

Задача А. Де конструкція

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Вам задано три цілих числа $0 \leq k \leq m < n$. Побудуйте двійковий рядок s довжиною n (s повинен складатися лише з символів 0 та 1), такий, що в кожному підрядку s довжини m є не більше k одиниць, а також в кожному підрядку s довжини $m + 1$ є принаймні k одиниць.

Можна показати, що при обмеженнях задачі такий рядок завжди існує.

Формат вхідних даних

У єдиному рядку вхідних даних містяться три цілих числа n , m та k ($0 \leq k \leq m < n \leq 100\,000$, $0 < m$).

Формат вихідних даних

Виведіть двійковий рядок довжиною n , що задовольняє умові задачі.

Якщо є багато розв'язків, ви можете вивести будь-який з них.

Приклади

standard input	standard output
4 2 1	0100
5 4 4	11111

Зауваження

В першому тесті, кожен підрядок довжини 2 містить не більше 1 одиниці, а кожен підрядок довжини 3 містить принаймні одну одиницю.

Задача В. Легенду вкрали

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 7 seconds
Ліміт використання пам'яті: 256 megabytes

А ви шо, очікували тут на легенду? Хех, не задрю вам...

У єдиному рядку задано ціле число n .

Вам потрібно знайти та вивести наступне значення:

$$\sum_{i=1}^n \left\lfloor \frac{n \cdot \gcd(n, i)}{i} \right\rfloor$$

Формат вхідних даних

У єдиному рядку задано ціле число n , $1 \leq n \leq 10^{11}$.

Формат вихідних даних

У єдиному рядку виведіть відповідь на задачу.

Приклади

standard input	standard output
50	544
1000000	183545531
6469693230	36120890374011

Зауваження

Легенди не буде, ви ж це зрозуміли?

Буде пояснення, що таке це ваше \gcd , $\gcd(a, b)$ позначає найбільший спільний дільник двох натуральних чисел a, b . Наприклад, $\gcd(6, 9) = 3$.

Задача С. Аеропорти та авіакомпанії

Назва вхідного файлу:	standard input
Назва вихідного файлу:	standard output
Ліміт часу:	1 second
Ліміт використання пам'яті:	256 megabytes

У великій країні є n аеропортів (пронумерованих від 1 до n), між якими працюють m авіакомпаній (пронумерованих від 1 до m). Кожен аеропорт з'єднаний безпосереднім двонаправленим рейсом з кожним іншим аеропортом, і кожен такий рейс обслуговується лише однією авіакомпанією. Щоб забезпечити зручність для пасажирів і збалансованість маршрутів, аеропорти вимагають, щоб кожен аеропорт мав однакову кількість рейсів від кожної авіакомпанії, а саме k рейсів, де $k = \frac{n-1}{m}$ (гарантується, що k ціле). Ваше завдання — спланувати рейси між аеропортами таким чином, або встановити, що задану умову неможливо виконати.

Формат вхідних даних

В єдиному рядку вхідних даних записані два числа n та m ($2 \leq n \leq 1000$; $1 \leq m \leq n-1$) — кількість аеропортів та кількість авіакомпаній. Гарантується, що $n-1$ ділиться націло на m .

Формат вихідних даних

Якщо задану умову виконати неможливо, виведіть рядок «NO» (без лапок).

Інакше виведіть в першому рядку «YES» (без лапок), а далі таблицю $n \times n$ з цілих чисел від 0 до m . На перетині i -го рядка та j -го стовпчика має знаходитись номер авіакомпанії (від 1 до m), яка обслуговує рейс між аеропортами i та j . Рейси двонаправлені, тому на перетині i -го рядка та j -го стовпчика має знаходитись те саме число, що й на перетині j -го рядка та i -го стовпчика. На діагоналі таблиці мають знаходитись нулі, оскільки не існує рейсів, обидва кінці яких знаходяться в одному аеропорту.

Приклади

standard input	standard output
3 1	YES 0 1 1 1 0 1 1 1 0
3 2	NO
4 3	YES 0 1 2 3 1 0 3 2 2 3 0 1 3 2 1 0
5 2	YES 0 1 2 2 1 1 0 1 2 2 2 1 0 1 2 2 2 1 0 1 1 2 2 1 0

Задача D. Мінімальне матсподівання

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Для двійкового рядка s та масиву дійсних чисел a однакової довжини n визначимо функцію $f(s, a) = (\sum_{i:s_i=0} a_i - \sum_{i:s_i=1} a_i)^2$.

