Cycle Sort

* ## Cycle Sort :—

>> It is a comparison - based sorting algorithm.

>>> used → when given nos. from range 1 to N.

Problems like :

Q. Find the missing number, where given nos. from range 1 to N.

Q. Find the duplicate no. , where given nos from range 1 to N.

>>> Cycle sort solve/sort the array in single pass only . i.e, in one for loop.

Example

$$\overset{0}{3}, \overset{1}{5}, \overset{2}{2}, \overset{3}{1}, \overset{4}{4}$$ ← Here, Numbers are from 1 to 5

↓ After sorting

$$\overset{0}{1}, \overset{1}{2}, \overset{2}{3}, \overset{3}{4}, \overset{4}{5}$$

→ Here, we can see that after sorting index = value − 1 because, index start from 0

NOTE **

* If range = [0, N]
  Then, Every element will be at index = value.

* If range = [1, N]
  Then, Every element will be at index = value − 1.

* **Algorithm :**

  Example :-

  $$\overset{0}{3}, \overset{1}{5}, \overset{2}{2}, \overset{3}{1}, \overset{4}{4}$$

  ↑
  start

  Here, Index start from 0.

>>> check, is 3 at its correct index?

  index = Value -1
  = 3-1
  = 2

  But, Here, 3 is at index 0
  So, swap 3 ~~with to~~ its correct index .i.e **index 2**

  $$\overset{0}{③}, \overset{1}{5}, \overset{2}{2}, \overset{3}{1}, \overset{4}{4}$$

  swap with index 2

  ⇓

  $$\overset{0}{②}, \overset{1}{5}, \overset{2}{3}, \overset{3}{1}, \overset{4}{4}$$ ←

  swap with index 1

  check → is 2 at its correct index?
  Here 2 should be at index 1
  So, swap with index 1.

  ⇓

  $$\overset{0}{⑤}, \overset{1}{2}, \overset{2}{3}, \overset{3}{1}, \overset{4}{4}$$

  swap with index 4

  check → is 5 at it's correct index?

  ⇓

  $$\overset{0}{④}, \overset{1}{2}, \overset{2}{3}, \overset{3}{1}, \overset{4}{5}$$

  swap with index 3

  check → is 4 at its correct index?

  ⇓

  $$\overset{0}{1}, \overset{1}{2}, \overset{2}{3}, \overset{3}{4}, \overset{4}{5}$$

$$\overset{0}{1}, \overset{1}{2}, \overset{2}{3}, \overset{3}{4}, \overset{4}{5}$$

Now, check → Is 1 at its correct index?
⇒ index = value -1 ⇒ 0 = 1-1 ⇒ Yes, It is......

>>> check → Is 2 at its correct index?
⇒ index = value -1 ⇒ 1 = 2-1 ⇒ Yes, It is ----

>> check → Is 3 at its correct index!
⇒ index = value -1 ⇒ 2 = 3-1 ⇒ Yes, It is ---

>>> check → Is 4 at its correct index?
⇒ index = value -1 ⇒ 3 = 4-1 ⇒ Yes, It is ----

>>> check → Is 5 at its correct index?
⇒ index = value -1 ⇒ 4 = 5-1 ⇒ Yes, It is ---

So, Loop Over

ans ⇒ $\overset{0}{1}, \overset{1}{2}, \overset{2}{3}, \overset{3}{4}, \overset{4}{5}$

* Time Complexity :

Worst case :-
In above Example
Total Swaps = 4 + 5
    or
comparisons

$$= (N-1) + N$$
$$= 2N-1 \qquad // \text{Ignore constants}$$

Worst case complexity $= O(N)$