

1

Given an array of size N containing only 0s, 1s, and 2s; sort the array in ascending order.

Example 1:

Input:

N = 5

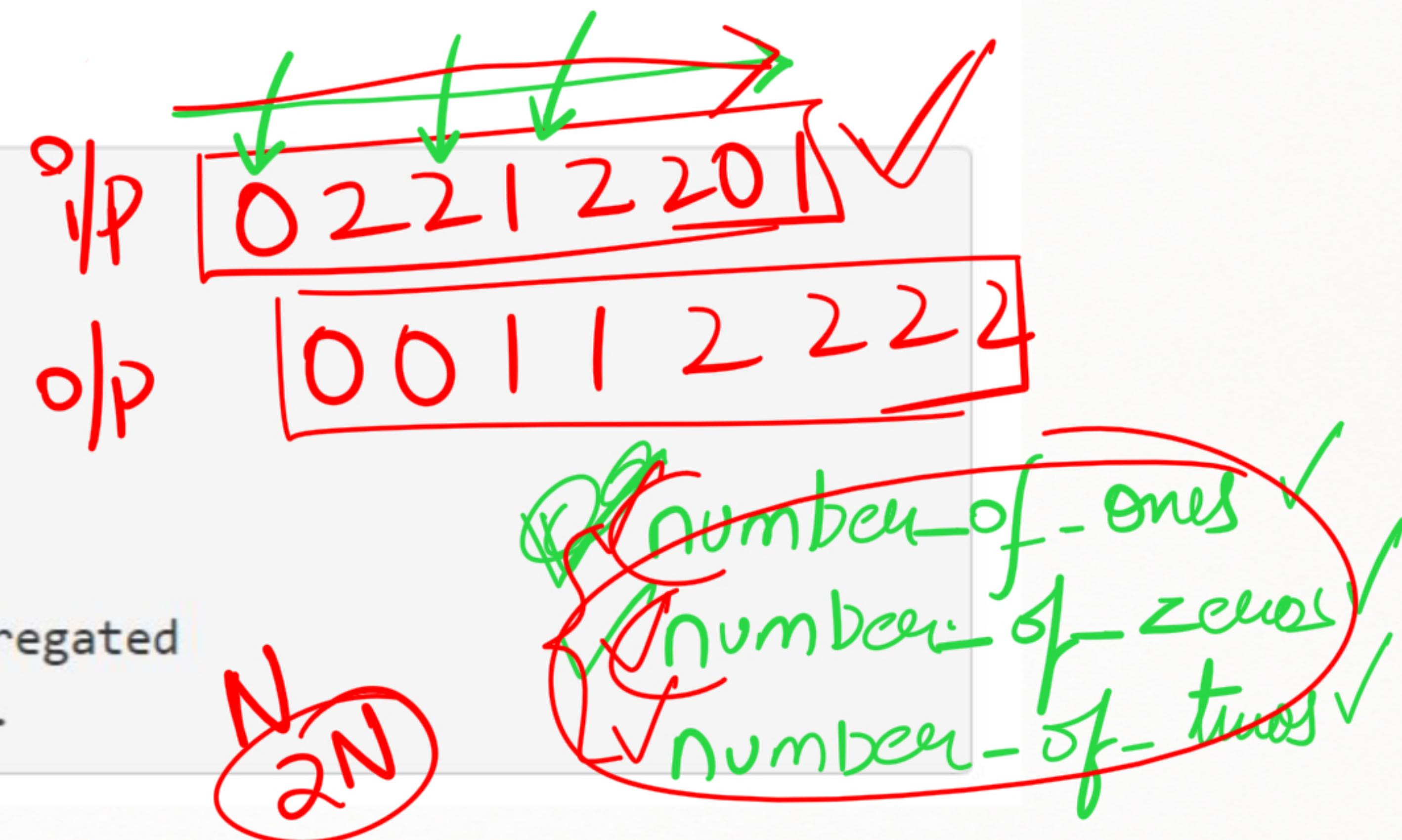
arr[] = {0 2 1 2 0}

Output:

0 0 1 2 2

Explanation:

