

## Задача А. Тура

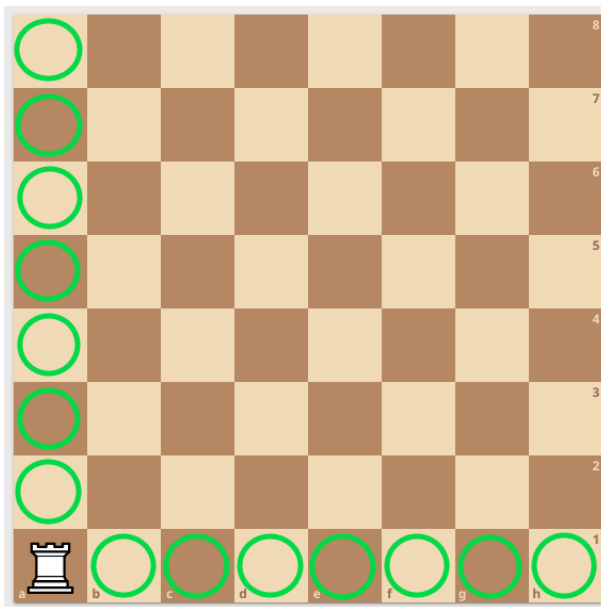
Ліміт часу: 1 second  
Ліміт використання пам'яті: 256 megabytes

Дано шахівниця розміром  $n \times m$ . Тобто з  $n$  рядками та  $m$  стовпчиками.

У цій шахівниці є лише одна фігура — тура. Вона знаходиться у нижньому лівому куті. Більше ніяких фігур немає.

Нагадаємо, що тура за один хід може переміститися на будь-яку додатню кількість клітин по горизонталі або вертикалі, але не по діагоналі.

Знайдіть кількість клітин, на які тура може переміститися за рівно один хід.



На малюнку зображена традиційна шахівниця розміру  $8 \times 8$ . У ній тура може переміститися на всі клітини, які помічені зеленим. Таких всього 14, тому відповідь 14.

### Формат вхідних даних

Перший рядок містить одне ціле число  $n$  ( $1 \leq n \leq 20$ ).

Другий рядок містить одне ціле число  $m$  ( $1 \leq m \leq 20$ ).

### Формат вихідних даних

Виведіть кількість клітин, на які тура може переміститися за один хід.

### Приклади

standard input	standard output
8 8	14
3 2	3

### Зауваження

Пояснення, чому до першого прикладу відповідь 14, можна побачити на малюнку вище.

У другому прикладі відповідь 3, бо тура може переміститися лише на одну позицію вправо та на дві позиції вгору.

## Задача В. Координати

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

Дано точку  $(x, y, z)$  у 3D-просторі.

Знайдіть **квадрат** відстані від цієї точки до центру координат (тобто, до точки  $(0, 0, 0)$ ).

Нагадаємо, що відстань між двома точками  $(x_1, y_1, z_1)$  та  $(x_2, y_2, z_2)$  визначається за формулою

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

### Формат вхідних даних

Перший рядок містить одне ціле число  $x$  ( $-100 \leq x \leq 100$ ).

Другий рядок містить одне ціле число  $y$  ( $-100 \leq y \leq 100$ ).

Третій рядок містить одне ціле число  $z$  ( $-100 \leq z \leq 100$ ).

### Формат вихідних даних

Виведіть одне ціле число.

### Приклад

standard input	standard output
1 -3 5	35

### Зауваження

В першому тесті нас цікавить квадрат відстані від точки  $(1, -3, 5)$  до  $(0, 0, 0)$ .

Підставляємо в формулу координати та отримуємо:

$$\begin{aligned} & \left( \sqrt{(1-0)^2 + (-3-0)^2 + (5-0)^2} \right)^2 = \\ & = \left( \sqrt{1 + 9 + 25} \right)^2 = 35 \end{aligned}$$

## Задача С. Улюблена іграшка Сакурако

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

Сакурако хоче пограти зі своєю улюбленою іграшкою, яка знаходиться в  $m$  метрах від неї. За один крок вона може пройти від 1 до  $k$  метрів вперед (якщо іграшка ближче, ніж  $k$  метрів, вона може переступити через неї). Сакурако поспішає, тому запитує вас, яка мінімальна кількість кроків потрібна, щоб дістатися до іграшки.

