GRAPHS...

video-45

"let's make it easy too"

Hard

Leetcode 2709

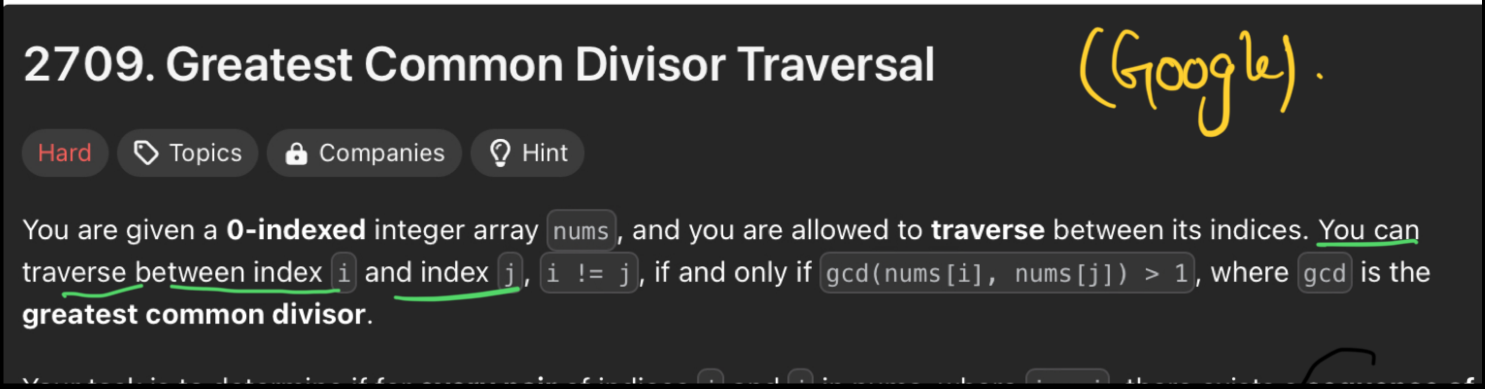
Easy

If you have tried my "Graph Concepts & Qns" playlist, these Qns, will seem very easy. Do try it once ;)

Facebook
Instagram → codestorywithMIK

(Twitter) → CSwithMIK

codestorywithMIK →



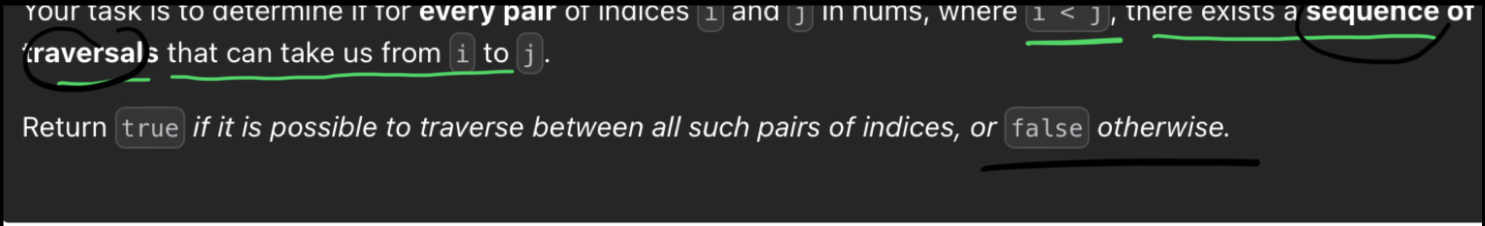
2709. Greatest Common Divisor Traversal

(Google).

Hard | Topics | Companies | Hint

You are given a **0-indexed** integer array `nums`, and you are allowed to **traverse** between its indices. You can traverse between index `i` and index `j`, `i != j`, if and only if `gcd(nums[i], nums[j]) > 1`, where `gcd` is the **greatest common divisor**.

Your task is to determine if for every pair of indices `i` and `j` in nums, where `i < j`, there exists a sequence of



Your task is to determine if for **every pair** of indices `i` and `j` in nums, where `i < j`, there exists a **sequence of traversals** that can take us from `i` to `j`.

Return `true` *if it is possible to traverse between all such pairs of indices, or* `false` *otherwise.*

Example :- nums = { $\overset{0}{2}$, $\overset{1}{3}$, $\overset{2}{6}$ }

g(6,3) = 3 >

i, j ✓✓ (0,1)

$0 \rightarrow 2 \rightarrow 1$

i, j (0, 2) ✓✓

gcd (2, 6) > 1

i, j (1,2)

↓

gcd (3, 6) ✓ > 1

Thought Process :-

(As it is).