Наприклад, для рядка $s = 0110100$ та масиву $a = [a_1, a_2, a_3, a_4, a_5, a_6, a_7]$ маємо $f(s, a) = ((a_1 + a_4 + a_5 + a_6) - (a_2 + a_3 + a_7))^2$.

Нехай $g(a)$ — це посортований за спаданням масив a .

Наприклад, $g([0.4, 0.74, 0.47, 0.7]) = [0.74, 0.7, 0.47, 0.4]$.

Розглянемо також інший приклад:

$f(0110, g([0.4, 0.74, 0.47, 0.7])) = f(0110, [0.74, 0.7, 0.47, 0.4]) = ((0.74 + 0.4) - (0.7 + 0.47))^2$.

Нехай $x^{(n)}$ — випадковий масив, кожна елемент якого незалежний і рівномірно розподілений на відрізок $[0, 1]$.

Нехай $h(s)$ для двійкового рядка s — це математичне сподівання випадкової змінної $f(s, g(x^{(n)}))$.

Задано ціле число n .

Знайдіть мінімально можливе значення $h(s)$ для двійкового рядка s довжини n . Нехай цей мінімум дорівнює m . Виведіть $m \cdot (n + 1)(n + 2)$. Можна довести, що це число буде цілим.

Також знайдіть кількість двійкових рядків s довжини n , для яких виконується $h(s) = m$. Виведіть цю кількість за модулем простого числа 998244353.

Формат вхідних даних

В єдиному рядку задано ціле число n ($1 \leq n \leq 10^6$).

Формат вихідних даних

В одному рядку виведіть два цілих числа. Перше число: m , помножене на $(n + 1)(n + 2)$. Друге число: кількість двійкових рядків, для яких досягається мінімум m , за модулем 998244353.

Приклади

standard input	standard output
2	2 2
5	4 6

Зауваження

В першому прикладі можна показати, що найменшим це математичне сподівання буде для рядків 01 та 10.

Задача Е. Різнобарвні трикутники

Назва вхідного файлу:	standard input
Назва вихідного файлу:	standard output
Ліміт часу:	2 seconds
Ліміт використання пам'яті:	256 megabytes

Олексій любить трикутники за те, що вони мають по три сторони. Він також любить вирішувати задачі, пов'язані з трикутниками.

У Олексія є багато паличок, розфарбованих у k різних кольорів, позначених числами від 1 до k . Олексій хоче скласти **невироджений** трикутник з трьох паличок, які мають **попарно різні** кольори.

Допоможіть Олексію здійснити свою мрію.

Трикутник називається *невиродженим*, якщо його площа додатня.

Формат вхідних даних

Перший рядок містить ціле число k ($3 \leq k \leq 50$) — кількість кольорів.

У наступних k рядках містяться описи паличок. У i -му з цих рядків спочатку записано ціле додатнє число c_i — кількість паличок кольору i . Далі йдуть c_i цілих додатних чисел — довжини паличок кольору i . Довжини паличок не перевищують 10^9 . Загальна кількість паличок по всіх кольорах не перевищує 10^6 .

Формат вихідних даних

Якщо шуканий трикутник скласти неможливо, виведіть число -1 .

У протилежному випадку виведіть шість цілих чисел, які описують трикутник так: колір і довжина першої палички; колір і довжина другої; колір і довжина третьої палички.

Якщо можливих неvirоджених трикутників декілька, виведіть будь-який з них.

Приклади

standard input	standard output
4 1 98 2 6 9 3 8 4 8 2 15 2022	2 9 3 8 4 15
3 1 10 1 20 1 30	-1

Зауваження

В першому тесті ми змогли скласти різнобарвний трикутник зі сторонами 9, 8, 15.

Задача F. Нормальне вгадування

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Це інтерактивна задача.

Є n гирьок, пронумерованих від 1 до n . i -а гирька має вагу a_i .

Є терези з двома шальками. Терези дають похибку, що має нормальний розподіл із середнім значенням 0 і стандартним відхиленням σ .

