



A. Cathedral

Обмеження на час роботи: для C, C++ і Python — 1 секунда, для Java і Kotlin — 2 секунди
Обмеження на використання пам'яті: 256МБ

У іспанському місці Капельяро є багато красивих та історичних будівель. Популярною локацією для огляду всього міста є вершина вежі легендарного Собору святої Лорети Севільяно, що знаходиться в історичному центрі міста. Під час огляду міста з вежі Микола зустрів туристку з Великобританії, яка попросила хлопця про допомогу. А саме — вона попросила знайти та показати, де знаходиться будівля старовинної бібліотеки. Микола не знає, де вона знаходиться та як виглядає, але знає, що вона є 7-ою найвищою будівлею міста (не враховуючи саму вежу Собору). Допоможіть Миколі знайти бібліотеку, або скажіть, що це неможливо однозначно зробити.

Вхідні дані

В першому рядку задано число N (ціле, $1 \leq N \leq 10^5$) — кількість будівель у місті, які можна побачити з вежі. В другому рядку йде N чисел H_i (ціле, $1 \leq H_i \leq 10^9$), i -те з яких є висотою i -ої будівлі.

Вихідні дані

Потрібно вивести порядковий номер будівлі, яка є бібліотекою. Якщо ж це не можливо — то виведіть «imposible» (без лапок).

Приклад

Вхідні дані	Вихідні дані
10 10 9 6 8 7 5 4 3 2 1	7



В. Річка та міста

Обмеження: час роботи — 1 секунда, використання пам'яті — 256МБ

Є річка, берегами якої розташовано N міст, кожне з яких розділено рікою на 2 частини. Хтось H здійснює прогулянку вздовж річки, починаючи з першого міста на лівому березі і прагне зібрати при цьому якнайбільше цікавих йому об'єктів (наприклад, сувенірних фотографій). При цьому H може лише двічі перебраться з одного берега на інший.

Вхідні дані

У першому рядку дано число міст N (ціле, $2 \leq N \leq 50000$). У кожному з наступних N рядків дано по парі чисел nl_i та nr_i (цілі, $0 \leq nl_i, nr_i \leq 20000$) — кількість об'єктів на лівому та правому берегах міста номер i .

Вихідні дані

Одне число — найбільша можлива кількість зібраних об'єктів.

Приклад

Вхідні дані	Вихідні дані
4 1 2 4 3 8 5 3 6	27
3 1 3 2 4 3 3	14
3 1 5 4 4 5 1	19

У першому прикладі можна зібрати $1+4+8=13$ об'єктів на лівому березі, потім $5+6=11$ на правому, а потім останні 3 на лівому.

У другому прикладі — 1 на лівому березі, $3+4+3=10$ — на правому і потім 3 на лівому.

У третьому прикладі — 1 на лівому березі, $5+4=9$ — на правому та $4+5=9$ — на лівому; у цьому випадку перехід через річку відбувається в першому місті та зворотний перехід — у другому.



С. Корінь квантової переваги

Обмеження: час роботи — 1 секунда, використання пам'яті — 256МБ

Для якихось своїх інтриг Гесер доручив мені, Антону Городецькому, переконати керівництво Денної Варти в тому, що Нічна Варта досягла квантової переваги. На щастя, всі рішення у Денній Варті приймає Завулон, а він закінчував лише сільськогосподарський технікум. Увійшовши до кімнати перемовин, я зобразив легку неприязнь, могу собі дозволити, я теж маю поза категоріями.

— Гесер доручив мені пояснити вам продуктивність нашого нового квантового комп'ютера. Не думайте, що я радий тут перебувати.

— Взаємно, Світлий.

— Наш комп'ютер називається «Алдан», назва всім зрозуміло звідки.

Завулон кивнув. Він не знав.

— В якості тестового завдання я пропоную таке. Якщо число помножити само на себе, вийде його квадрат. Зворотна операція називається квадратний корінь. Ми можемо взяти якесь колосальне число, на тисячу цифр, це більше за кількість атомів у Всесвіті. Таке число неможливо ні уявити, ні порахувати до нього на найшвидшому комп'ютері. Навіть щоб просто записати його на папері, доведеться витратити близько години. «Алдан» здатний майже миттєво видати, скільки цифр у квадратному корені з цього велетенського числа.