### Формат вхідних даних

Перший рядок містить  $m$  та  $k$  ( $1 \leq m, k \leq 10^{18}$ ) — відстань до іграшки від Сакурако, максимальний розмір кроку Сакурако.

Зверніть увагу, що рекомендується використовувати 64-бітні цілі числа (наприклад, `long long` у C++).

### Формат вихідних даних

Виведіть одне число — мінімальна кількість кроків, які потрібно зробити Сакурако.

### Система оцінювання

У цій задачі є умовні блоки. Якщо ваш розв'язок працює правильно для певних обмежень, він отримає певну кількість балів. Зауважте, що тестування відбувається тест за тестом, а не за блоками.

- (40 балів):  $m \leq 10^5$ ;
- (60 балів): без додаткових обмежень.

### Приклади

standard input	standard output
5 3	2
9 3	3

### Зауваження

- У першому прикладі Сакурако може зробити кроки 3 та 2. Також вона може зробити 2 та 3.  
У другому прикладі їй потрібно тричі зробити крок 3.

## Задача D. Не улюблена іграшка Сакурако

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

Сакурако набридла її улюблена іграшка, тому тепер вона хоче пограти з іншою, яка знаходиться в  $m$  метрах від неї в коридорі без кінця.

За один крок вона може пройти від 1 до  $k$  метрів вперед (якщо іграшка ближча, ніж  $k$  метрів, вона може перестрибнути через неї), але іграшка не хоче грати з Сакурако, тому вона рухається на  $q$  метрів вперед від Сакурако **одночасно** з нею.

Сакурако поспішає, тому вона запитує вас, яка мінімальна кількість кроків потрібна, щоб дістатися до іграшки, якщо це можливо, і якщо вона починає з позиції 1 в коридорі.

### Формат вхідних даних

Перший рядок містить три цілі числа  $m, k, q$  ( $1 \leq m, k, q \leq 10^{18}$ ) — відстань до іграшки від Сакурако, максимальний розмір кроку Сакурако та розмір кроку іграшки.

Зверніть увагу, що рекомендується використовувати 64-бітні цілі числа (наприклад, `long long` у C++).

### Формат вихідних даних

Виведіть єдине число — мінімальна кількість кроків, які потрібно зробити Сакурако, або  $-1$ , якщо дістатися до іграшки неможливо.

### Система оцінювання

У цій задачі є умовні блоки. Якщо ваш розв'язок працює правильно для певних обмежень, він отримає певну кількість балів. Зауважте, що тестування відбувається тест за тестом, а не за блоками.

- (50 балів):  $m \leq 10^5$ ;
- (50 балів): без додаткових обмежень.

### Приклади

standard input	standard output
6 8 4	2
10 9 5	3

### Зауваження

У першому прикладі хронологія виглядає наступним чином:

- Сакурако рухається з 0 до 8 в той час, як іграшка рухається з 6 до 10;
- Сакурако рухається з 8 до 14 (зауважте, що вона зробила не максимальний розмір) в той час, як іграшка рухається з 10 до 14.

У другому прикладі хронологія виглядає наступним чином:

- Сакурако рухається з 0 до 9 в той час, як іграшка рухається з 10 до 15;
- Сакурако рухається з 9 до 18 в той час, як іграшка рухається з 15 до 20;
- Сакурако рухається з 18 до 25 в той час, як іграшка рухається з 20 до 25.

## Задача Е. Успішний інвестор

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

Будучи професійним трейдером, Сакурако створила план на наступні  $n$  днів. У  $i$ -й день вона повинна інвестувати певну кількість монет у діапазоні  $[a_i - x, a_i + x]$ , де  $x$  — це фіксоване ціле число, яке Сакурако має на самому початку. Зверніть увагу, що вона може інвестувати навіть від'ємну кількість монет, що є неймовірним.

