

Segregate 0 1 2

```
class Node:
    def __init__(self, val):
        self.data = val
        self.next = None

def segregate012(head):
    dummyOne = Node(-1)
    dummyZero = Node(-1)
    dummyTwo = Node(-1)
    prevOne = dummyOne
    prevZero = dummyZero
    prevTwo = dummyTwo
    curr = head
    while curr is not None:
        temp = curr.data
        if temp == 0:
            prevZero.next = curr
            prevZero = curr
            curr = curr.next
        elif temp == 1:
            prevOne.next = curr
            prevOne = curr
            curr = curr.next
        else:
            prevTwo.next = curr
            prevTwo = curr
            curr = curr.next
    if prevZero.next != None:
        prevZero.next = None
    if prevOne.next != None:
        prevOne.next = None
    if prevTwo.next != None:
        prevTwo.next = None
    prevZero.next = dummyOne.next
    prevOne.next = dummyTwo.next
    return dummyZero.next
```

```
head1 = Node(2)
head1.next = Node(1)
# head1.next.next = Node(1)
# head1.next.next.next = Node(1)
# head1.next.next.next.next = Node(2)
# head1.next.next.next.next.next = Node(1)
# head1.next.next.next.next.next.next = Node(0)

# head2 = Node(7)
# head2.next = Node(8)
# head2.next.next = Node(9)

head = segregate012(head1, )

while head != None:
    print(head.data, end=' ')
    head = head.next
```