**Name: Kunal Baghele**

**Java Assignment-3**

1. **Given two sorted arrays A and B of size p and q, write a Java program to merge elements of A with B by maintaining the sorted order i.e. fill A with first p smallest elements and fill B with remaining elements.**  
   Input :  
   int[] A = { 1, 5, 6, 7, 8, 10 }  
   int[] B = { 2, 4, 9 }  
   Output:  
   Sorted Arrays:  
   A: [1, 2, 4, 5, 6, 7]  
   B: [8, 9, 10]

package assignment3;

import java.util.Arrays;

import java.util.Scanner;

public class SortAndSplitArray {

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

System.out.print("Enter size of array A[] : ");

int p = s.nextInt();

System.out.print("Enter size of array B[] : ");

int q = s.nextInt();

int a[] = new int[p];

int b[] = new int[q];

int c[] = new int[p + q];

int i, j, k;

for (i = 0; i < p; i++) {

System.out.print("Enter " + (i + 1) + " element of A[] :");

a[i] = s.nextInt();

}

System.out.println("----------------------------------------");

for (j = 0; j < q; j++) {

System.out.print("Enter " + (j + 1) + " element of B[] :");

b[j] = s.nextInt();

}

System.out.println("\nInitial Arrays:");

System.out.println("A: " + Arrays.toString(a));

System.out.println("B: " + Arrays.toString(b));

i = 0;j = 0;

for (k = 0; k < p + q; k++) {

if (i < p) {

c[k] = a[i];

i++;

} else {

c[k] = b[j];

j++;

}

}

Arrays.sort(c);

for (k = 0; k < p; k++) {

a[k] = c[k];

}

for (k = 0; k < q; k++) {

b[k] = c[p + k];

}

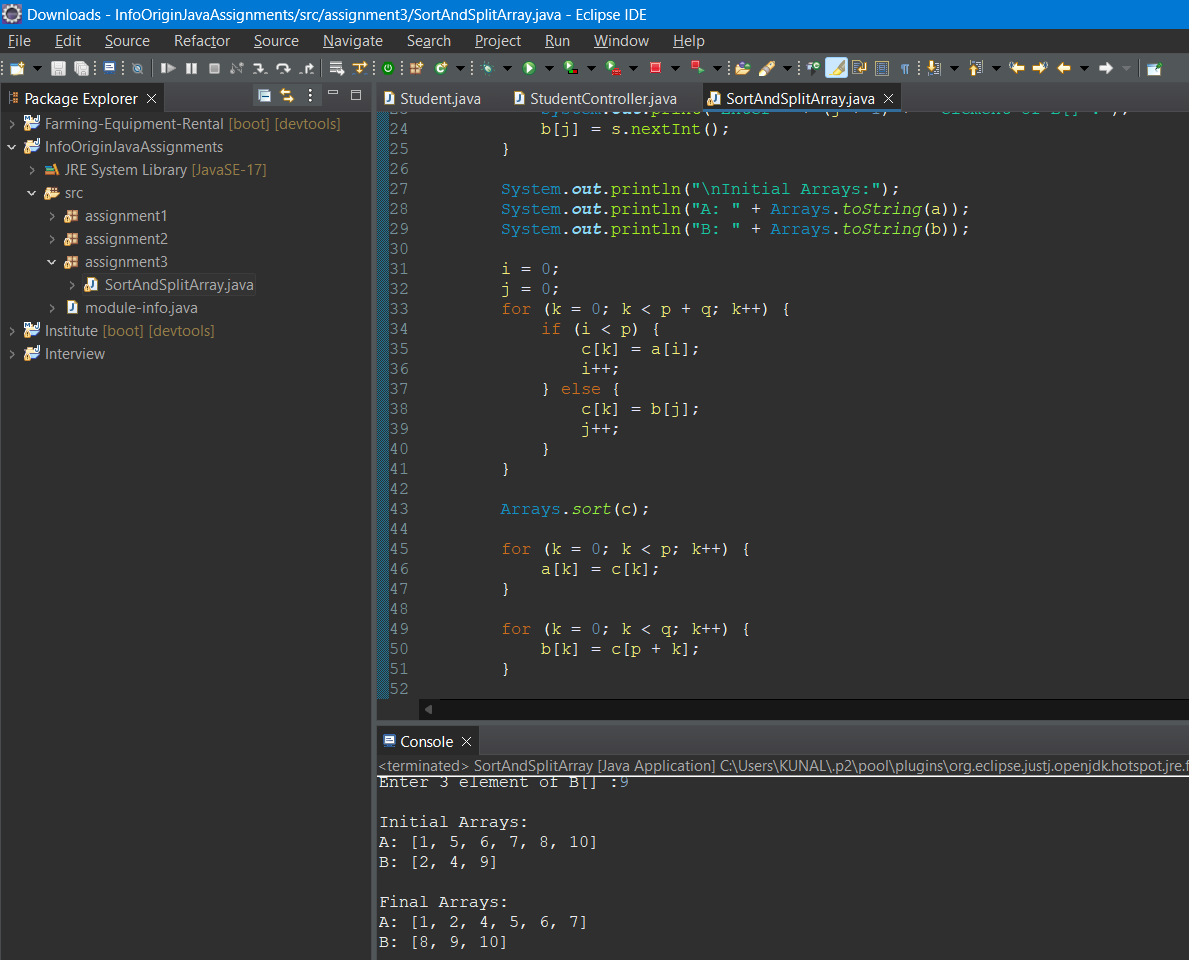
System.out.println("\nFinal Arrays:");

System.out.println("A: " + Arrays.toString(a));

System.out.println("B: " + Arrays.toString(b));

}

}



1. **Write a Java program to create an array of its anti-diagonals from a given square matrix.**

Example:  
Input :  
1 2  
3 4  
Output:  
[  
[1],  
[2, 3],  
[4]  
]

package assignment3;

import java.util.Scanner;

public class AntiDiagonals {

public static void main(String[] args) {

Scanner s = new Scanner(System.***in***);

System.***out***.println("Enter the size of the square matrix (n): ");

int n = s.nextInt();

int[][] matrix = new int[n][n];

// Input matrix elements

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

System.***out***.println("Enter elements of the matrix:");

matrix[i][j] = s.nextInt();

}

}

// Create and print the array of anti-diagonals

int[][] result = *createAntiDiagonals*(matrix);

System.***out***.println("Output:");

*printMatrix*(result);

}

private static int[][] createAntiDiagonals(int[][] matrix) {

int n = matrix.length;

int[][] result = new int[2 \* n - 1][];

for (int k = 0; k < 2 \* n - 1; k++) {

int i, j;

if (k < n) {

i = 0;

j = k;

} else {

i = k - n + 1;

j = n - 1;

}

int size = Math.*min*(k + 1, 2 \* n - k - 1);

result[k] = new int[size];

for (int m = 0; m < size; m++) {

result[k][m] = matrix[i][j];

i++;

j--;

}

}

return result;

}

private static void printMatrix(int[][] matrix) {

for (int[] row : matrix) {

for (int element : row) {

System.***out***.print(element + " ");

}

System.***out***.println();

}

}

}