За один запит можна поставити якісь гирьки на ліву шальку, якісь — на праву. Терези покажуть різницю ваг на лівій і правій шальках (ліва мінус права) з похибкою.

Дозволено зробити не більше як $4n$ запитів. Потрібно знайти вагу кожної гирьки з похибкою не більшою за $\frac{7\sigma}{\sqrt{n}}$.

Формат вхідних даних

$1 \leq n \leq 200$,
 $1 \leq \sigma \leq 10$,
 $100 \leq a_i \leq 1000$.

Протокол взаємодії

Почніть взаємодію, зчитавши ціле число n і дійсне число σ в першому рядку.

Щоб задати запит, виведіть його у форматі

? m_1 m_2

x_1 x_2 ... x_{m_1}

y_1 y_2 ... y_{m_2}

Тут m_1 — кількість гирьок на лівій шальці, m_2 — на правій, x_i — індекси гирьок на лівій шальці, y_i — на правій. Всі x_i, y_i мають бути попарно різними.

У відповідь на запит, ви отримаєте число: результат показів терезів.

Коли ви визначили ваги гирьок, виведіть їх в форматі

! a_1 a_2 ... a_n .

Ваша відповідь буде зарахована, якщо вага кожної гирька відхиляється від справжньої не більше, ніж на $\frac{7\sigma}{\sqrt{n}}$.

Приклад

standard input	standard output
2 4.5	? 2 0
314.15 271.82	1 2
	590.123
	? 1 1
	1
	2
	34.845
	! 312.484 277.639

Задача G. Поверніть всіх на місце

Назва вхідного файлу:	standard input
Назва вихідного файлу:	standard output
Ліміт часу:	1 second
Ліміт використання пам'яті:	256 megabytes

Дано перестановку p_1, p_2, \dots, p_n чисел від 1 до n . Ви можете застосовувати до неї наступну операцію:

- Виберіть якесь $k \leq \frac{n}{2}$. Після цього, виберіть k пар позицій (a_i, b_i) , так щоб кожна позиція була вибрана максимум один раз (формально, $1 \leq a_i, b_i \leq n$, всі $2k$ чисел a_i, b_i попарно різні). Після цього, для кожного i від 1 до k , обміняйте місцями числа p_{a_i}, p_{b_i} .

Ви хочете отримати з початкової перестановки перестановку $(1, 2, \dots, n)$. Яку найменшу кількість операцій вище для цього необхідно застосувати? Також, виведіть приклад.

Можна показати, що це завжди можливо за не більше ніж n операцій.

Формат вхідних даних

Перший рядок містить єдине ціле число t ($1 \leq t \leq 1000$) — кількість наборів вхідних даних. Далі слідує опис наборів вхідних даних.

Перший рядок кожного набору вхідних даних містить одне ціле число n ($1 \leq n \leq 1000$) — довжину перестановки.

Другий рядок кожного набору вхідних даних містить n цілих чисел p_1, p_2, \dots, p_n ($1 \leq p_i \leq n$, всі p_i попарно різні) — елементи перестановки.

Гарантується, що сума n по всім наборам вхідних даних не перевищує 1000.

Формат вихідних даних

Для кожного набору вхідних даних, в першому рядку виведіть одне ціле число m ($0 \leq m \leq n$) — мінімальну кількість операцій, яку необхідно застосувати, щоб отримати з початкової перестановки перестановку $(1, 2, \dots, n)$. Далі виведіть описи m операцій.

Для кожної операції, спочатку виведіть одне ціле число k ($1 \leq k \leq \frac{n}{2}$) — кількість вибраних вами пар. В i -му з наступних k рядків виведіть два числа a_i, b_i ($1 \leq a_i, b_i \leq n$, всі $2k$ чисел a_i, b_i попарно різні).

Приклад

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

Задача Н. Xor-Sum

Назва вхідного файлу:	standard input
Назва вихідного файлу:	standard output
Ліміт часу:	5 seconds
Ліміт використання пам'яті:	512 megabytes

Скоро наближаються міжгалактичні змагання з CodeBall-y — I-Code-PC. Особливістю цих змагань є те, що в них можуть брати участь команди з довільною кількістю спортсменів, бо організатори навчилися поєднувати результати між командами з різною кількістю учасників за ідеально-збалансованою формулою.