Завулон явно вразився, але витримав якісну театральну паузу:

— Якщо ваша машина настільки гарна, нехай виведе зараз і першу цифру.

Вхідні дані

Єдине число N (ціле, $0 \leq N < 10^{1000}$). Гарантується, що N — повний квадрат.

Вихідні дані

В одному рядку через пробіл: ціле додатне число D , кількість цифр у квадратному корені з N , і цифра L , перша цифра цього кореня.

D. AI замінить програмістів

Обмеження: час роботи — 1 секунда, використання пам'яті — 256МБ

На дворі 2224 рік. Олег Свічкейс навчається на Землі в X-OSPU (eXtreme Open Source Programming University). В цьому році керівництво X-OSPU прийняло історичне рішення і першим у галактиці дозволило вступати на навчання роботам із просунутим штучним інтелектом. І ось, сьогодні куратор групи Олега привів до студентів їх нового одноклассника, який вступив одразу на чотирнадцятий курс (ну а скільки курсів, ви думаєте, буде потрібно, щоб вивчити C++223?).

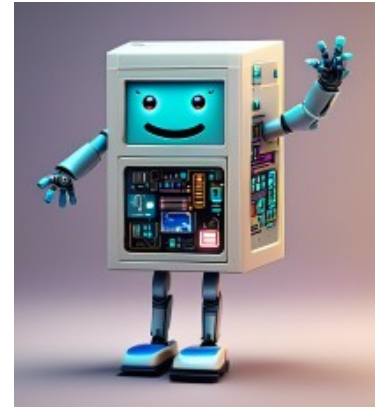
— Привіт, люди! Я — робот-програміст моделі GH-COP1L0T, — сказав новий одноклассник.

Студенти почали із цікавістю розглядати свого нового товариша. А ось Олег дивився на нього похмуро.

— А правда, — нарешті запитав Олег, — що роботи-програмісти витіснять людей?

— Ну що ви, люди, — GH-COP1L0T вивів на свій екранчик доброзичливу посмішку, — нікого я не витісню. Ви подивіться, який я маленький!

Для простоти будемо вважати, що середньостатистична людина-програміст є прямокутним паралелепіпедом розмірами $H \times W \times D$. Робота-програміста моделі GH-COP1L0T, у свою чергу, будемо вважати прямокутним паралелепіпедом розмірами $h \times w \times d$. Скільки роботів-програмістів моделі GH-COP1L0T потрібно для того, щоби повністю витіснити одну людину-програміста? Людина-програміст вважається повністю витісненою, якщо сумарний об'єм роботів-програмістів дорівнює або перевищує об'єм людини-програміста.



Вхідні дані

У першому рядку дано три числа, розділених пробілами: H , W та D — розміри середньостатистичної людини-програміста. У другому рядку в такому самому форматі дано числа h , w та d — розміри робота-програміста моделі GH-COP1L0T. Всі числа є цілими і знаходяться у проміжку $[1; 2000]$.

Вихідні дані

Виведіть єдине число — мінімальну необхідну кількість роботів-програмістів моделі GH-COP1L0T.

Приклад

Вхідні дані	Вихідні дані
1700 450 300 340 150 100	45
1600 400 300 1200 600 280	1
1800 1600 1650 1500 530 320	19

Е. AI замініть страйкарів

Обмеження на час роботи: для C і C++ — 3 секунди, для Java і Kotlin — 5 секунд, для Python — 15 секунд
Обмеження на використання пам'яті: 256МБ

На дворі 2224 рік. Олег Свічкейс навчається на Землі в X-OSPU (eXtreme Open Source Programming University). Тільки що в Олега закінчилися заняття, і він вирішив пообідати у студентській їдальні. Але, прийшовши до їдальні, він побачив несподівану картину. На підлозі їдальні, яку можна вважати площиною, були розкладені крихти хліба, які можна вважати точками на площині. Навколо них стояли смарт-тостери зі штучним інтелектом і клацали своїми важільками. За всім цим нервово спостерігав адміністратор ІТ-інфраструктури їдальні.

— Їдальня тимчасово не працює, — сказав адміністратор. — Ми сьогодні оновили прошивку наших розумних тостерів, і вони після цього з'їхали з глузду та влаштували страйк.

— А чого вони хочуть? — поцікавився Олег.

