

Arrays   $\Rightarrow$  Fixed Size

Vectors   $\Rightarrow$  Dynamic Size

**Maps** (key, value) Pairs

key	Value
GJ	Gujarat
MH	Maharashtra
DL	Delhi

Numbers

D.T	D.T
(5) $\rightarrow$	5
(3) $\rightarrow$	10

int, char, string, bool  
 $\downarrow$   
double  
float

String  $\rightarrow$  Int

Write a Program to check if all the characters in a string are having same number of occurrences.

For Example

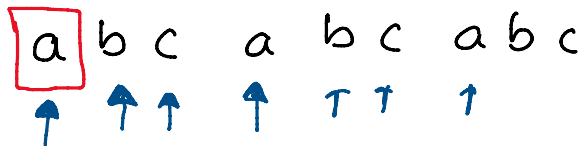
s = "abcabcabc"

Output = True

a  $\rightarrow$  3

b  $\rightarrow$  3

c  $\rightarrow$  3



s = "aabbc"

Output = False

a  $\rightarrow$  2

b  $\rightarrow$  2

char	int
a $\rightarrow$ 2	3
b $\rightarrow$ 2	3
c $\rightarrow$ 1	1

Output = raise

a --> 2

b --> 2

c --> 1



When we initialize the value as integer datatype --> The default value is 0.

### Problem Statement :

Count Common Words With One Occurrence

Given two string arrays `words1` and `words2`, return the **number of strings** that appear **exactly once** in **each of the two arrays**.

#### Example 1:

**Input:** `words1 = ["leetcode", "is", "amazing", "as", "is"]`, `words2 = ["amazing", "leetcode", "is"]`

**Output:** 2

#### Explanation:

- "leetcode" appears exactly once in each of the two arrays. We count this string.
- "amazing" appears exactly once in each of the two arrays. We count this string.
- "is" appears in each of the two arrays, but there are 2 occurrences of it in `words1`. We do not count this string.
- "as" appears once in `words1`, but does not appear in `words2`. We do not count this string.

Thus, there are 2 strings that appear exactly once in each of the two arrays.

#### Example 2:

**Input:** `words1 = ["b", "bb", "bbb"]`, `words2 = ["a", "aa", "aaa"]`

**Output:** 0

**Explanation:** There are no strings that appear in each of the two arrays.

#### Example 3:

**Input:** `words1 = ["a", "ab"]`, `words2 = ["a", "a", "a", "ab"]`

**Output:** 1

**Explanation:** The only string that appears exactly once in each of the two arrays is "ab".

## Input:

words1 = ["leetcode", "is", "amazing", "as", "is"],

Array <String>

words2 = ["amazing", "leetcode", "is"]

Map 1

String	Int
leetcode	1
is	2
amazing	1



String	Int
leetcode	1
is	2
amazing	1



① if value  $\neq 1 \Rightarrow$  Reject

② Iterate in Map1 and find common keys.

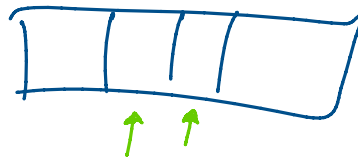
③ If we find common key, increase the counter.

Map - ?

String	Int
leetcode	1
is	1
amazing	1



String	Int
leetcode	1
is	1
amazing	1



cont..

skip

## Problem Statement:

[Rings and Rods](https://leetcode.com/problems/rings-and-rods/description/?envType=problem-list-v2&envId=hash-table)

From <<https://leetcode.com/problems/rings-and-rods/description/?envType=problem-list-v2&envId=hash-table>>

if  $n$  Rings  $\Rightarrow 2n$  Length

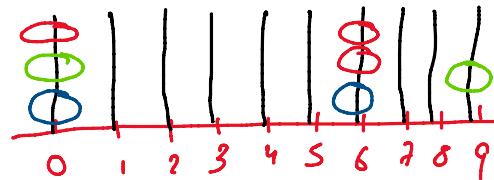
Rings  $\Rightarrow R, G, B$  (3 Rings)

Pods  $\Rightarrow 0$  to  $9$  (10 Pods)

Example  $\Rightarrow$

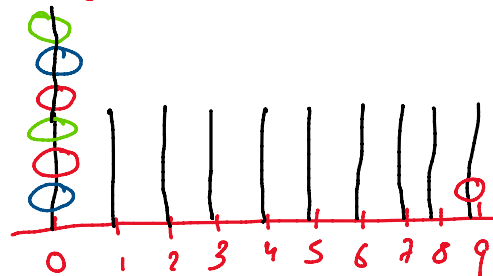
$\uparrow \uparrow \uparrow \uparrow \uparrow$

Ans  $\Rightarrow 1$



$\uparrow \uparrow \uparrow \uparrow$

Ans  $\Rightarrow 1$



key	Value			
int	Vector			
0	<table><tr><td>1</td><td>1</td><td>1</td></tr></table> Ans <small>R G B</small>	1	1	1
1	1	1		
1	<table><tr><td></td><td></td><td></td></tr></table> <small>R G B</small>			
2	<table><tr><td></td><td></td><td></td></tr></table> <small>R G B</small>			
6	<table><tr><td>2</td><td></td><td>1</td></tr></table> <small>R G B</small>	2		1
2		1		
9	<table><tr><td></td><td>1</td><td></td></tr></table> <small>R G B</small>		1	
	1			

Example - 2 ""B0B6G0R6R0R6G9"  
"  $\uparrow \uparrow \uparrow \uparrow \uparrow$

Ans[0]  $\rightarrow$  Red

Ans[1]  $\rightarrow$  Green

Ans[2]  $\rightarrow$  Blue

map < int, vector<int> > mp ;

key	Value
int	Set
0	<div><div></div><div>R G B</div></div>
1	<div><div></div><div>R G B</div></div>
2	<div><div></div><div>R G B</div></div>
6	<div><div></div><div>R G B</div></div>
9	<div><div></div><div>R G B</div></div>

'0' and '1's

5 →

A →

Now C →

0's and

0 → 48  
 1 → 49  
 2 → 50

10 → 58

$$48 - 48 = 0$$

$$50 - 48 = 2$$

$$58 - 48 = 10$$

⇒ ASCII Numbers

0 → 48

1 → 49

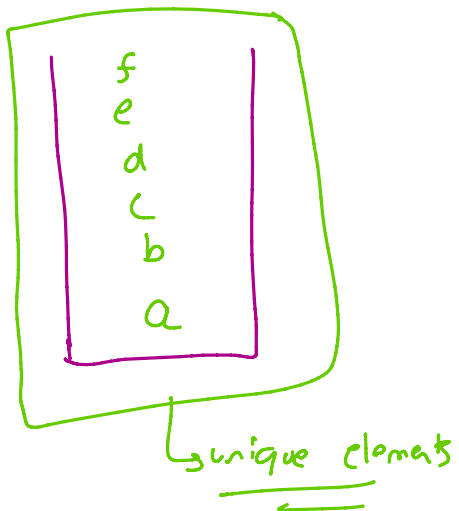
48 - '0'

$$48 - 48 = 0$$

2 → 50  
3 → 51

50 - '0'  
50 - 48 = 2

⇒ Set → A collection of only unique data.



a a a b b c c c d d e e f f  
↑ ↑ ↑ ↑ ↑ ↑ \*

'Set'

1. Create a Map<int, Set>
2. Set stores only unique elements. (We will store the rings).
3. We will check if any rod is having 3 unique elements in it.
4. That will be our answer.