Кращому тренеру міжнародного класу, Олександрові Володимировичу Вечуру належить вибрати команду-представника для участі в майбутньому змаганні.

Для цього тренер визначив числову характеристику A_i - показник спортсмена i , який позначає вміння писати код, а також створив формули, що визначають показники команди T :

- показник швидкості написання коду в команді $S = A_1 \oplus A_2 \oplus \dots \oplus A_n$, де \oplus позначає операцію побітового виняткового АБО
- показник якості (чистоти) написання коду в команді $Q = \sum_{i \in T} A_i$

Як відомо, якщо команда пише код швидше, ніж встигає його налагоджувати та перевіряти, припускаючись при цьому безлічі помилок, то це свідчить про її незбалансованість, що призводить до сповільнення команди та додавання їй штрафних очок. Тому в хорошій команді, на думку тренера, показник якості коду має **стро́го** перевищувати показник швидкості коду, тобто $Q > S$.

Крім цього тренування спортсменів проходило за особливою методологією, яка свідчила, що в команді з m учасників існує m ролей, кожна з яких важлива.

Допоможіть тренеру, визначивши скільки різних команд він може скласти з n учасників. Оскільки ця кількість може бути дуже великою, виведіть її за модулем 998244353.

Формат вхідних даних

У першому рядку задане ціле число n ($1 \leq n \leq 10^6$) — кількість спортсменів.

У другому рядку задано n цілих чисел — A_1, \dots, A_n ($1 \leq A_i \leq 10^6$), де A_i позначає показник вміння писати код у спортсмена i .

Формат вихідних даних

У єдиному рядку виведіть одне ціле число — кількість способів вибрати команду, яка задовольняє умовами тренера, за модулем 998244353.

2 способи скласти команду вважаються різними якщо кількість учасників у команді різна, або існує роль в команді, за якою закріплені різні учасники.

Приклади

standard input	standard output
1 3	0
4 1 2 3 4	46
4 1 2 4 4	38

Зауваження

Розглянемо другий приклад:

Підходящі команди:

- $\{1, 3\}$ (2 способи з урахуванням ролей)

- $\{2, 3\}$ (2 способи з урахуванням ролей)
- $\{1, 2, 3\}$ (6 способів з урахуванням ролей)
- $\{1, 3, 4\}$ (6 способів з урахуванням ролей)
- $\{2, 3, 4\}$ (6 способів з урахуванням ролей)
- $\{1, 2, 3, 4\}$ (24 способи з урахуванням ролей)

Всього $24 + 6 + 6 + 6 + 2 + 2 = 46$.

Задача I. Гра вгадування

Назва вхідного файлу: *standard input*
Назва вихідного файлу: *standard output*
Ліміт часу: 2 seconds
Ліміт використання пам'яті: 1024 mebibytes

Це інтерактивна задача.

Ваше завдання — написати програму, яка вгадує секретне число шляхом повторних запитів та відповідей. Секретне число — невід'ємне ціле число менше 10^{18} .

Позначимо x як секретне число. У запиті ви вказуєте невід'ємне ціле число a . У відповідь на нього ви отримаєте суму цифр числа $(x + a)$. Тут сума цифр числа означає суму всіх цифр у його десятковому записі. Наприклад, сума цифр числа 4096 дорівнює $4 + 0 + 9 + 6 = 19$.

Ви можете задати не більше 75 запитів.

Протокол взаємодії

Щоб надіслати запит про число a , використовуйте наступний формат:

`query a`

Тут a має бути цілим числом від 0 до $10^{18} - 1$, включно. У відповідь на цей запит, ви отримаєте суму цифр числа $(x + a)$, де x — секретне число.

Якщо ви впевнені, що загадане число — y , використовуйте наступний формат:

`answer y`

Тут y — секретне число, яке ви визначили, ціле число від 0 до $10^{18} - 1$, включно. Ви можете відправити відповідь лише один раз. Після відправки відповіді ваша програма повинна завершитися.

