MajorityElement.java

```
package LAB_Problem.Problem4;

public class MajorityElement {
    public static int findMajorityElement(int[] nums) {
        int count = 0, candidate = 0;
        for (int num : nums) {
            if (count == 0) {
                candidate = num;
            }
            count += (num == candidate) ? 1 : -1;
        }
        return candidate;
    }

    public static void main(String[] args) {
        System.out.println(findMajorityElement(new int[]{1,2,2,2,3,3,3,3,3,3,
3}));
    }
}
```

MergeSortedArray.java

```
/**
 * ============================================================
 * Program Name: MergeSortedArray
 * Description :
 * Author      : Keshav Abhishek
 * Created On  : 11-02-2025
 * Organization: C.V. Raman Global University
 * ============================================================
 * Copyright (c) 2025, All rights reserved.
 * ============================================================
 */

package LAB_Problem.Problem4;

import java.util.Scanner;
public class MergeSortedArray {

    public static void merge(int[] nums1, int m, int[] nums2, int n) {
        int i = m - 1;
        int j = n - 1;
        int k = m + n - 1;

        while (i >= 0 && j >= 0) {
            if (nums1[i] > nums2[j]) {
                nums1[k--] = nums1[i--];
            } else {
                nums1[k--] = nums2[j--];
            }
        }

        while (j >= 0) {
            nums1[k--] = nums2[j--];
        }
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the size of the first array (m): ");
        int m = scanner.nextInt();
        int[] nums1 = new int[m + m];

        System.out.println("Enter the elements of the first array:");
        for (int i = 0; i < m; i++) {
```

```
            nums1[i] = scanner.nextInt();
        }

        System.out.print("Enter the size of the second array (n): ");
        int n = scanner.nextInt();
        int[] nums2 = new int[n];

        System.out.println("Enter the elements of the second array:");
        for (int i = 0; i < n; i++) {
            nums2[i] = scanner.nextInt();
        }

        merge(nums1, m, nums2, n);

        System.out.print("Merged array: ");
        for (int num : nums1) {
            System.out.print(num + " ");
        }
        scanner.close();
    }
}
```

MissingNumber1toN.java

```
/**
 * ============================================================
 * Program Name: MissingNumber1toN
 * Description :
 * Author      : Keshav Abhishek
 * Created On  : 11-02-2025
 * Organization: C.V. Raman Global University
 * ============================================================
 * Copyright (c) 2025, All rights reserved.
 * ============================================================
 */

package LAB_Problem.Problem4;

public class MissingNumber1toN {
    public void missing(int arr[]){
        int sum = 0;
        int actualSum = (arr.length+1) * (arr.length + 2) / 2;

        for(int i : arr){
            sum += i;
        }

        System.out.println("Missing: " + (actualSum - sum));
    }

    public static void main(String[] args){
        MissingNumber1toN M = new MissingNumber1toN();
        M.missing(new int[]{1,2,3,4,5,6,8});
    }
}
```

Pair.java

```
/**
 * ============================================================
 * Program Name: Pair
 * Description :
 * Author      : Keshav Abhishek
 * Created On  : 11-02-2025
 * Organization: C.V. Raman Global University
 * ============================================================
 * Copyright (c) 2025, All rights reserved.
 * ============================================================
 */

package LAB_Problem.Problem4;

import java.util.Arrays;
public class Pair {
    public static int[] findPairWithSum(int[] nums, int target) {
        for (int i = 0; i < nums.length; i++) {
            for (int j = i + 1; j < nums.length; j++) {
                if (nums[i] + nums[j] == target) {
                    return new int[]{nums[i], nums[j]};
                }
            }
        }
        return new int[]{};
    }

    public static void main(String[] args) {
        int[] nums1 = {1, 2, 2, 3, 4};

        int target = 7;
        int[] pairWithSum = findPairWithSum(nums1, target);
        if (pairWithSum.length > 0) {
            System.out.println("Pair with sum " + target + ": " + Arrays.
toString(pairWithSum));
        } else {
            System.out.println("No pair with sum " + target + " found.");
        }
    }
}
```

