LAB 4

PRACTICE:

Example 4.1: Initialization of an Array

**package** PRACTICE;

**public** **class** EX4\_1 {

**public** **static** **void** main(String[] args) {

**int**[][] myarray = **new** **int**[2][2];

myarray[0][0] = 1;

myarray[0][1] = myarray[1][0] = 0;

myarray[1][1] = 1;

System.***out***.println("Array elements are");

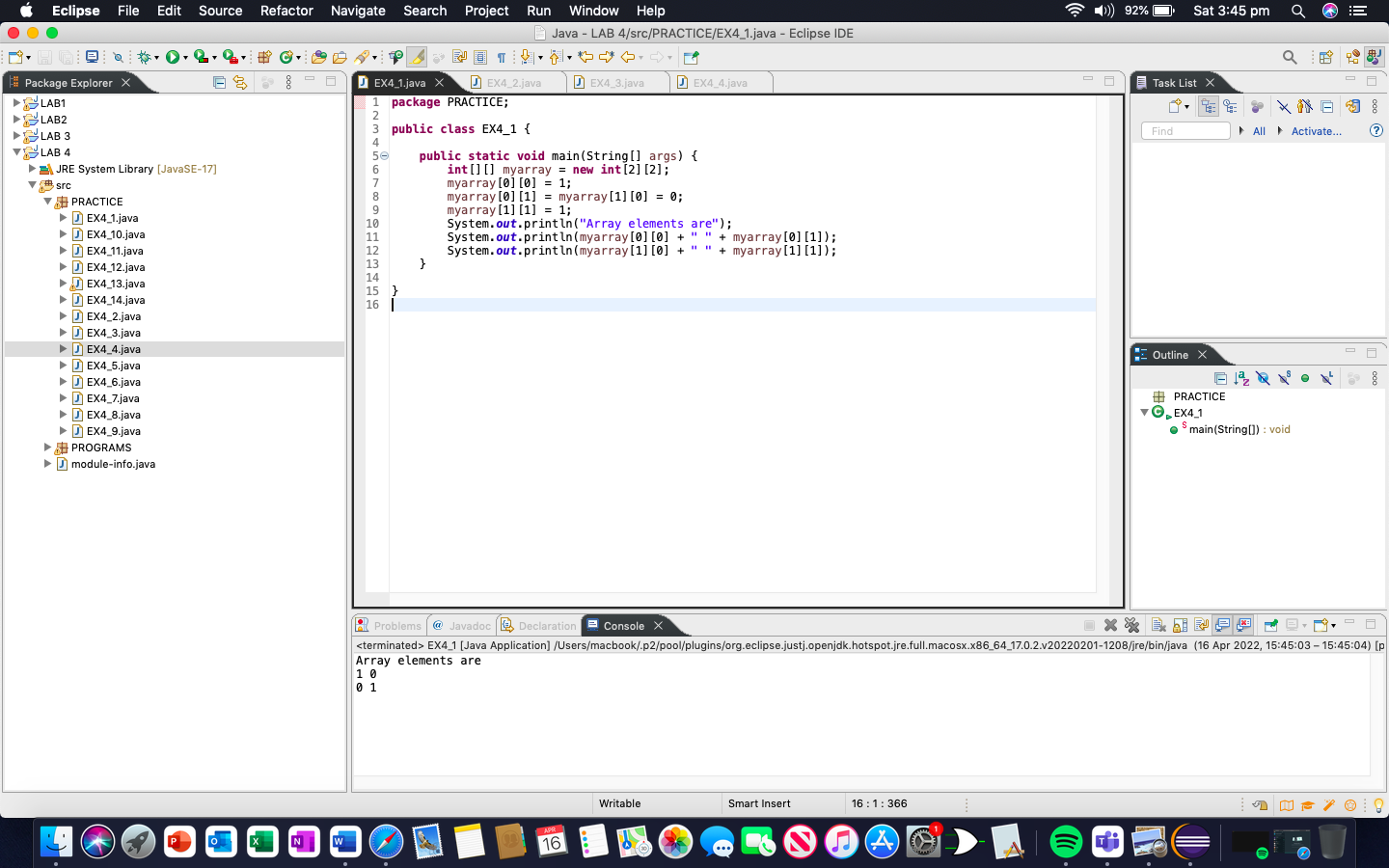
System.***out***.println(myarray[0][0] + " " + myarray[0][1]);

System.***out***.println(myarray[1][0] + " " + myarray[1][1]);

}

}

OUTPUT



Example 4.2: The following example shows the 2d array declaration with initialization.