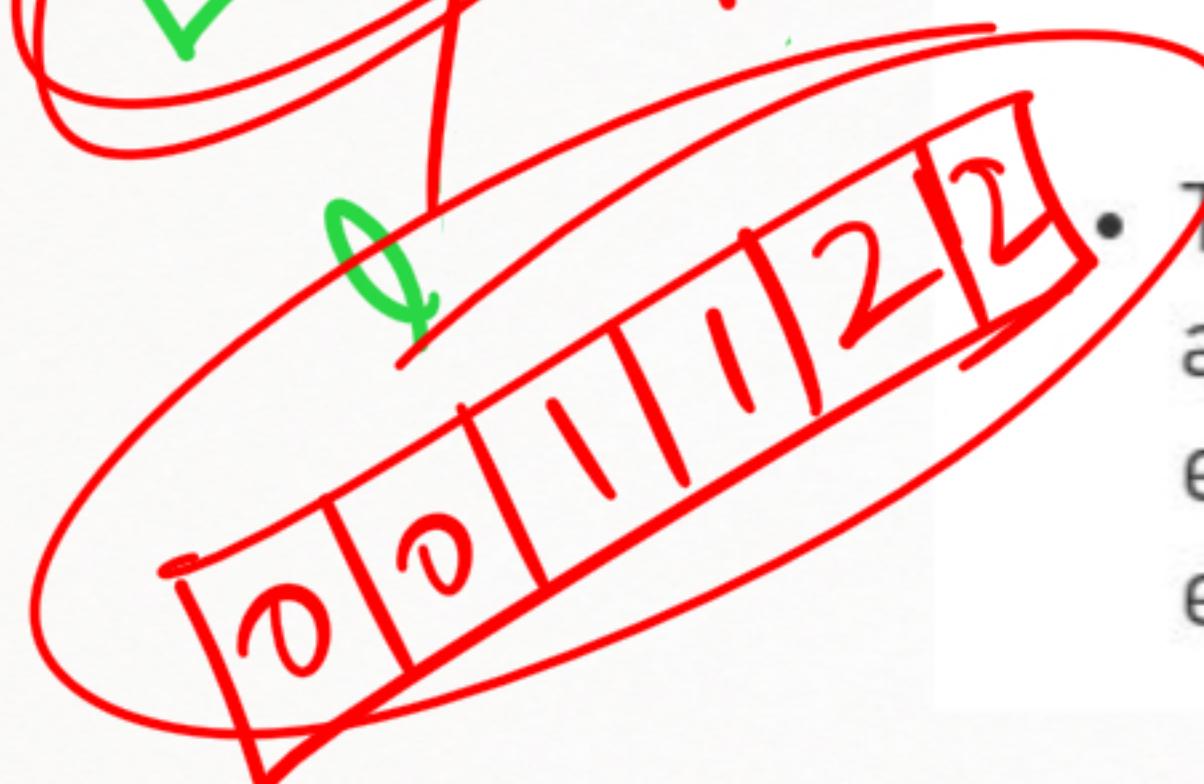
0s 1s and 2s are segregated into ascending order.



Pseudocode

These are the following steps:

- Maintain 3 variables low, high and mid
 - low - all elements before low are 0 ✓
 - mid - all elements between low and mid are 1 ✓
 - high - all elements after high are 2 ✓
- Initially low, mid are set at 0 and high is at n-1
- Now, we iterate mid from 0 to high, and for every element
 - if it is equal to 0, we swap it with element at low, and increement low and mid
 - else if it is equal to 2, we swap it with element at high, and decreement high
 - else we just increement mid (i.e element is equal to 1)
- This method ensures partition, as low and high maintain elements according to their values, and then change their positions, ensuring all elements before low are lower than low_value and all elements after high are higher than high_value



In place

~~Space~~

2

Given an array of size $N-1$ such that it only contains distinct integers in the range of 1 to N . Find the missing element.

Example 1:

Input:
 $N = 5$
 $A[] = \{1, 2, 3, 5\}$
Output: 4

does not
contain
duplicates

Example 2:

Input:
 $N = 10$
 $A[] = \{6, 1, 2, 8, 3, 4, 7, 10, 5\}$
Output 9

First N
natural
Numbers

$$\frac{n(n+1)}{2} = \frac{10(10+1)}{2} = 55$$

③

Delete first occurrence
in the doubly linked list
given

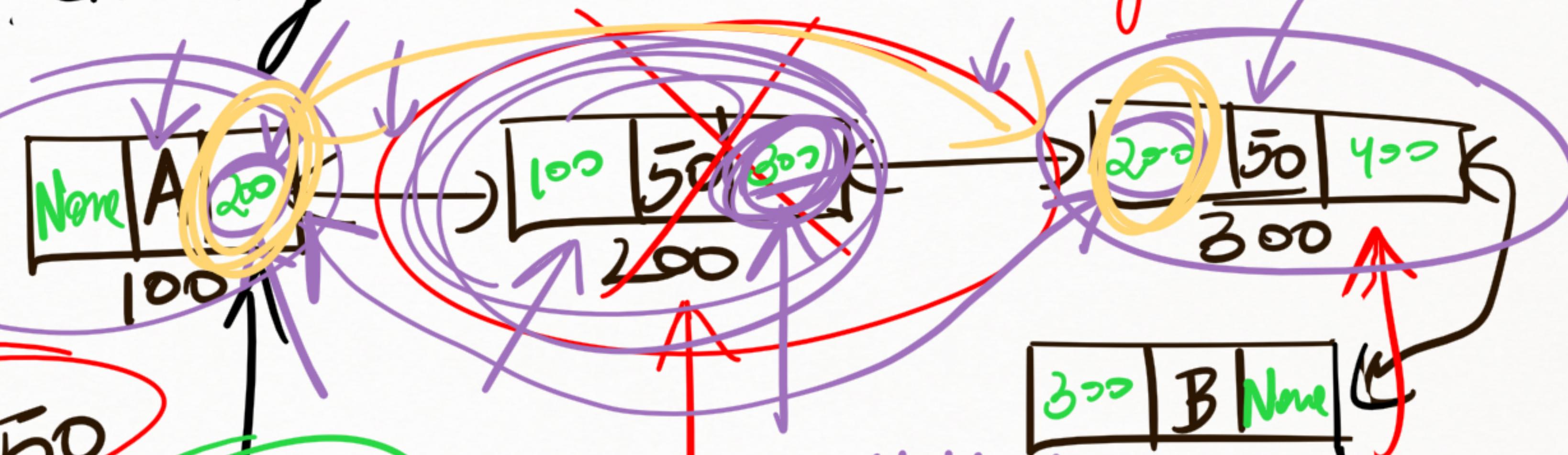
Example

tempP

Delete 50

temp.next.next.next =
temp

Head



temp = self. Head 400

while temp.next.data != kg
temp = temp.next

④ Delete the given element in doubly linked list with the help of tail. First occurrence needs to be deleted if duplication.

Homework
★ ★

