, 30, 15]$, ми маємо граф на 4 вершинах з наступними відстанями між ними:

- $d(1, 2) = 1$
- $d(1, 3) = d(1, 4) = d(2, 3) = d(2, 4) = 30$
- $d(3, 4) = 15$

Тут парування $((1, 2), (3, 4))$ має 16, а $((1, 3), (2, 4))$ і $((1, 4), (2, 3))$ мають ваги 60, тому вартість всього масиву рівна 60.

Вам дано масив b довжини n з **попарно різних** елементів. Знайдіть суму вартостей всіх перестановок цього масиву, за модулем $10^9 + 7$.

Нагадаємо, що досконале парування в зваженому графі на $2k$ вершинах — це набір з k ребер такий, що кожна вершина є кінцем рівно одного ребра. Вагою досконалого парування вважається сума ваг всіх вибраних k ребер.

Формат вхідних даних

Перший рядок містить єдине ціле число n ($1 \leqslant n \leqslant 99\,999$, n **непарне**) — довжину масиву.

Другий рядок містить n **попарно різних** цілих чисел $b_1, b_2, \ldots, b_n$ ($1 \leqslant b_i \leqslant 10^8$) — елементи масиву.

Формат вихідних даних

Виведіть суму вартостей всіх перестановок цього масиву, за модулем $10^9 + 7$.

Приклади

standard input	standard output
1 300	300
3 1 30 15	300
5 42 69 228 1488 2021	605448

Зауваження

В прикладі:

- Вартості масивів $[1, 30, 15]$ та $[15, 30, 1]$ рівні 60.
- Вартості масивів $[1, 15, 30]$, $[30, 15, 1]$, $[15, 1, 30]$, $[30, 1, 15]$ рівні 45.

Сума по всім перестановкам рівна $60 \cdot 2 + 45 \cdot 4 = 300$.

Задача L. Задача на кореневу декомпозицію з великими обмеженнями

Ліміт часу: 4 seconds
Ліміт використання пам'яті: 1024 megabytes

Дано натуральне число n. Знайдіть, скільки серед чисел $n \bmod 1$, $n \bmod 2$, $\ldots$, $n \bmod n$ різних чисел.

Формат вхідних даних

Перший і єдиний рядок містить одне ціле число n ($1 \leqslant n \leqslant 10^{12}$).

Формат вихідних даних

Виведіть єдине число — кількість різних чисел серед $n \bmod 1$, $n \bmod 2$, $\ldots$, $n \bmod n$.

Приклади

standard input	standard output
1	1
2	1
3	2

Зауваження

В першому прикладі, ми розглядаємо всього одне число: $1 \bmod 1 = 0$, тому відповідь 1.

В другому прикладі, маємо $2 \bmod 1 = 2 \bmod 2 = 0$, тому відповідь знову 1.

В третьому прикладі, маємо $3 \bmod 1 = 3 \bmod 3 = 0$, а також $3 \bmod 2 = 1$, всього два різні числа.