— Ніхто не може зрозуміти! — адміністратор сплеснув руками. — Але ми думаємо, що їхні вимоги якимось зашифровані у хрестиках, які утворюються розкладеними на підлозі крихтами.

Дійсно, крихти хліба утворювали хрестики. Кожен хрестик складався з п'яти крихт, чотири з яких були кінцями двох відрізків, що не лежать на одній прямій, а п'ята — серединою обох відрізків, а отже, і точкою їхнього перетину. При цьому кожна крихта могла входити у склад одразу декількох різних хрестиків.

— Треба якось порахувати ці їхні хрестики, — сказав адміністратор. — Але, щоб не нарахувати зайвого, виберемо деяке R , і будемо рахувати тільки такі хрестики, в яких співвідношення довжин короткого відрізка до довжини довгого відрізка не менше за $R/1000$.

Допоможіть Олегу якнайшвидше порахувати хрестики, щоб йому не довелося летіти додому на Ганімед голодним.

Вхідні дані

У першому рядку дано два числа, розділених пробілом: N (ціле, $1 \leq N \leq 4000$) та R (ціле, $0 \leq R \leq 1000$) — відповідно: кількість крихт хліба на підлозі та помножене на 1000 мінімальне співвідношення довжин відрізків, з яких складаються хрестики. Далі слідує N рядків, кожен з яких містить по два числа, розділених пробілом: x_i та y_i (цілі, $-10^6 \leq x_i, y_i \leq 10^6$) — координати i -ої крихти хліба. Гарантується, що ніякі дві крихти хліба не знаходяться в одній точці.

Вихідні дані

Виведіть єдине число — кількість хрестиків, які утворюються крихтами хліба на підлозі.

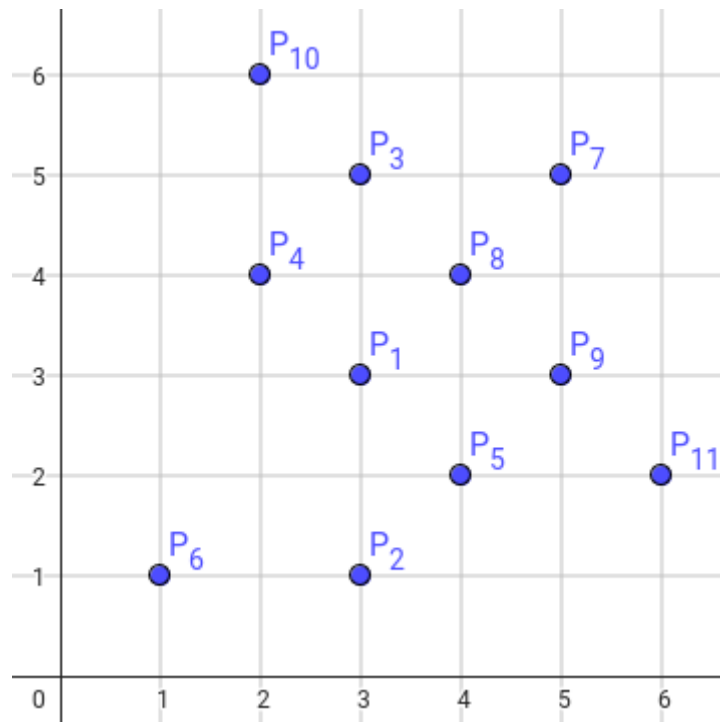




Приклад

Вхідні дані	Вихідні дані
11 600 3 3 3 1 3 5 2 4 4 2 1 1 5 5 4 4 5 3 2 6 6 2	3

Хрестики утворюються групами крихт із наступними номерами: $\{3, 2, 4, 5, 1\}$, $\{3, 2, 7, 6, 1\}$ та $\{7, 1, 3, 9, 8\}$. Група крихт $\{7, 1, 10, 11, 8\}$ не утворює хрестик, бо співвідношення довжин відрізків, що утворюються парами крихт $\{7, 1\}$ та $\{10, 11\}$, дорівнює 0.5, що менше за значення $R/1000$.