RemoveDuplicate.java

```
package LAB_Problem.Problem4;
import java.util.Arrays;
public class RemoveDuplicate {
    public static int[] removeDuplicates(int[] nums) {
        if (nums.length == 0) return new int[0];
        int i = 0;
        for (int j = 1; j < nums.length; j++) {
            if (nums[j] != nums[i]) {
                i++;
                nums[i] = nums[j];
            }
        }
        int[] result = new int[i + 1];
        for (int k = 0; k <= i; k++) {
            result[k] = nums[k];
        }
        return result;
    }

    public static void main(String[] args) {
        int[] nums = {1, 1, 2, 2, 3, 4, 4, 5};
        int[] newArray = removeDuplicates(nums);
        System.out.println(Arrays.toString(newArray));
    }
}
```

SecondLargestEle.java

```
/**
 * ============================================================
 * Program Name: SecondLargestEle
 * Description :
 * Author      : Keshav Abhishek
 * Created On  : 11-02-2025
 * Organization: C.V. Raman Global University
 * ============================================================
 * Copyright (c) 2025, All rights reserved.
 * ============================================================
 */
package LAB_Problem.Problem4;

public class SecondLargestEle {
    public static int secondLarge(int arr[]){

        int max = arr[0];
        int temp = arr[0];

        for (int i = 0; i < arr.length; i++) {
            if (arr[i] > max) {
                temp = max;
                max = arr[i];
            } else if (arr[i] > temp && arr[i] != max) {
                temp = arr[i];
            }
        }

        return temp;
    }

    public static void main(String[] args) {
        System.out.print("Second largest element: " + secondLarge(new int[]{1,3,
9,8,7}));
    }
}
```

Sorted.java

```
/**
 * ============================================================
 * Program Name: Sorted
 * Description :
 * Author      : Keshav Abhishek
 * Created On  : 11-02-2025
 * Organization: C.V. Raman Global University
 * ============================================================
 * Copyright (c) 2025, All rights reserved.
 * ============================================================
 */



package LAB_Problem.Problem4;

public class Sorted {
    public static int isSorted(int arr[]){
        int flag = 1;
        for (int i = 0; i < arr.length-1; i++) {
            if(arr[i]<arr[i+1]){}
            else{
                flag = 0;
                break;
            }
        }
        return flag;
    }

    public static void main(String[] args) {
        System.out.println("Sorted (true / false): " + (1==isSorted(new int[]{1,
2,3,4,5})));
    }
}
```

UnionIntersection.java

```
/**
 * ============================================================
 * Program Name: UnionIntersection
 * Description :
 * Author      : Keshav Abhishek
 * Created On  : 11-02-2025
 * Organization: C.V. Raman Global University
 * ============================================================
 * Copyright (c) 2025, All rights reserved.
 * ============================================================
 */

package LAB_Problem.Problem4;

import java.util.Arrays;

public class UnionIntersection {

    public static int[] union(int[] nums1, int[] nums2) {
        int[] temp = new int[nums1.length + nums2.length];
        int k = 0;

        for (int num : nums1) {
            if (!contains(temp, k, num)) {
                temp[k++] = num;
            }
        }

        for (int num : nums2) {
            if (!contains(temp, k, num)) {
                temp[k++] = num;
            }
        }

        int[] result = Arrays.copyOf(temp, k);
        return result;
    }

    public static int[] intersection(int[] nums1, int[] nums2) {
        int[] temp = new int[Math.min(nums1.length, nums2.length)];
        int k = 0;

        for (int num1 : nums1) {
            for (int num2 : nums2) {
```

```
            if (num1 == num2 && !contains(temp, k, num1)) {
                temp[k++] = num1;
                break;
            }
        }
    }
    int[] result = Arrays.copyOf(temp, k);
    return result;
}

private static boolean contains(int[] arr, int length, int key) {
    for (int i = 0; i < length; i++) {
        if (arr[i] == key) {
            return true;
        }
    }
    return false;
}

public static void main(String[] args) {
    int[] nums1 = {1, 2, 2, 3, 4};
    int[] nums2 = {3, 4, 4, 5, 6};

    int[] unionArray = union(nums1, nums2);
    System.out.println("Union: " + Arrays.toString(unionArray));

    int[] intersectionArray = intersection(nums1, nums2);
    System.out.println("Intersection: " + Arrays.toString(
intersectionArray));
}
}
```