**package** PRACTICE;

**public** **class** EX4\_2 {

**public** **static** **void** main(String[] args) {

//2-d array initialised with values

**int**[][] intArray = {

{1,2},

{3,4},

{5,6}

};

//print the array

System.***out***.println("Intialized two dimensional array");

**for** (**int** i=0;i<3;i++) {

**for**(**int** j=0;j<2;j++) {

System.***out***.print(intArray[i][j] + " ");

}

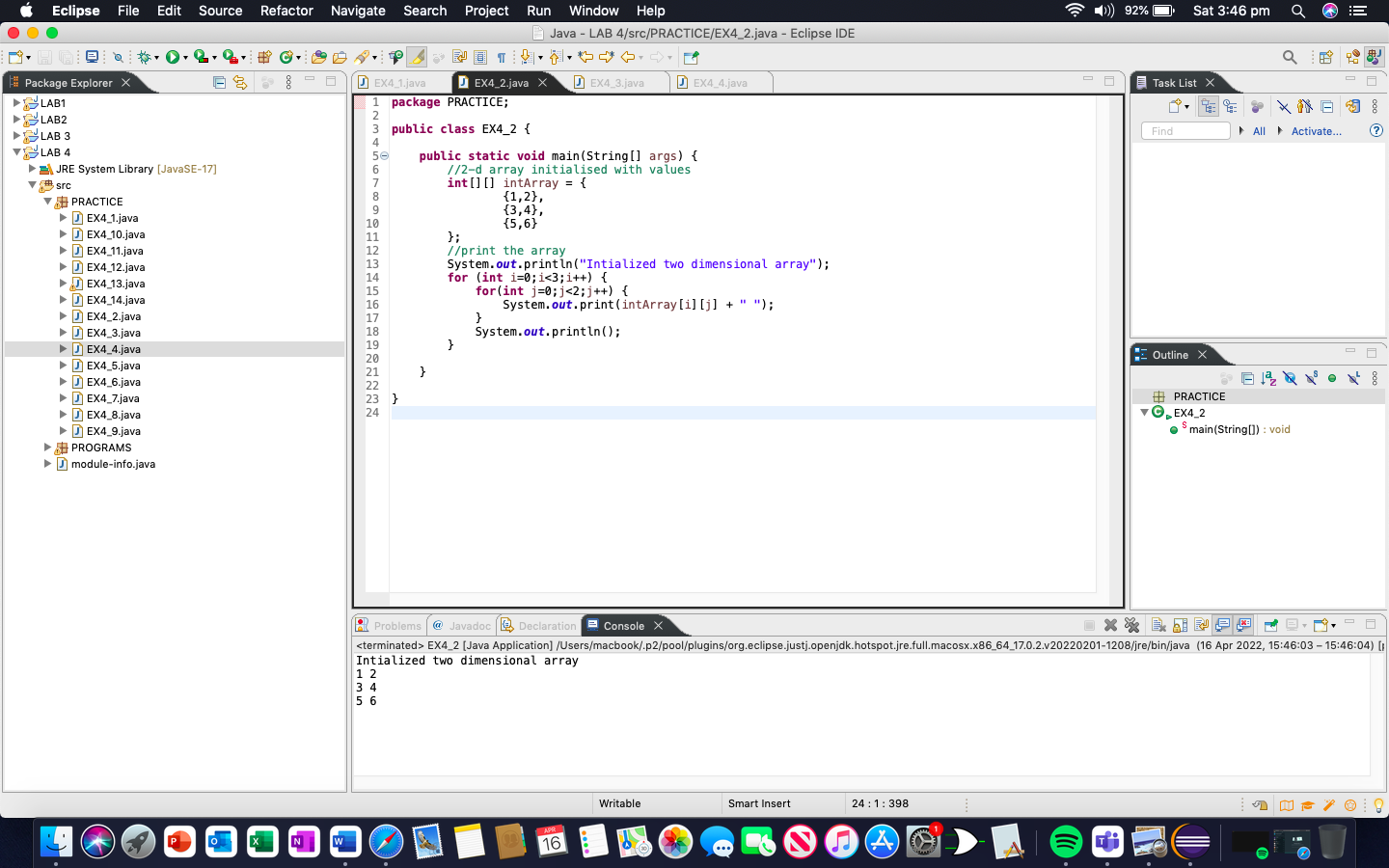
System.***out***.println();

}

}

}

OUTPUT



Example 4.3: Using loop to initialize the array.

