Date: 8.1.2024

Session Topic: Arrays

Task 1

Question:

Running Sum of 1d Array

Solution:

```
class Solution {
    public int[] runningSum(int[] nums) {
        int arr[]=new int[nums.length];
        arr[0]=nums[0];
        for(int i=0;i<nums.length-1;i++)
        {
            int sum=0;
            for(int j=i+1;j>=0;j--)
            {
                sum+=nums[j];
            }
            arr[i+1]=sum;
        }
        return arr;
    }
}
```

Date: 8.1.2024

Session Topic: Arrays

Task 2

Question:

Number of Good Pairs

Solution:

```
class Solution {
    public int numIdenticalPairs(int[] nums) {
        int count=0;
```

```
        for(int i=0;i<nums.length;i++)
        {
            for(int j=i+1;j<nums.length;j++)
            {
                if(nums[i]==nums[j] && i < j)
                {
                    count++;
                }
            }
        }
        return count;
    }
}
```

Date: 8.1.2024

Session Topic: Arrays

Task 3

Question:

Find Greatest Common Divisor of Array

Solution:

```
class Solution {
    public int findGCD(int[] nums) {
        Arrays.sort(nums);
        int num=Integer.MAX_VALUE;
        for(int i=1;i<=nums[nums.length-1];i++)
        {
            if(nums[0]%i==0 && nums[nums.length-1]%i==0)
            {
                num=i;
            }
        }
        return num;
    }
}
```

Date: 8.1.2024

Session Topic: Arrays

Task 4

Question:

Unique Number of Occurrences

Solution:

```
class Solution {
    public boolean uniqueOccurrences(int[] arr) {
        Map<Integer,Integer>map=new HashMap<>();
        List<Integer>list=new ArrayList<>();
        for(int i=0;i<arr.length;i++)
        {
            map.put(arr[i],map.getOrDefault(arr[i],0)+1);
        }
        for(Map.Entry<Integer,Integer>i:map.entrySet())
        {

            if(!list.contains(i.getValue()))
            {
                list.add(i.getValue());
            }
            else
            {
                return false;
            }
        }
        return true;
    }
}
```

Date: 8.1.2024

Session Topic: Arrays

Task 5

Question:

Divide Array Into Equal Pairs

Solution:

```
class Solution {
    public boolean divideArray(int[] nums) {
        Arrays.sort(nums);
        if(nums.length%2==1)
```

```
        {
            return false;
        }
        for(int i=0;i<nums.length-1;i=i+2)
        {
            if(nums[i]!=nums[i+1])
            {
                return false;
            }
        }
        return true;
    }
}
```

Date: 8.1.2024

Session Topic: Arrays

Task 6

Question:

Find the Duplicate Number

Solution:

```
class Solution {
    public int findDuplicate(int[] nums) {
        Map<Integer,Integer>map=new HashMap<>();
        for(int i=0;i<nums.length;i++)
        {
            map.put(nums[i],map.getOrDefault(nums[i],0)+1);
        }
        for(Map.Entry<Integer,Integer>i:map.entrySet())
        {
            if(i.getValue()>=2)
            {
                return i.getKey();
            }
        }
        return 0;
    }
}
```

Date: 8.1.2024

Session Topic: Arrays

Task 7

Question:

Find All Duplicates in an Array
Solution:

```
class Solution {
    public List<Integer> findDuplicates(int[] nums) {
        List<Integer>list=new ArrayList<>();
        Map<Integer,Integer>map=new HashMap<>();
        for(int i=0;i<nums.length;i++)
        {
            map.put(nums[i],map.getOrDefault(nums[i],0)+1);
        }
        for(Map.Entry<Integer,Integer>i:map.entrySet())
        {
            if(i.getValue()>1)
            {
                list.add(i.getKey());
            }
        }
        return list;
    }
}
```

Date: 8.1.2024

Session Topic: Arrays

Task 8

Question:

Find Peak Element
Solution:

```
class Solution {
    public int findPeakElement(int[] nums) {
        int max=Integer.MIN_VALUE,index=0;
        for(int i=0;i<nums.length;i++)
        {
            if(nums[i]>max)
```

```
            {
                max=nums[i];
                System.out.print(max);
                index=i;
            }
        }
        return index;
    }
}
```