Приклад

<i>standard input</i>	<i>standard output</i>
15	query 3
1	query 25
	answer 75

Зауваження

Якщо ваш вивід порушує будь-які з умов вище (неправильний формат, a або y виходять за межі діапазону, більше 75 запитів, додатковий вивід після відправки відповіді і т.д.), ваше рішення буде визнано неправильним. Пам'ятайте, що деякі середовища вимагають очищення буферів виводу.

Задача J. Логістика XOR-ляндії

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: `7 seconds`
Ліміт використання пам'яті: `256 megabytes`

У XOR-ляндії є n міст. i -е місто знаходиться за координатами v_i . Уряд XOR-ляндії планує модернізувати транспортну систему, побудувавши високоякісні двосторонні дороги між деякими парами міст.

Побудування дороги між містами v_i and v_j коштує $XOR(v_i, v_j)$ монет. Тут XOR , або \oplus , позначає операцію побітового виняткового АБО. **Дорога не проходить через жодні міста, окрім цих двох.**

Нажаль, уряд має обмежену кількість коштів, тому він розглядає можливість модернізувати транспорт лише між певною підмножиною міст із індексами між l і r . Після модернізації повинен існувати шлях між будь-якими парами міст у цій множині через побудовані якісні дороги.

Уряд розглядає декілька можливих планів модернізації, кожен з яких відповідає певній підмножині міст із індексами між l і r . Для кожного плану, ваша завдання порахувати мінімальну вартість модернізації за таким планом.

Формат вхідних даних

Перший рядок вводу має два цілих числа n і m ($1 \leq n \leq 20\,000$, $1 \leq m \leq 50\,000$).

Другий рядок вводу має n цілих чисел v_i ($0 \leq v_i < 2^{20}$) розділених пробілами.

Наступні m рядків мають по два числа — l_i і r_i ($1 \leq l \leq r \leq n$) — що описують підмножину міст, що розглядаються у відповідному плані.

Формат вихідних даних

Для кожного з планів, виведіть мінімальну кількість коштів для проведення модернізації за ним у окремому рядку.

Приклад

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

Зауваження

Наприклад, для третього запиту, достатньо провести дороги між містами $(1, 3)$ та $(2, 3)$, що коштуватимуть $1 \oplus 3 = 2$ та $2 \oplus 3 = 1$ відповідно.

Задача К. Червона кнопка

Назва вхідного файлу:	standard input
Назва вихідного файлу:	standard output
Ліміт часу:	6 seconds
Ліміт використання пам'яті:	256 megabytes

Дано два неспадуючі масиви дійсних чисел a_1, \dots, a_n та b_1, \dots, b_n такі, що $0 \leq a_i, b_i \leq 1$ та $a_n = b_n = 1$. Аліса та Боб грають в наступну гру: n разів вони по черзі (спочатку Аліса, потім Боб) можуть або натиснути на червону кнопку, або спасувати. Якщо Аліса натискає червону кнопку в свій i -й хід, то гра завершується, з ймовірністю a_i перемагає Аліса, а з ймовірністю $1 - a_i$ перемагає Боб. Аналогічно, якщо Боб натискає червону кнопку в свій i -й хід, то гра завершується, з ймовірністю b_i перемагає Боб, а з ймовірністю $1 - b_i$ перемагає Аліса. Зверніть увагу, що з умови $a_n = b_n = 1$ випливає, що гра точно завершиться на n -му ході (але, звичайно, може завершитися і раніше).

Надходять q запитів типу "змінити один елемент якогось із масивів". Після кожного запиту потрібно вивести ймовірність того, що за правильної гри обох гравців виграє Аліса.

Формат вхідних даних

Перший рядок містить два цілих числа n, q ($2 \leq n, q \leq 2 \cdot 10^5$) – кількість раундів, та кількість запитів.

Наступні 2 рядки містять по n дійсних чисел a_1, \dots, a_n та b_1, \dots, b_n ($0 \leq a_1 \leq \dots \leq a_n = 1$, $0 \leq b_1 \leq \dots \leq b_n = 1$).

У наступних q рядках задані запити у форматі $c \ i \ x$ ($1 \leq i < n$, $0 \leq x \leq 1$). Якщо $c = A$, то потрібно встановити $a_i = x$, якщо ж $c = B$, то потрібно встановити $b_i = x$.