**package** PRACTICE;

**public** **class** EX4\_3 {

**public** **static** **void** main(String[] args) {

//declare an array of int

**int**[][] intArray = **new** **int**[3][3];

**for**(**int** i=0;i<3;i++) {

**for**(**int** j=0; j<3; j++) {

intArray[i][j] = i+1;

System.***out***.print(intArray[i][j] + " ");

}

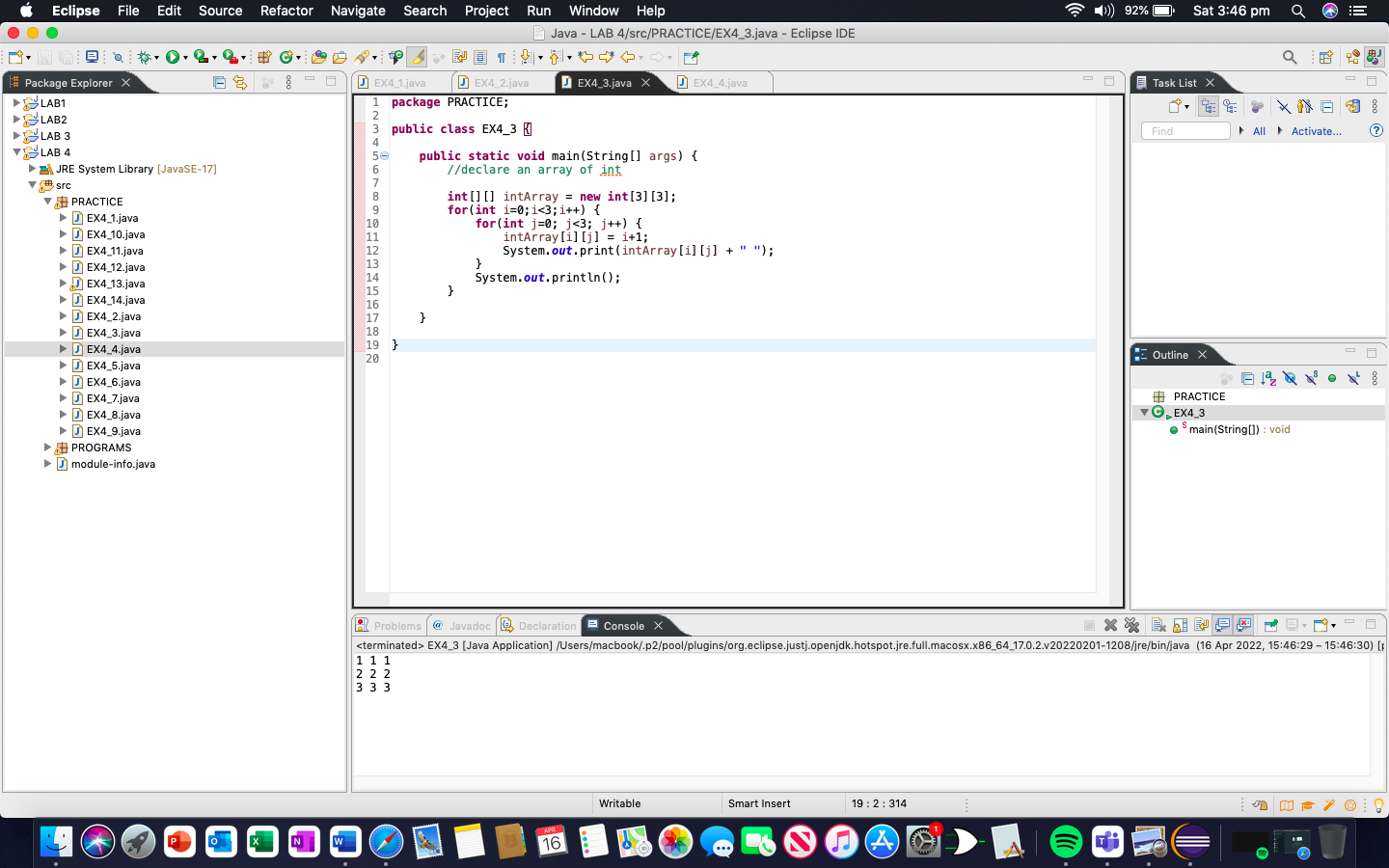
System.***out***.println();

}

}

}

OUTPUT



Example 4.4: The following program shows how an individual element is accessed and printed.

