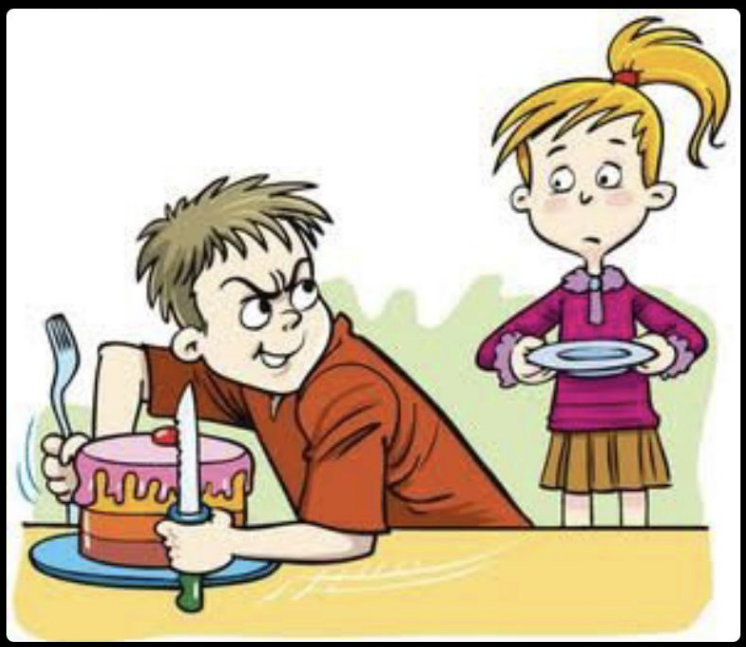
GREEDY

codestorywithMIK



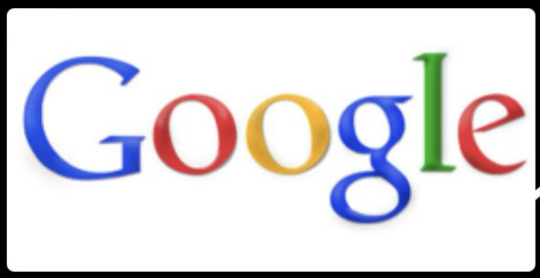
VIDEO-15

LEETCODE
-2366
~~Hard~~

Easy
बना देंगे

Minimum Replacements To Sort The Array

Company Tags :-



2366. Minimum Replacements to Sort the Array

Hard 760 26 Add to List Share

You are given a **0-indexed** integer array `nums`. In one operation you can replace any element of the array with **any two** elements that **sum** to it.

- For example, consider `nums = [5,6,7]`. In one operation, we can replace `nums[1]` with `2` and `4` and convert `nums` to `[5,2,4,7]`.

Return *the minimum number of operations to make an array that is sorted in* ***non-decreasing*** *order.*

Example :- nums = [3,2,3,1,2,3,4,6,17,19,21]

Output = 6

3,3,3,3

1+1+1 oprn = 1+2+3 = 6

Some Facts :-

Fact-1

Some Facts :-

Fact-1

12 → 4, 3, 5

1 opr
1 opr.

operations = (parts - 1);

Fact-2

{ 1, 2, 11, 3 }

i i+1

2, 3, 3, 3

→ Parts = nums[i] / nums[i+1] <= 3

11/3 = 3

if (nums[i] % nums[i+1] != 0)
  Parts ++;

