Todays Content
  1. Re-arrange arr[]
  2. Count frequy of all elements.

Few Basics:

  1. Say $a > 0$ & $a < M$:
        $a \% M = a;$ Ex: $5 \% 7 = 5$ $2 \% 10 = 2$
        $a / M = 0;$ Ex: $5/7 = 0$ $2/3 = 0$

Modular:

Given an ar[N] re-arrange array value, such that ar[i] = ar[ar[i]]

Note: all array elements are unique.

Constraints:

$1 <= N <= 10^5$

$0 <= ar[i] < N$

	0	1	2	3	4	5	6	7
#Eg, ar[N] = {	6	4	2	1	7	0	3	5 }

ar[i] = ar[ar[i]]

b[) {	3	7	2	4	5	6	1	0 }

Ideal:

1. Create b[N];   TC: O(N+N) = O(N) SC: O(N)
2. i=0; i < N; i++) {
      b[i] = ar[ar[i]]

3. Copy b[) → ar[)
      i=0; i < N; i++) {
         ar[i] = b[i]
      }

Idea2: Iterate m ar[) & make ar[i] = ar[ar[i]]

	0	1	2	3	4	5	6	7
ar[N] = {	6̶	4̶	2̶	1̶	7̶	0̶	3	5 }
	3	7	2	7	5	3	7	3

Issue: If we update ar[i] = ar[ar[i]] we will end up loosing data

Catch: We need old & new value, without extra space?

#Idea: No Extra Space

#Hrs Since Bing Bang	Days	Time
100 hrs	4	4hrs = 4 am
40 hrs	1	16hrs = 4 pm
78 hrs	3	6hrs = 6 am
n hrs	n/24	n%24

obs: In n, Both day & Time are present

$n/24$ = Days $n\%24$ = Time

Hint: $0 <= ar[i] < N$ & $[0 .. N-1]$ & Specality $\%N = [0.. N-1]$

In $ar[i]$: Store old & new value together

Case 1:

$ar[i] \to ar[i] \% N = new : rem$

$ar[i] \to ar[i]/N = old : Bno$

$$\boxed{ar[i] = old * N + new}$$

$ar[i]/N = (old * N + new)/N$

$= (old * N/N + new/N)$

$= (old + 0)$

$= old$

$ar[i]\%N = (old * N + new) \% N$

$= (old * N \% N + new \% N) \% N$

$= (0 + new \% N) \% N$

$= (new) \% N = new$

Case 2:

$ar[i] \to ar[i] \% N = old : rem$

$ar[i] \to ar[i]/N = new : Bno$

$$\boxed{ar[i] = N * new + old}$$

$ar[i]/N = (N * new + old)/N$

$= (N * new/N + old/N)$

$= (new + 0)$

$= new$

$ar[i]\%N = (N * new + old) \% N$

$= ((N * new) \% N + old \% N) \% N$

$= (0 + old \% N) \% N$

$= (old \% N)$

$= old$

Approach : $ar[i] = old * N + new$

	0	1	2	3	4	5	6	7
$ar[N] = \{$	6	4	2	1	7	0	3	5 $\}$

#Step1: Iterate in ar[] multiply by N

	0	1	2	3	4	5	6	7
$ar[N] = \{$	$6*8$	$4*8$	$2*8$	$1*8$	$7*8$	$0*8$	$3*8$	$5*8 \}$

#Step2: Iterate in ar[] & add new value to all elements

	0	1	2	3	4	5	6	7
$ar[N] = \{$	$6*8+3$	$4*8+7$	$2*8+2$	$1*8+4$	$7*8+5$	$0*8+6$	$3*8+1$	$5*8+0 \}$
	i							

$$ar[0] += ar[\,ar[0]/N\,] = ar[6]/N = 3$$
$$ar[1] += ar[\,ar[1]/N\,] = ar[4]/N = 7$$
$$ar[2] += ar[\,ar[2]/N\,] = ar[2]/N = 2$$
$$ar[3] += ar[\,ar[3]/N\,] = ar[1]/N = 6$$
$$ar[i] += ar[\,ar[i]/N\,]/N$$

#Step3: Iterate in ar[] & update $ar[i] = ar[i] \% N$

	0	1	2	3	4	5	6	7
$ar[N] = \{$	$6*8+3$	$4*8+7$	$2*8+2$	$1*8+4$	$7*8+5$	$0*8+6$	$3*8+1$	$5*8+0 \}$

	0	1	2	3	4	5	6	7
$ar[N] = \{$	3	7	2	4	5	6	1	0 $\}$

```
vector<int> re-arrange(vector<int> raw){ TC: O(N) SC: O(1)

    TODD
```

3

3Q Given ar[N]: Print frequny of all numbers from [0..N-1]

Constraints:

   $0 \leq ar[i] < N$

   $\qquad\qquad\quad 0\quad 1\quad 2\quad 3\quad 4$

En: ar[5] = { 2   4   4   0   3 }

Output

   <mark>0 : 1</mark>

   <mark>1 : 0</mark>

   <mark>2 : 1</mark>

   <mark>3 : 1</mark>

   <mark>4 : 2</mark>

Idea1: For every element from [0.. N-1]:

   Iterate m array q get frequency q print it.

   TC: $O(N*N) = O(N^2)$ SC: $O(1)$

Idea2: At ar[i] : Stre ele q it's freq information.

Way1: ar[i] ⟨ $ar[i]/N = ele$; Qno

$\qquad\qquad\qquad$ <mark>$ar[i] = N*ele + freq$</mark>

$\qquad\qquad$ $ar[i] \% N = freq$; rem

$ar[i] \% N = (N*ele + freq) \% N$

$\qquad = ((N*ele)\%N + freq\%N)\%N$

$\qquad (0 + freq\%N)\%N$

$\qquad (freq\%N)$: Issue: At max freq == N. If all ele are same

$\qquad\qquad\qquad$ if freq == N: $(freq\%N) = 0$

Way2: ar[i] $\begin{cases} \rightarrow ar[i]/N = freq : Qw \\ \rightarrow ar[i]\%N = ele \quad : rem \end{cases}$

$\boxed{ar[i] = freq * N + ele}$

ar[i] % N

$(freq * N + ele)\%N$

$((freq*N)\%N + ele\%N)\%N$

$= (0 + ele\%N)\%N$

$= (ele)\%N$

$= ele$

ar[i] / N

$(freq * N + ele)/N$

$(freq*N/N + ele/N)$

$(freq + 0)$

$= freq$

#DryRun $\boxed{ar[i] = freq * N + ele}$

	0	1	2	3	4	5	6	7
En: ar[8] = {	2	4	4	5	2	3	5	2 }

Step1: Iterate ar[] & stre freq

	0	1	2	3	4	5	6	7
En: ar[8] = {	2	4	4+3N	5+N	2+2N	3+2N	5	2 }

$i = 0; i < N; i++\{$
$\quad ar[ar[i]\%N] += N$
$\}$

Step2: Iterate in ar[] & print freq

	0	1	2	3	4	5	6	7
En: ar[8] = {	2̶ 0	4̶ 0	4̶+3N	5̶+N	2̶+2N	3̶+2N	5̶	2̶ }

	0	1	2	3	4	5	6	7
ar[] = {	0	0	3	1	2	2	0	0 }

$i = 0; i < N; i++\{$
$\quad ar[i] = ar[i]/N.$
$\}$