

Anagramas

Una cadena **X** es un anagrama de una cadena **Y** si se pueden reacomodar las letras de **X** de tal modo que se obtenga exactamente la cadena **Y**. Por ejemplo, las cadenas "dog" y "god" son anagramas, también lo son las cadenas "baba" y "aabb", pero las cadenas "abbac" y "caaba" no lo son.

Como te puedes dar cuenta es muy fácil identificar si una cadena es un anagrama de otra, pero, ¿es igual de fácil identificar todas aquellas palabras de una lista de **N** cadenas que son anagramas de una cadena **S** indicada? Sabemos que tu respuesta a esta pregunta es un rotundo "Sí".

Problema

Tu tarea como podrás imaginar es escribir un programa que dada una lista de cadenas y una cadena **S** cuente cuantas cadenas de la lista son un anagrama de **S**

Entrada

En la primer línea dos números enteros **N** y **M** que representan el número de cadenas que contiene la lista y la longitud que tiene cada cadena respectivamente (todas las cadenas son de esta longitud). La segunda línea contiene una cadena de **M** caracteres, la cadena **S**. Cada una de las siguientes **N** líneas contiene una cadena de **M** caracteres la cual representa una de las cadenas de la lista para comparar.

Salida

Tu programa deberá imprimir una línea con un único número entero, el número de cadenas de la lista que son un anagrama de **S**

Ejemplo

Entrada	
5 6 omijal jpablo jalomi imojal alomij marinr	3

Límites

- $1 \leq N \leq 10000$
- $1 \leq M \leq 100$
- Todas las cadenas solo contienen letras minúsculas y no contienen espacios

La serie

Pedro es un aficionado de las series numéricas. Es tan aficionado que se divierte mucho inventando series nuevas cada que puede, y decimos cada que puede, porque al ser también un joven estudiante Pedro tiene que preparar sus labores diarias que al final le han quitado mucho de su tiempo libre.

Como Pedro no tiene mucho tiempo, ha tenido que pensar en una manera de generar series numéricas utilizando el mismo método y sin perder mucho tiempo, así que comienza a generar series de la siguiente manera:

Pedro tomará un número **K** como el número que inicia la serie en la posición **1**. El siguiente elemento de la serie se obtiene sumando al último elemento que se ha agregado en la serie y la posición en la serie del elemento que se va a agregar. Por ejemplo: si pedro toma el número **5** como el primer elemento de la lista, el siguiente elemento será el último elemento (**5**) mas la posición del elemento que sigue de colocar (**2**) es decir: **7**

Como compañero de Pedro, no te gusta verlo sufrir por no tener tanto tiempo como quiere para hacer sus series, así que te en una labor de solidaridad le quieres obsequiar un programa con el cual Pedro puede indicar cuál es el número inicial y cuántos elementos quiere obtener de la serie para que el programa se lo genere automáticamente.

Problema

Tu tarea es realizar un programa que dados la cantidad de elementos **N** que debe tener una serie y el valor inicial **K** de la serie imprima la serie para Pedro.

Entrada

La entrada consiste de dos líneas. En la primer línea el número **N**. En la segunda línea el número **K**.

Salida

Tu programa debe imprimir una linea con **N** números enteros separados por espacio. La serie que se obtiene con los valores dados siguiendo el método para generar series de Pedro.

Ejemplo

Entrada	
5 10	10 12 15 19 24

Límites

- $1 \leq N \leq 100$
- $1 \leq K \leq 100$

Sumas y Restas

Este problema solo requiere de sumar, restar y mucho ingenio. Sumar y restar es fácil, pero fue tanto el ingenio que tuvimos que poner que no nos quedo más ingenio para hacer una historia de este. Vamos directo al problema.

Problema

Dado un arreglo de **N** números enteros, tienes que encontrar si existe alguna manera que usando solo los símbolos '+' y '-' entre los elementos del arreglo se forme una expresión que al evaluarla el valor que se obtenga sea un número múltiplo de **M**.

Entrada

La entrada consiste de dos líneas. En la primer línea los números **N** y **M** separados por un espacio. En la segunda línea **N** separados por espacio.

Salida

Tu programa debe imprimir en una línea la palabra "Si" sin las comillas en caso de que exista alguna manera de generar la expresión, imprime una línea con la palabra "No" en caso de que no sea posible generar la expresión.

Ejemplo

Entrada	
5 4 1 2 3 4 6	Si

Límites

- $1 \leq N \leq 1000$
- $1 \leq M \leq 1000$
- Para un 50% de los casos de prueba $1 \leq N \leq 20$
- Se eliminarán de evaluación todos aquellos envíos que impriman un resultado que no haya sido computado procesando los datos de entrada.

Explicación

Si se pusieran los símbolos '+' y '-' entre los elementos formando la siguiente expresión:

1 - 2 + 3 + 4 + 6

Al evaluar la expresión el resultado sería 12 el cuál es divisible por 4, por eso la respuesta al caso de prueba es Si

Base de energía

Cuadradonia como te puedes imaginar por su nombre es una ciudad que usan o abusan de los cuadrados. Prácticamente todo lo que hay en cuadradonia desde habitantes hasta edificios y monumentos son cuadrados, tienen una justificación que suena objetiva, al ser todo cuadrado es más fácil de alinear, al ser fácil de alinear es fácil de ordenar, de construir, bueno, básicamente es más fácil para todo.

A pesar de que es supuestamente más fácil todo, es un poco difícil llegar a ciertas respuestas cuando se tiene un pensamiento cuadrado, así que te han buscado por tu pensamiento siempre fuera de la caja para que les apoyes a resolver un pequeño problemita.

Cuadradonia está implementando un nuevo sistema de energía basandose en energía que absorben del suelo, tienen un plano que ya no es sorpresa que sea un cuadrado de **N** por **N** celdas de energía, cada una de estas celdas provee una cantidad **C_{ij}** de energia. Para completar el sistema, Cuadradonia tiene que construir una base de energía (que aún no ha sido construida pero también será cuadrada) sobre las celdas de energía, la base proveerá a Cuadradonia de tanta energía como la suma de las energías que tienen las celdas sobre las que esta se construye.

Cuadradonia sabe que necesitan al menos **X** energía para poder satisfacer todas las necesidades de la ciudad, así que ellos quieren saber, en cuántas posiciones diferentes se puede colocar la base de modo que entregue la energía que Cuadradonia necesita.

Problema

Dados el mapa de celdas de energía y la cantidad de energía que se requiere para satisfacer las necesidades de Cuadradonia, encuentra de cuántas maneras diferentes se podría colocar la base para entregar la energía necesaria.

Entrada

La primer línea de entrada contiene 2 números enteros separados por espacio **N** y **X** Las siguientes **N** líneas contienen cada una **N** números enteros separados por espacio representando la cantidad de energía que entrega esa posicion en las celdas de energía.

Salida

Tu programa deberá imprimir una línea con un número entero, el número de formas diferentes en las que se puede colocar la base de energía para satisfacer las necesidades energéticas de Cuadradonia.

Ejemplo

Entrada	
2 4 3 4 3 2	2

Límites

- $1 \leq \mathbf{N} \leq 1000$
- $1 \leq \mathbf{X} \leq 10^9$
- $1 \leq \mathbf{C_{ij}} \leq 100$
- Para un 50% de los casos de prueba $1 \leq \mathbf{N} \leq 100$

Explicación

En el caso de ejemplo solo se puede construir la base de dos maneras. Una en la posición (1,1) que tiene 4 unidades de energía. La otra abarcando todo el cuadrado que suma un total de 12 unidades de energía.