(Brute Force)

nums = { $\overset{0}{2}$, $\overset{1}{3}$, $\overset{2}{6}$ }

{ (0,1), (0,2), (1,2) }

```
for(i=0 ; i<n; i++) {
    for(j= i+1 ; j<n; j++) {
        => gcd(nums[i], nums[j]) > 1
    }
```

True

Adj

0 → 2

2 → 0, 1

1 → 2

T.C :- O(n² * log(a)) :

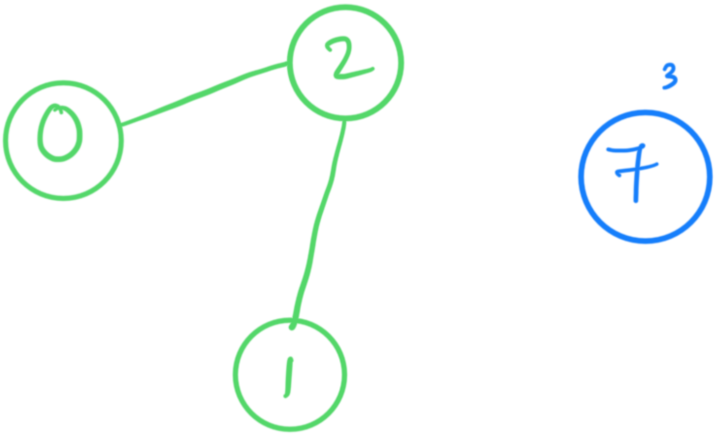
T.C: O(n × log(a))

Tip → index to index , number to number → Jump → Graph

Improvement :-

$$nums = \{\overset{0}{2}, \overset{1}{3}, \overset{2}{6}, \overset{3}{7}\}$$

0 → 2
2 → 0, 1
1 → 2



Component → DSU.

How we will use DSU?

→ "First, go to my "Graph Concepts" Playlist & study DSU."

$$nums = \{\overset{0}{2}, \overset{1}{3}, \overset{2}{6}\}$$

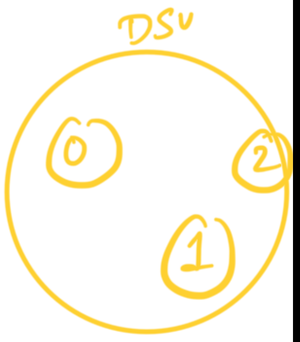
2 → 2

3 → 3

6 → 2 * 3

Factors	number
2 →	{2, 6}
3 →	{3, 6}

DSU



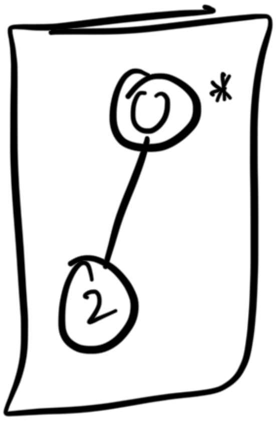
factor	index
2 →	0, 2
3 →	1, 2

$$nums = \{\overset{0}{12}, \overset{1}{13}, \underset{i}{\overset{2}{16}}\}$$

$12 \rightarrow (2 * 2 * 3)$

$\cancel{6}\ 3/3 = 1$

Factor	idx
$2 \rightarrow$	$\underline{0}$
$3 \rightarrow$	0



2

$13 \rightarrow (1)$

$16/2 \rightarrow \{2 * 2 * 2 * 2\}$

$8/2 = 4/2 = 2/2 = 1$

Story Points:

① i = 0 to n-1 → iterate.

② nums[i] → prime factor.

↓ Check already.

Yes → Union (j, i)

```
for (factor = 2 ; factor * factor <= nums[i];
                         fact ++) {

        if (nums[i] % factor != 0)
                   contin;

        if (mp.find(fact) == mp.end()){
              mp[factor] = i ;
        } else{
            Union(mp[factor] , i);
        }

Simplify. // while (nums[i] % factor == 0) {

                nums[i] = nums[i] / factor;

         }
}

     if (nums[i] > 1) {
          if( mp.e
            mp [nums[i]] = i ;
          el.
            Unio( idx, i);
     }

  if (cnt == 1

       re Tru;
```

map.

ur. →

$$\left\{ \begin{array}{ccc} 0 & 1 & 2 \\ 2, & \underset{i}{3,} & 6 \end{array} \right\}$$

map:

$2 \to 0^{id}$

$\boxed{3} \to \underset{\uparrow \text{idx}}{1}$