Однак вона не була б професійним трейдером, якби не мала козиря у рукаві. Її стратегія полягає в тому, що вона інвестуватиме строго більше монет, ніж вона інвестувала в попередній день. Формально, якщо вона вирішить інвестувати **цілочисельні** монети  $b_1, b_2, \dots, b_n$  відповідно в кожен з  $n$  днів, то умова  $b_1 < b_2 < \dots < b_n$  повинна виконуватись, де  $a_i - x \leq b_i \leq a_i + x$ .

Ваше завдання — виявити, чи може вона виконати свої інвестиції згідно з її стратегією, і вивести один з можливих планів Сакурако.

### Формат вхідних даних

Перший рядок містить два цілі числа  $n$  та  $x$  ( $1 \leq n \leq 2 \cdot 10^5; 0 \leq x \leq 10^9$ ) — кількість днів та фіксоване ціле число Сакурако.

Другий рядок містить  $n$  цілих чисел  $a_1, a_2, \dots, a_n$  ( $0 \leq a_i \leq 10^9$ ) — значення, які описують план Сакурако.

### Формат вихідних даних

У першому рядку виведіть "YES", якщо план Сакурако може бути успішним, і "NO" в іншому випадку.

Якщо у вашому виході в першому рядку "YES", виведіть  $n$  цілих чисел  $b_1, b_2, \dots, b_n$  ( $-2 \cdot 10^9 \leq b_i \leq 2 \cdot 10^9$ ) її остаточних інвестицій у другому рядку.

**Зверніть увагу, що може бути багато можливих відповідей; вам дозволяється вивести будь-яку, яка відповідає умовам, описаним у легенді.**

### Система оцінювання

У цій задачі є умовні блоки. Якщо ваше рішення працює правильно для певних обмежень, воно отримає певну кількість балів. Зверніть увагу, що тестування відбуватиметься тест за тестом, а не за блоками.

- (16 балів):  $x = 1$ ; тобто, фіксоване число Сакурако дорівнює 1.
- (84 бали): без додаткових обмежень.

### Приклад

standard input	standard output
5 4	YES
1 4 2 1 3	-3 0 1 2 3

### Зауваження

У першому прикладі,

- $1 - 4 \leq -3 \leq 1 + 4$ ;
- $4 - 4 \leq 0 \leq 4 + 4$ ;
- $2 - 4 \leq 1 \leq 2 + 4$ ;
- $1 - 4 \leq 2 \leq 1 + 4$ ;
- $3 - 4 \leq 3 \leq 3 + 4$ .

## Задача F. Різдвяний вечір

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

Різдво наближається, і Сакурако святкуватиме його зі своїми близькими друзями. Щоб зробити свято ще більш захоплюючим, вони планують провести обмін подарунками в Різдвяний вечір.

Сакурако знає, що обмін подарунками відбудеться за круглим столом, і цей обмін буде дуже простим: кожен подарує свій подарунок людині, що сидить праворуч від нього. Всі вже вирішили, де вони сидітимуть, окрім Сакурако.

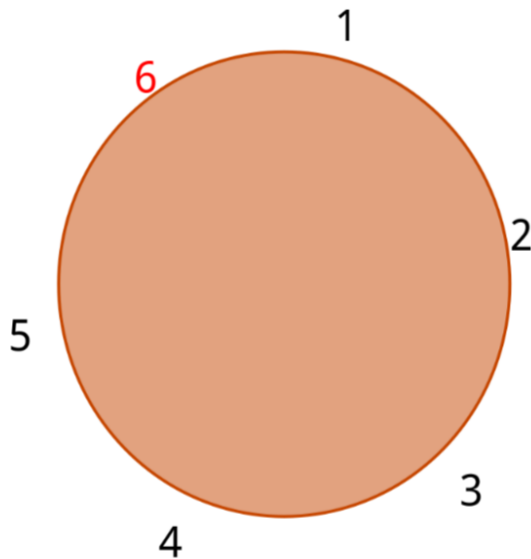
Вона також знає ціну подарунка кожного друга; подарунок  $i$ -го друга коштує  $a_i$  монет. Звісно, після обміну хтось може бути розчарованим, якщо отримає подарунок дешевший, ніж той, що він подарував. Інтуїтивно, їхнє розчарування описується як абсолютна різниця між вартістю подарунка, який вони подарували, і подарунка, який отримали. Значення, яке цікавить Сакурако, — це загальне розчарування.

