

Breakdown:

Hay dos reglas que se necesitan cumplir para poder afirmar que una palabra x encaja con la palabra y :

- Ambas palabras necesitan tener el mismo número de caracteres
- En caso de cumplir la primera regla, debe de ser posible poder sustituir cualquier caracter de la primera palabra con los pares de letras proporcionados para poder formar la segunda palabra, en caso de que tengan caracteres distintos

Inputs:

De inputs, se van a recibir primero dos enteros m ($1 \leq m \leq 500$) y n ($1 \leq n \leq 50$), en donde m es el número de traducciones que se van a recibir y n el número de palabras que habrá por comparar con otras

Ejemplo

| | |
|-------|-------|
| 5 | 2 |
| a | z |
| b | y |
| c | x |
| d | w |
| e | t |
| hello | helli |
| word | world |

Seguido de m y n , se reciben las combinaciones de caracteres a y b , estos caracteres indican que a puede ser sustituido por su pareja de b .

*notq, todas las entradas son en minúsculas y no hay caracteres especiales.

Reglas pre-establecidas por los inputs:

- $(1 \leq m \leq 500)$

- con esto se sabe que en donde sea que se vayan a guardar las combinaciones, su tamaño máximo será 500.

- $(1 \leq n \leq 50)$

- en caso de guardar el conjunto de palabras a comparar, se sabe que no habrán más de 50.

- $(1 \leq \text{word} \leq 50)$

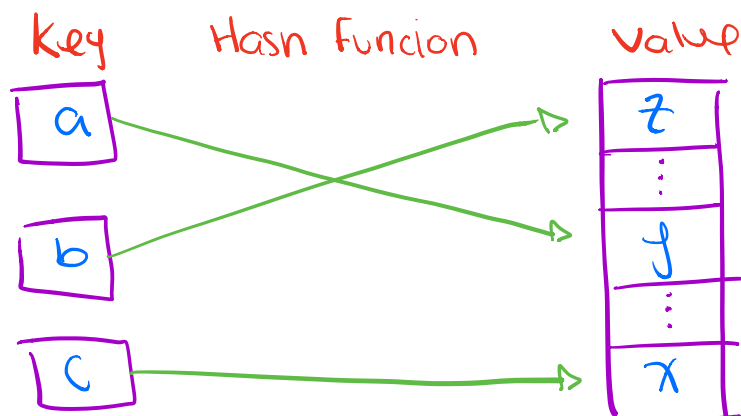
- cada palabra tendrá como mínimo una letra y como máximo, 50.

Output

- en caso de que se cumplan las dos reglas, dará como output por palabra evaluada un "yes", de lo contrario, dará un "no".

Posible solución (1era)

- utilizar una hash table para guardar las traducciones en ella, guardando a como clave y b como valor. (unhash)
- la hash table se utilizaría al momento de hacer las traducciones para comprobar la regla 2.



- Una vez guardado todo el diccionario, se compararán las dos palabras. Primero, se compara el tamaño de cada una. En caso de ser diferentes, se termina el proceso.
- En caso de cumplir con la primera regla evaluar y comparar letra por letra hasta encontrar una diferencia.
- En caso de haber diferencia, buscar en la hashtable el carácter de la palabra **a** y comparar el valor con el carácter de la palabra **b**.

Ejemplo

hello \leftrightarrow hello

| | | |
|---|-------|---|
| a | _____ | x |
| b | _____ | y |
| c | _____ | z |
| d | _____ | n |
| e | _____ | i |
| f | _____ | j |

ya que tienen la misma cantidad de caracteres, se evalúa la siguiente regla.
Al recorrer las palabras, se llega a la conclusión de que ambas son iguales y se regresa un "yes"

aaba \leftrightarrow aabx

| | | |
|---|-------|---|
| a | _____ | x |
| b | _____ | y |
| c | _____ | z |
| d | _____ | n |
| e | _____ | i |
| f | _____ | j |

cumple con la primera regla y comienza a recorrer las palabras hasta encontrar una diferencia. La última letra es diferente y al buscar en el diccionario, concuerdan todas y se regresa un "yes"

abcd \leftrightarrow aacd

| | | |
|---|-------|---|
| a | _____ | x |
| b | _____ | y |
| c | _____ | z |
| d | _____ | n |
| e | _____ | i |
| f | _____ | j |

Cumple con la primera regla, pero al recorrer las palabras se encuentra con un carácter distinto, lo busca en el diccionario y no encuentra relación. Se termina el proceso y regresa un "no" como respuesta.