

A. Гіпотенуза

Time limit: 1 s

Memory limit: 256 M

Дано два числа a та b . Виведіть гіпотенузу трикутника із заданими катетами з точністю 6 знаків.

Вхідні дані
Два числа.

Вихідні дані

Examples

Input	Output
3 4	5.0

B. Кролики

Time limit: 1 s

Memory limit: 256 M

Когда земляне, наконец, нашли обитаемую планету, они назвали ее OLYMP и отправили на нее вместе с космическим кораблем одного кролика. Кролику понравился климат новой планеты и через месяц он произвел на свет еще одного кролика. Далее кролики продолжили размножаться с такой же скоростью, то есть каждый месяц каждый из кроликов, присутствующих на планете, производил на свет еще одного кролика. Однако, размножение кроликов сдерживал откуда-то взявшийся на планете монстр. Как только в начале какого-то месяца кроликов становилось строго больше, чем k , он приходил и съедал k кроликов. Определите, сколько кроликов будет на планете через n месяцев после прибытия туда космического корабля с первым кроликом. Число n от 0 до 100 включительно. Число k от 0 до 10000 включительно. Результат работы метода не превосходит 2000000000.

Вхідні дані

Целое число n - количество месяцев. Целое число k - количество кроликов, съедаемых монстром.

Вихідні дані

Целое число, равное количеству кроликов на планете OLYMP через n месяцев после поселения туда первого кролика.

Examples

Input	Output
0 10	1
1 10	2
100 1024	2048

16 0	65536

C. Наступне і попереднє

Time limit: 1 s

Memory limit: 256 M

Напишіть програму, яка зчитує ціле число і виводить текст, аналогічний наведеному у прикладі. Пробіли, розділові знаки, великі і малі букви важливі!

Examples

Input	Output
25	The next number for the number 25 is 26. The previous number for the number 25 is 24.

D. Сума цифр числа

Time limit: 1 s

Memory limit: 256 M

Задано чотирьохзначне число N. Напишіть програму, яка визначає суму цифр даного числа.

Вхідні дані

число N.

Вихідні дані

знайдена сума.

Examples

Input	Output
1012	4

E. НСД та НСК

Time limit: 1 s

Memory limit: 256 M

Дано два числа. Знайдіть їх найбільший спільний дільник та найменше спільне кратне.

Вхідні дані

два числа.

Вихідні дані

два числа.

Examples

Input	Output
2 4	2 4

F. Кількість чисел

Time limit: 1 s

Memory limit: 256 M

Вводиться послідовність чисел до тих, поки не буде введено два рівних числа підряд. Підрахуйте кількість чисел у послідовності.

Вихідні дані

Виведіть кількість чисел (включаючи два останніх).

Examples

Input	Output
3 5 24 4 3 5 3 5 3 5 5	11

G. Факторізація числа

Time limit: 1 s

Memory limit: 256 M

Факторизацією числа N назвемо сукупність не менше двох натуральних чисел, кожне з яких більше за одиницю, які у добутку дають дане число N . Порядком множників будемо нехтувати, тому $2 * 4$ і $4 * 2$ – однакові факторизації числа 8. Для прикладу, існує всього 4 різні факторизації числа 30:

$2 * 3 * 5,$
 $5 * 6,$
 $3 * 10,$
 $2 * 15.$

Напишіть програму FACTOR, яка визначає кількість різних факторизацій даного числа N .

Вхідні дані

У єдиному вхідному рядку міститься одне число $N(2 \leq N \leq 2000000000)$.

Вихідні дані

У єдиному рядку вихідного потоку повинно міститися єдине число – кількість різних факторизацій даного числа N .