**package** PRACTICE;

**public** **class** EX4\_4 {

**public** **static** **void** main(String[] args) {

//two dimensional array definition

**int**[][] intArray = {

{1,2},

{4,8}

};

//Access individual element of array

**int** val = intArray[0][1];

//print the element

System.***out***.println("Accessed array value " + val);

System.***out***.println("contents of array");

//print the individual elements of array

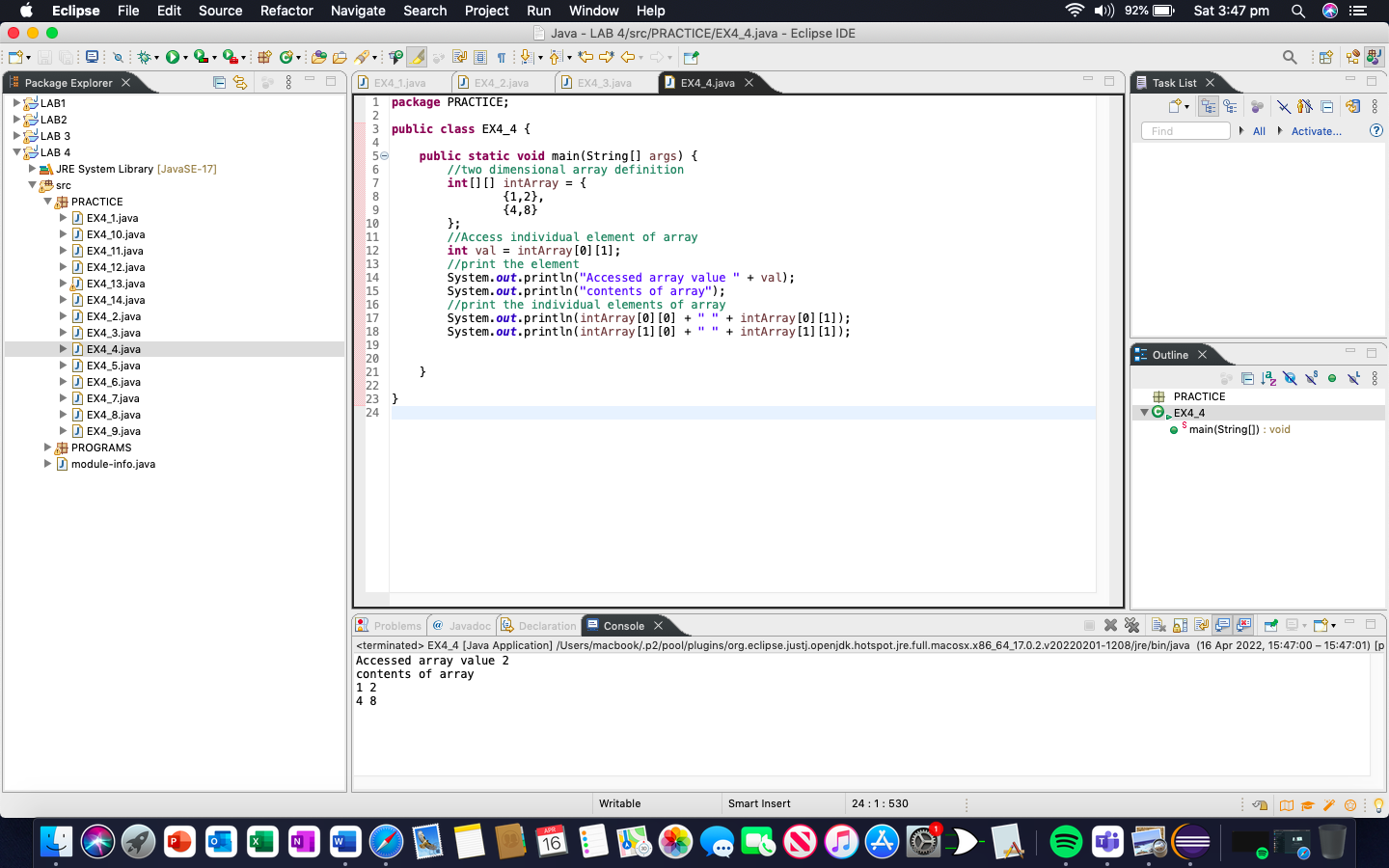
System.***out***.println(intArray[0][0] + " " + intArray[0][1]);

System.***out***.println(intArray[1][0] + " " + intArray[1][1]);

}

}

OUTPUT



Example 4.5: The following program shows the printing of a 2d array using a ‘for’ loop.

