Third Maximum Number

# Question

Given integer array nums, return the third maximum number in this array. If the third maximum does not exist, return the maximum number.

**Example 1:**

Input: nums = [3,2,1]

Output: 1

Explanation: The third maximum is 1.

**Example 2:**

Input: nums = [1,2]

Output: 2

Explanation: The third maximum does not exist, so the maximum (2) is returned instead.

**Example 3:**

Input: nums = [2,2,3,1]

Output: 1

Explanation: Note that the third maximum here means the third maximum distinct number.

Both numbers with value 2 are both considered as the second maximum.

# Pseudo Code

## v1.0

Sort the given Array by using the Appropriate Sorting Algorithm

Remove the duplicates from the new Array

If the new Length of the Array is less than 2

Return arr[length]

Else

Return arr[length - 2]

## 

## v2.0

If size == 1

Return nums[size - 1]

Declare three Variables greater than “Integer” and initialize them to their minimum value.

Declare and initialize counter variable to 0

Run the First For Loop i = 0 to size

If maxOne < nums[i]

maxOne = nums[i]

Run the Second For Loop i = 0 to size

If nums[i] < maxOne and nums[i] >= maxTwo

maxTwo = nums[i]

Run the Third For Loop i = 0 to size

If nums[i] < maxOne and nums[i] < maxTwo and nums[i] >= maxThree

maxThree = nums[i]

counter++

If counter > 0

Return maxThree

Else

Return maxOne

# 

# Source Code

## v1.0 (Language C)

1. #define THRICE 3
2. int thirdMax(int\* nums, int numsSize) {
4. int tempVar = 0;
5. for(int i=0 ; i<(numsSize - 1) ; i++) {
7. for(int j=0 ; j<(numsSize - i - 1) ; j++) {
9. if(nums[j] > nums[j + 1]) {
10. tempVar = nums[j];
11. nums[j] = nums[j + 1];
12. nums[j + 1] = tempVar;
13. }
14. }
15. }
17. int length = 0, pointerTwo = 0;
18. for(int i=0 ; i<(numsSize - 1) ; i++) {
20. if(nums[i] == nums[i + 1]) {
21. pointerTwo++;
22. }
23. else {
24. nums[++length] = nums[++pointerTwo];
25. }
26. }
28. if(length < 2)
29. return nums[length];
31. else
32. return nums[length - 2];
33. }

## v2.0(Language C)

1. int thirdMax(int\* nums, int numsSize){
3. if(numsSize == 1)
4. return nums[numsSize - 1];

7. long maxValueOne = LONG\_MIN, maxValueTwo = LONG\_MIN, maxValueThree = LONG\_MIN;
8. int counter = 0;
10. for(int i=0 ; i<numsSize ; i++) {
11. if(maxValueOne < nums[i]) {
12. maxValueOne = nums[i];
13. }
14. }
16. for(int i=0 ; i<numsSize ; i++) {
17. if(nums[i] < maxValueOne && nums[i] >= maxValueTwo)
18. maxValueTwo = nums[i];
19. }
21. for(int i=0 ; i<numsSize ; i++) {
22. if(nums[i] < maxValueOne && nums[i] < maxValueTwo && nums[i] >= maxValueThree) {
23. maxValueThree = nums[i];
24. counter++;
25. }
26. }
28. if(counter > 0)
29. return maxValueThree;
31. else
32. return maxValueOne;
33. }