Examples

Input	Output
30	4

Н. Гра "Життя"

Time limit: 1 s

Memory limit: 256 M

Гра «Життя» була придумана англійським математиком Джоном Конвейем у 1970 році. Уперше опис цієї гри опубліковано в жовтневому випуску (1970) журналу Scientific American, у рубриці «Математичні ігри» Мартіна Гарднера. Місце дії цієї гри - «всесвіт» - це розмічена на клітинки поверхня. Кожна клітинка на цій поверхні може перебувати у двох станах: бути живою або бути мертвою. Клітинка має вісім сусідів. Розподіл живих клітинок на початку гри називається першим поколінням. Кожне наступне покоління розраховується на основі попереднього за такими правилами: - порожня (мертва) клітинка з рівно трьома живими клітинками-сусідами оживає; - якщо в живій клітинки є два або три живих сусіда, те ця клітинка продовжує жити; у противному випадку (якщо сусідів менше двох або більше трьох) клітка вмирає (від «самітності» або від «перенаселеності»). У цій задачі розглядається гра «Життя» на торі. Уявимо собі прямокутник розміром n рядків на m стовпців. Для того, щоб перетворити його в тор подумки «склеїмо» його верхню сторону з нижньою, а ліву із правою. Таким чином, у кожній клітинки, навіть якщо вона раніше перебувала на границі прямокутника, тепер є рівно вісім сусідів. Ваше завдання полягає в тому, щоб знайти конфігурацію клітинок, що буде через k поколінь від заданого.

Вхідні дані

Перший рядок вхідного потоку містить три цілих числа: n, m, k ($4 \leq n, m \leq 100, 0 \leq k \leq 100$). Наступні n рядків містять по m символів кожен і описують початкову конфігурацію. j -ий символ i -го рядка містить «.» (точка), якщо відповідна клітинка мертва, і «*» - якщо жива.

Вихідні дані

У вихідний потік виведіть конфігурацію клітинок через k поколінь після початкового у тому ж форматі, у якому конфігурація задається у вхідних даних.

Examples

Input	Output
<pre> 5 5 1 **... ..** .*... ..*.. ...*. </pre>	<pre> .*.** *... .*.* .*... .*... **.. </pre>
<pre> 5 5 5 **... ..** .*... ..*.. ...*. </pre>	<pre> .***. *... *... .**. *... </pre>
<pre> 4 7 5 .*.*.* *.*.*.* .*.*.* *.*.*.* </pre>	<pre> </pre>

I. Ланцюжок слів

Time limit: 1 s

Memory limit: 256 M

Ланцюжком слів довжини n назвемо послідовність слів w_1, w_2, \dots, w_n таку, що для $1 \leq i < n$ слово w_i є власним префіксом слова w_{i+1} . Нагадаємо, що слово u довжини k називається власним префіксом слова v довжини l , якщо $l > k$ і перші k букв слова v співпадають із словом u . Задана множина слів $S = \{s_1, s_2, \dots, s_m\}$. Знайдіть максимальну довжину ланцюжка слів, який можна побудувати, використовуючи (можливо, не всі) слова цієї множини.

Вхідні дані

Перший рядок вхідного потоку містить ціле число m ($1 \leq m \leq 255$). Наступні m рядків містять по одному слову із множини S . Всі слова не пусті, мають довжину, не більшу за 255 символів, і складаються тільки з маленьких букв латинського алфавіту.

Вихідні дані

У вихідний потік виведіть відповідь на задачу.

Examples

Input	Output
<pre> 3 a ab abc </pre>	<pre> 3 </pre>

5	2
a	
ab	
bc	
bcd	
add	

J. Точки у многокутнику

Time limit: 1 s

Memory limit: 256 M

Многокутник складається із N ($N \leq 10000$) вершин, координати яких задані в декартовій системі координат і є цілими числами (по модулю не більші 1 000 000). Потрібно знайти кількість точок з цілочисельними координатами, які лежать в самому многокутнику (не на границі). Сторони многокутника є взаємонеперетинаючимися (за виключенням сусідніх – у вершинах).

Вхідні дані

У першому рядку вхідного потоку знаходиться число N , в наступних N рядках – пари чисел – координати точок. Якщо з'єднати точки в даному порядку, а також з'єднати першу і останню точки, то отримаємо заданий многокутник.

Вихідні дані

У вихідному потоку повинно міститись число – шукана кількість точок.

Examples

Input	Output
4	361
-10 -10	
-10 10	
10 10	
10 -10	