Більш формально, якщо  $b_1, b_2, \dots, b_n$  — це вартості подарунків друзів відповідно, то загальне розчарування обчислюється як:

$$|b_1 - b_2| + |b_2 - b_3| + \dots + |b_{n-1} - b_n| + |b_n - b_1|.$$

Сакурако, у свою чергу, купила подарунок за  $x$  монет. Оскільки вона ще не вирішила, де сидіти, ви можете дати їй підказку. Скажіть їй, яке мінімальне загальне розчарування може бути, якщо Сакурако вибере своє місце оптимально, адже вона хоче, щоб настрої Різдвяного вечора був якомога веселішим. З вашою підказкою вона зможе самостійно знайти оптимальне місце.

Наприклад, якщо у Сакурако є 5 друзів, де  $i$ -й друг купив подарунок за  $i$  монет, а подарунок Сакурако коштує 6 монет, то одне з можливих місць, де може сидіти Сакурако, ілюструється нижче:



Тут Сакурако сидить між першим і п'ятим другом. Отже, загальне розчарування становить

$$|1 - 2| + |2 - 3| + |3 - 4| + |4 - 5| + |5 - 6| + |6 - 1| = 10.$$

### Формат вхідних даних

Перший рядок містить два цілі числа  $n$  та  $x$  ( $1 \leq n \leq 2 \cdot 10^5; 0 \leq x \leq 10^9$ ) — кількість друзів та вартість подарунка Сакурако.

Другий рядок містить  $n$  позитивних цілих чисел  $a_1, a_2, \dots, a_n$  ( $0 \leq a_i \leq 10^9$ ) — вартості подарунків друзів.

### Формат вихідних даних

Виведіть одне ціле число — мінімальне загальне розчарування після того, як Сакурако займе своє місце.

## Система оцінювання

У цій задачі є умовні блоки. Якщо ваше рішення працює правильно для певних обмежень, воно отримує певну кількість балів. Зверніть увагу, що тестування відбуватиметься тест за тестом, а не за блоками.

1. (40 балів):  $n \leq 10^3$ ;
2. (60 балів): без додаткових обмежень.

## Приклад

standard input	standard output
5 6 1 2 3 4 5	10

## Задача G. Чефір загубився

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

Нещодавно Чефір загубився на координатній осі. Зараз він знаходиться на позиції  $n$ . Щоб дістатися до Сакурако, яка знаходиться на позиції  $x$ , він може багаторазово виконувати наступні операції: Якщо ваша поточна позиція  $a$ , ви можете змінити її на:

- $a = a - q$ , де  $q$  — мінімальний дільник  $a$ , такий що  $q > 1$ .
- $a = a - w$ , де  $w$  — максимальний дільник  $a$ , такий що  $w < a$ .

Ваше завдання не лише визначити, чи існує послідовність операцій, яка допоможе Чефіру дістатися з позиції  $n$  до позиції  $x$ , але й вивести цю послідовність.

### Формат вхідних даних

Перший рядок містить два цілі числа  $n$  та  $x$  ( $2 \leq x < n \leq 10^{12}$ ).

### Формат вихідних даних

Якщо це можливо, виведіть "YES"; в іншому випадку виведіть "NO".

Якщо це можливо, у наступному рядку виведіть одне ціле число  $k$  ( $0 \leq k \leq 500$ ) — кількість операцій в скороченому записі, які ви плануєте виконати. **Немає** потреби мінімізувати  $k$ .

Кожен з наступних  $k$  рядків повинен містити два цілі числа  $op, y$  ( $op \in \{1, 2\}$  та  $1 \leq y \leq 10^{12}$ ) — тип операції, яку ви виконуєте, та кількість разів, яку ви будете повторювати цю операцію.