F. Chat limits

Обмеження на час роботи: для C, C++, Java і Kotlin — 2 секунди, для Python — 4 секунди
Обмеження на використання пам'яті: 256МБ

ІТ компанія намагається розрахувати на скільки їм вистачить поточного безкоштовного плану використання робочого месенджера з урахуванням кількості співробітників та середньої кількості повідомлень. Сам месенджер має обмеження безкоштовного використання в S повідомлень. Після досягнення цього ліміту писати в месенджер буде неможливо без оформлення платної підписки. Для розрахунків будемо використовувати такі дані:

- З самого початку у компанії працює N співробітників
- В середньому кожного місяця в компанії з'являється E нових співробітників
- В середньому один співробітник відправляє M нових повідомлень щомісяця

Враховуючи числа N , S , E та M вам необхідно розрахувати на скільки **повних** місяців вистачить підписки. Зверніть увагу, що це число приблизне, отже ми не шукаємо мінімальне чи максимальне значення, а просто очікуване, отже ми уявляємо що кожного місяця до нас приходять рівно E нових співробітників, а також кожен співробітник відправляє рівно M повідомлень.

Вхідні дані

В першому та єдиному рядку задано чотири числа — N (ціле, $0 \leq N \leq 10^5$), S (ціле, $1 \leq S \leq 10^9$), E (ціле, $0 \leq E \leq 10^4$) та M (ціле, $1 \leq M \leq 10^5$). Гарантується, що хоча б одне із чисел N або E не дорівнює 0.

Вихідні дані

Потрібно вивести кількість місяців, скільки ІТ компанія зможе користуватися безкоштовним планом.

Приклад

Вхідні дані	Вихідні дані
5 400 5 10	3
10 2000 0 10	20

Пояснення до першого прикладу:

- 1 місяць: 5 співробітників, 50 повідомлень
- 2 місяць: 10 співробітників, 100 повідомлень (150 загалом)
- 3 місяць: 15 співробітників, 150 повідомлень (300 загалом)
- 4 місяць: 20 співробітників, 200 повідомлень (500 загалом)



Г. Знайти цифру в степені двійки

Обмеження: час роботи — 2 секунди, використання пам'яті — 256МБ

Знайти останню (саму праву) позицію цифри D у числі 2^N , $N \leq 10^7$, або вивести -1 , якщо такої нема. Позиції нумеруються з правого кінця, починаючи з 0: 0 — одиниці, 1 — десятки і так далі.

Вхідні дані

В єдиному рядку входу два числа: N (ціле, $0 \leq N \leq 10^7$) та D (ціле, $0 \leq D \leq 9$).

Вихідні дані

В єдиному рядку ціле число P , сама права позиція цифри D у числі 2^N , або -1 , якщо така цифра в ньому не зустрічається.

Приклад

Вхідні дані	Вихідні дані
0 0	-1
0 1	0
16 5	2
64 8	18
7692882 5	11



H. Bicycle Paths

Обмеження: час роботи — 1 секунда, використання пам'яті — 256МБ

У місті H вирішили модернізувати схему руху транспорту та побудувати спеціальні велодоріжки. Однак перед такою глобальною зміною вирішили спочатку провести експеримент і збудувати лише дві прямокутні (за планом міста) велодоріжки, які можуть перетинати чи торкатися одна одної, а також мати спільні ділянки. Згідно з прийнятими правилами велосипедисти можуть пересуватися виключно цими велодоріжками. Є початковий (X_A, Y_A) та кінцевий (X_B, Y_B) пункти, які лежать на цих доріжках. Необхідно знайти мінімальну відстань, яку має проїхати велосипедист доріжками для переміщення між цими пунктами.

Вхідні дані

Перший і другий рядок містять по чотири числа: X_1, Y_1, X_2, Y_2 (цілі, $1 \leq X_1 < X_2 \leq 10000$, $1 \leq Y_1 < Y_2 \leq 10000$) — координати лівого нижнього (X_1 та Y_1) і правого верхнього (X_2 та Y_2) кутів прямокутників-велодоріжок на плані міста. У першому рядку — перша велодоріжка, у другому — друга.

Третій рядок містить чотири числа: X_A, Y_A, X_B і Y_B (цілі, $1 \leq X_A, Y_A, X_B, Y_B \leq 10000$) — координати початкової та кінцевої точки. Гарантується, що кожна точка лежить на стороні одного з прямокутників.

Вихідні дані

Програма повинна видати одне ціле число — мінімальну відстань або -1 , якщо досягти кінцевої точки з початкової неможливо.

Приклад

Вхідні дані	Вихідні дані
3 3 7 5 5 1 9 10 4 5 8 10	9