2966. Divide Array Into Arrays With Max Difference

Solved

Medium | Topics | Companies | Hint

You are given an integer array `nums` of size `n` and a positive integer `k`.

Divide the array into one or more arrays of size `3` satisfying the following conditions:

- **Each** element of `nums` should be in **exactly** one array.
- The difference between **any** two elements in one array is less than or equal to `k`.

Return *a* **2D** *array containing all the arrays. If it is impossible to satisfy the conditions, return an empty array. And if there are multiple answers, return* ***any*** *of them.*

```
class Solution {
    func divideArray1(_ nums: [Int], _ k: Int) -> [[Int]] {
        var aa = nums.sorted()
        var ans = [[Int]]()
        var count = 0
        var temp = [Int]()
        var a: Int = 0
        var b: Int = 0
        for i in aa {
            if(count == 0) {
                temp.append(i)
                a = i
                count = 1
            } else {
```

```
            if(count == 1) {
                if(i-a <= k) {
                    temp.append(i)
                    b = i
                } else {
                    return []
                }
                count = 2
            } else {
                if(i - b <= k && i - a <= k) {
                    temp.append(i)
                    ans.append(temp)
                    temp = []
                    count = 0
                } else {
                    return[]
                }
            }
        }
        // print(ans)
    }

    return ans
}

func divideArray(_ nums: [Int], _ k: Int) -> [[Int]] {
    var aa = nums.sorted()
    var ans = [[Int]]()
    print(aa)
    var i = 0
    while (i < nums.count) {
        if(aa[i+2] - aa[i] <= k ) {
            ans.append([aa[i], aa[i+1], aa[i+2]])
        } else {
            return []
        }
```

```
            i = i + 3
        }

        return ans
    }
}
```