### Система оцінювання

У цьому завданні кожен тест оцінюється окремо.

У цьому завданні ви можете отримати часткові бали, лише відповівши, чи є спосіб для Чефіра дістатися з позиції  $n$  до позиції  $x$ .

Щоб отримати повні бали, ви повинні вказати точний спосіб переходу з  $n$  до  $x$ .

Якщо ви правильно визначите, що існує послідовність, але виведете  $k = 0$  і не надасте саму послідовність після цього, ви отримаєте 50% балів.

### Приклади

standard input	standard output
20 9	YES 2 1 1 2 1
25 9	YES 2 1 2 2 1
24 13	NO

### Зауваження

У першому прикладі оптимальне рішення таке:

- спочатку відняти 2 як найменший дільник 20, отримуючи  $n = 18$ ;
- відняти 9 від 18 як найбільший дільник  $n$ , який не дорівнює  $n$ .

Тепер  $n = 9$ , що дорівнює  $x$ .

У другому прикладі  $n$  змінюється наступним чином:

$25 \rightarrow 20 \rightarrow 18 \rightarrow 9$ .

Ми можемо довести, що немає способу дістатися з 24 до 13.



## Задача Н. Сакурако та падаюче дерево

Ліміт часу: 1.5 seconds  
Ліміт використання пам'яті: 256 megabytes

Коли є кіт і різдвяна ялинка...

Сакурако боїться, що Чефір повалить її красиву різдвяну ялинку і зламає всі  $m$  прикрас на ній. Щоб цього уникнути, вона вибере деякі вершини дерева та підкріпить їх.

Дерево можна представити як граф з вершинами, позначеними числами від 1 до  $n$ , в якому кожна вершина, крім кореневої (позначеної як вершина 1), має батьківську вершину з номером, що має значення менше за його власне.

Прикраса не зламається, якщо і тільки якщо вона з'єднана принаймні з однією іншою прикрасою. Пара прикрас  $(x, y)$  вважається з'єднаною, якщо всі вершини на шляху від  $x$  до  $y$  підкріплені. Шлях від  $x$  до  $y$  — це список вершин, який забезпечить найшвидший прохід між ними, якщо ви можете рухатися лише від вершини до її батьківських або дочірніх вершин.

Ваше завдання — знайти мінімальну кількість вершин, які потрібно підкріпити, щоб жодна з прикрас Сакурако не зламалася.

### Формат вхідних даних

Перший рядок містить два цілі числа  $n$  та  $m$  ( $2 \leq m \leq n \leq 10^6$ ).

Другий рядок містить  $n - 1$  чисел  $p_2, p_3, \dots, p_n$  ( $2 \leq i \leq n$ ) — вершина, до якої приєднана  $i$ -та вершина.

Третій рядок містить  $m$  чисел  $a_1, a_2, \dots, a_m$  ( $1 \leq a_i \leq n$ ) — позиції кожної з  $m$  прикрас. Гарантується, що позиції попарно різні.

### Формат вихідних даних

Виведіть одне число — мінімальну кількість вершин, які потрібно зміцнити, щоб кожна прикраса залишалася цілою.

### Система оцінювання

У цій задачі є умовні блоки. Якщо ваш розв'язок працює правильно для певних обмежень, він отримає певну кількість балів. Зауважте, що тестування відбувається тест за тестом, а не за блоками.

- (13 балів):  $n \leq 15$ ;
- (10 балів):  $m = 2$ ;
- (31 бал):  $n \leq 10^4$ ;
- (46 балів): без додаткових обмежень.

### Приклад

standard input	standard output
11 6 1 1 1 1 5 6 7 8 8 8 2 3 4 9 10 11	8

### Зауваження

У першому тестовому випадку ми виберемо вершини 1, 2, 3, 4, 8, 9, 10, 11. Прикраси будуть з'єднані наступним чином:  $1 \iff 2$ ,  $1 \iff 3$ ,  $4 \iff 5$ ,  $5 \iff 6$ .