Гарантується, що після кожного запиту масиви a_i та b_i мають вигляд, описаний в умові. Дійсні числа задані з точністю щонайбільше 9 знаків після коми.

Формат вихідних даних

Виведіть $q + 1$ дійсне число – ймовірність перемоги Аліси перед запитами та після кожного з q запитів.

Відповідь буде зарахована, якщо абсолютна або відносна похибка не буде перевищувати 10^{-9} .

Приклад

standard input	standard output
3 2	0.4
0.2 0.6 1.0	0.5
0.6 0.8 1.0	0.6
A 1 0.5	
B 1 0.2	

Задача L. Їжа для черв'яків

Назва вхідного файлу:	<i>standard input</i>
Назва вихідного файлу:	<i>standard output</i>
Ліміт часу:	4 seconds
Ліміт використання пам'яті:	1024 mebibytes

Маленькі пінгвіни люблять їсти черв'яків. Одного разу їхня мати знайшла довгого черв'яка у вигляді рядка з кількома літерами. Вона вирішила розрізати цього черв'яка на шматочки і погодувати ними маленьких пінгвінят.

Але пташенята можуть їсти лише шматочки з рядками символів, які відповідають умовам ICPC. Тут рядок вважається ICPC, якщо виконуються наступні умови.

- Він складається лише з символів 'C', 'T' і 'P'.
- Два з цих трьох символів зустрічаються однаково кількість разів (можливо, нуль разів) у рядку, а третій символ зустрічається строго більше разів.

Наприклад, "ICPC" і "PPPPPP" є ICPC, але "PIC", "PICCCC" і "PIPE" - ні.

Вам надається рядок на черв'яку, який знайшла мати. Мати хоче розрізати черв'яка на шматочки так, щоб на кожному шматочку рядок був ICPC рядком. Ваше завдання — підрахувати кількість способів підготувати черв'яка таким чином. Оскільки ця кількість може бути великою, виведіть її за модулем 998 244 353.

Формат вхідних даних

Єдиний рядок вхідних даних містить рядок S , що складається лише з символів 'C', 'T' і 'P', який знаходиться на черв'яку у формі рядка. Довжина S знаходиться в межах від 1 до 10^6 , включно.

Формат вихідних даних

Виведіть єдине число — кількість способів представити рядок S як конкатенацію одного або кількох ICPC рядків за модулем 998 244 353.

Приклади

<i>standard input</i>	<i>standard output</i>
ICPC	2
СССІІРРР	69
РІССІССІРРІ	24

Зауваження

У першому прикладі рядок "ICPC" можна представити двома способами.

- Один ICPC рядок, "ICPC".
- Конкатенація чотирьох ICPC рядків, "I", "C", "P" і "C".

Задача М. Позитивне мислення

Назва вхідного файлу: `standard input`
Назва вихідного файлу: `standard output`
Ліміт часу: 1 second
Ліміт використання пам'яті: 256 megabytes

Ви вирішили змінитися і почати думати позитивно. Ви починаєте з масиву цілих чисел. За одну операцію ви можете вибрати будь-яке число і зменшити його або збільшити на 1. Обчисліть мінімальну кількість операцій, щоб зробити добуток всіх чисел у цьому масиві додатнім.

Формат вхідних даних

Перший рядок містить одне ціле число t ($1 \leq t \leq 2 \cdot 10^5$) — кількість наборів тестових даних.

Перший рядок кожного набору містить одне ціле число n ($1 \leq n \leq 2 \cdot 10^5$) — довжину масиву.

Наступний рядок містить n цілих чисел a_1, \dots, a_n ($-10^9 \leq a_i \leq 10^9$) — масив a .

Гарантується, що сума n по всіх наборах не перевищує $2 \cdot 10^5$.

Формат вихідних даних

Для кожного набору тестових даних. виведіть одне ціле число — мінімальну кількість операцій, щоб зробити добуток всіх чисел у цьому масиві додатнім.

Приклад

standard input	standard output
3	3
3	0
-3 -2 -2	3
9	
9 9 8 2 4 4 3 5 3	
3	
0 0 0	

Зауваження

В першому наборі вхідних даних достатньо тричі додати одиничку до другого елемента, отримавши масив $[-3, 1, -2]$ з добутком 6.