

Date : 09/11/2023

1. Given a list of numbers, reverse the list

TC : $O(n)$

SC : $O(1)$

```
def reverseNumbers(nums):
```

```
    if len(nums) == 0:
```

```
        return []
```

```
    left = 0
```

```
    right = len(nums) - 1
```

```
    while left < right:
```

```
        temp = nums[left]
```

```
        nums[left] = nums[right]
```

```
        nums[right] = temp
```

```
        left += 1
```

```
        right -= 1
```

```
    return nums
```

```
nums = [2,3,4,4,5]
```

```
print(reverseNumbers(nums))
```

Test Cases:

TC1:

input = [2,3,4,5,6]

output = [6,5,3,2]

TC2:

input = []

output = []

TC3:

input = [1]

output = [1]

=====

2. Given an array with integer numbers between 1 and 100, except 1 integer, find the missing integer

TC = $O(n)$

```

SC = O(n)
def findMissingNum(nums):
    num_dict = {}
    numLen = 3
    res = -1
    for n in nums:
        if n not in num_dict:
            num_dict[n] = n
    print(num_dict)
    for n in range(1,numLen):
        if n not in num_dict:
            res = n
    return res

nums = [1]
print(findMissingNum(nums))

```

Test Cases:

TC 1:

input : [1,2,3,4]

output : [5]

TC2:

input : []

output : -1

TC3:

input : [1]

output : -1 if nums = [1]

=====

3. Given a list of numbers , find the third largest number

TC : O(n)

SC : O(1)

import heapq

```
def findKLargestNum(nums):
```

```
    if len(nums) == 0 or len(nums) == 1 or len(nums) == 2:
```

```
        return -1
```

```
    heapq.heapify(nums)
```

```
    return nums[2]
```

```
nums = [1,2,1]
```

```
print(findKLargestNum(nums))
```

Test Cases:

TC1 :

input : [2,1,3,4,5]

output : [3]

TC2:

input : []

output : []

TC3 :

input : [1]

output : [1]