**Problem :Given an array, rotate the array to the right by k steps, where k is non-negative.**

**Example 1:**

**Input:** nums = [1,2,3,4,5,6,7], k = 3

**Output:** [5,6,7,1,2,3,4]

**Explanation:**

rotate 1 steps to the right: [7,1,2,3,4,5,6]

rotate 2 steps to the right: [6,7,1,2,3,4,5]

rotate 3 steps to the right: [5,6,7,1,2,3,4]

**Example 2:**

**Input:** nums = [-1,-100,3,99], k = 2

**Output:** [3,99,-1,-100]

**Explanation:**

rotate 1 steps to the right: [99,-1,-100,3]

rotate 2 steps to the right: [3,99,-1,-100]

**Constraints:**

* 1 <= nums.length <= 105
* -231 <= nums[i] <= 231 - 1
* 0 <= k <= 105

class Solution {

public:

void reverse(vector<int>& nums, int i, int j) {

while(i<j) {

swap(nums[i++], nums[j--]);

}

}

void rotate(vector<int>& nums, int k) {

k = k % nums.size();

reverse(nums, 0, nums.size()-1);

reverse(nums, 0, k-1);

reverse(nums, k, nums.size()-1);

}

};