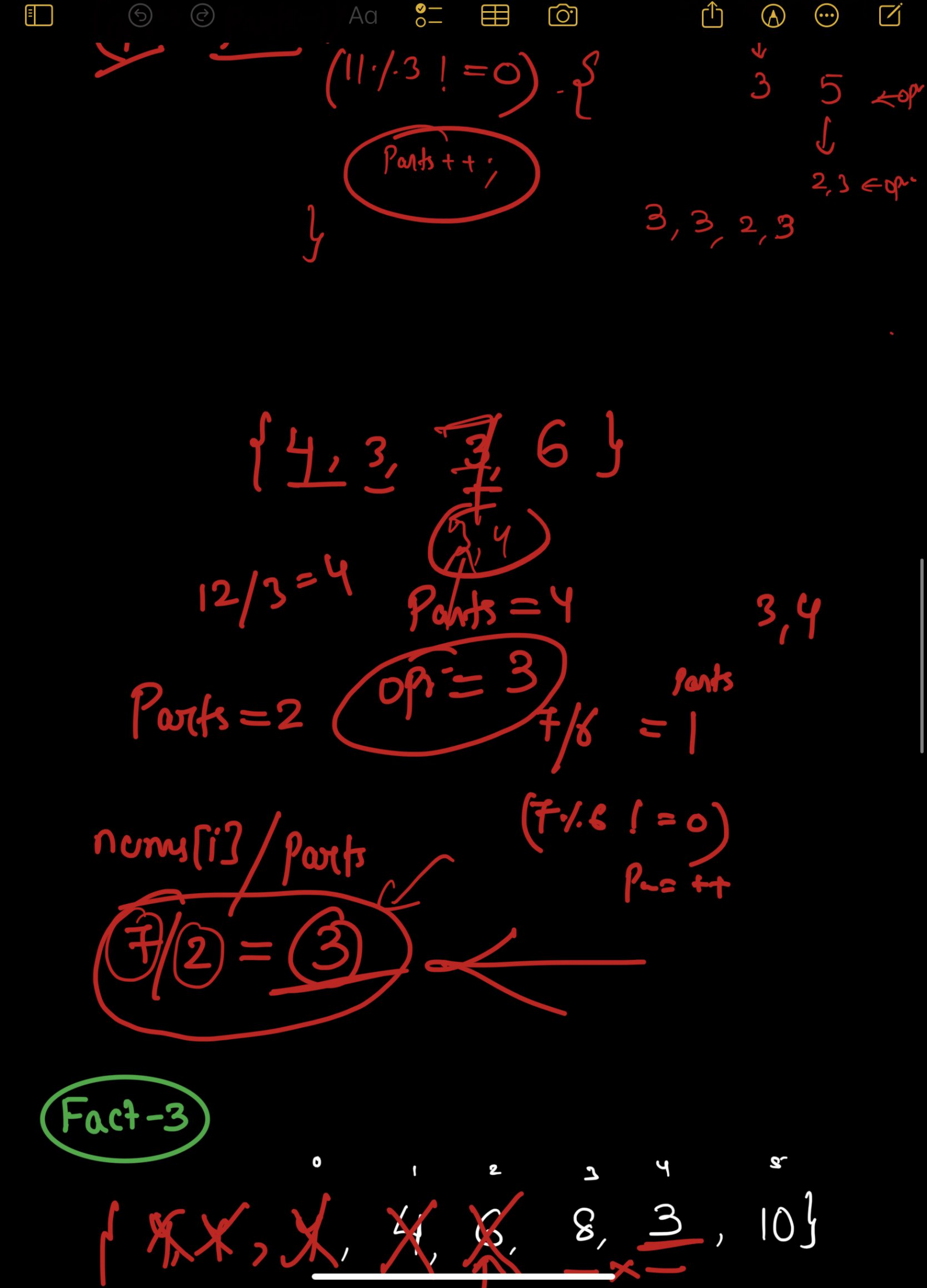
Parts

11/3 = 3

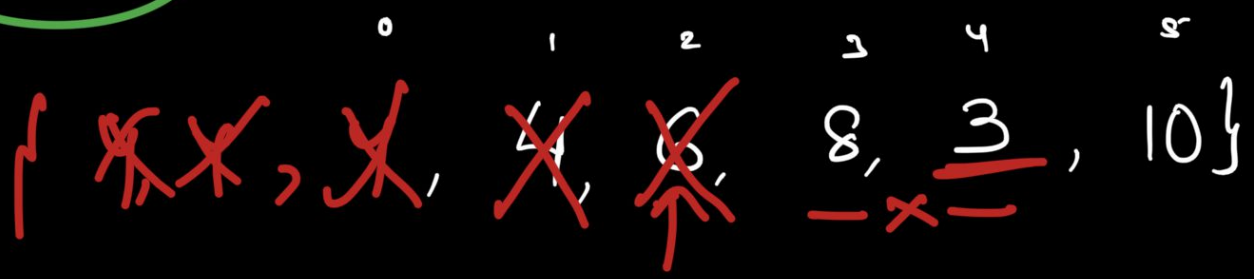
(oprns = Parts - 1)

(11 % 3 != 0) - F

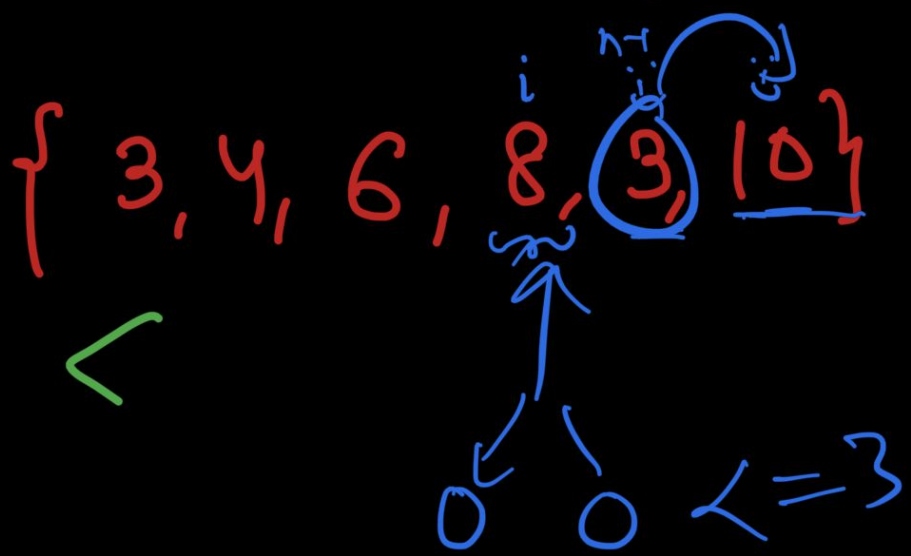
3 8 ← 1 opn
↓
3 5 ← opn


(11%3 != 0) {
parts++;
}
3
5
2,3
3, 3, 2, 3
{4, 3, 3, 6}
12/3=4
parts=4
3, 4
parts=2
opr=3
parts
7/6 = 1
(7%6 != 0)
p-= ++
nums[i]/parts
7/2 = 3
Fact-3
0 1 2 3 4 5
{ X, X, 4, 6, 8, 3, 10 }

Fact-3



x, y <= 3



```
for(i = n-2 ; i>=0 ; i--) {
    if (nums[i]<=nums[i+1] )
        continue ;
```

```
for(i = n-2 ; i>=0 ; i--) {
    if (nums[i]<=nums[i+1] )
        continue;
    parts = nums[i]/nums[i+1];
    if (nums[i] % nums[i+1] !=0)
        parts++;
    operations += (parts-1);
    nums[i] = nums[i]/parts;
}
return
```

O(n)