**package** PRACTICE;

**public** **class** EX4\_5 {

**public** **static** **void** main(String[] args) {

//two dimension array definition

**int**[][] intArray = **new** **int**[3][3];

//printing 2-d array

System.***out***.println("The two-dimensional array:");

**for**(**int** i=0;i<3;i++) {

**for**(**int** j=0; j<3; j++) {

intArray[i][j] = i\*j;

System.***out***.print(intArray[i][j] + " ");

}

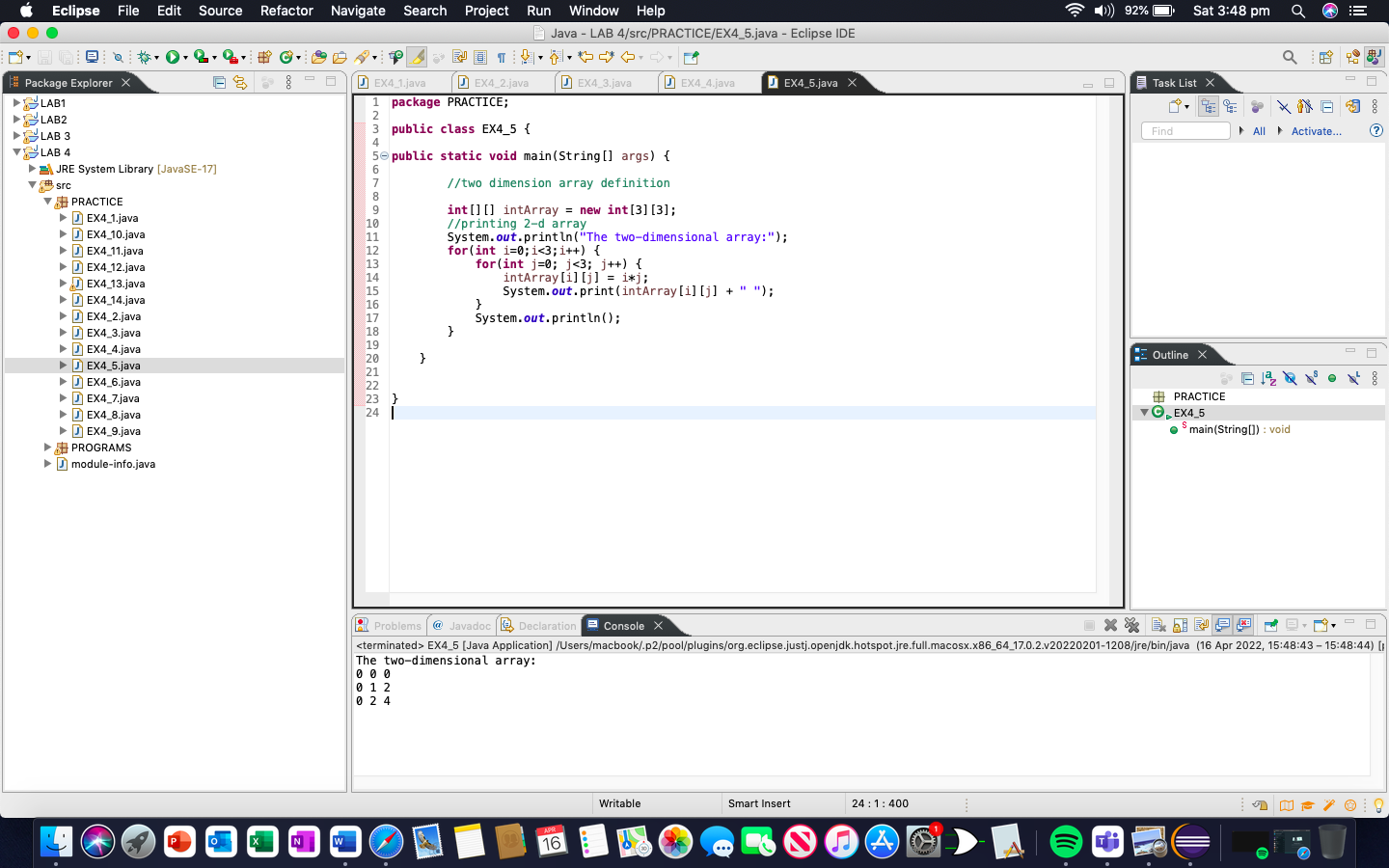
System.***out***.println();

}

}

}

OUTPUT



Example 4.6: The following program gives the length of the array (number of rows) as well as the size of each row.

**package** PRACTICE;

**public** **class** EX4\_6 {

**public** **static** **void** main(String[] args) {

//initialize 2-d array

**int**[][] myarray = { { 1,2,3}, {4,5}};

System.***out***.println("length of array:" + myarray.length); //number of rows

**for**(**int** i=0;i<myarray.length;i++) {

//length of each row

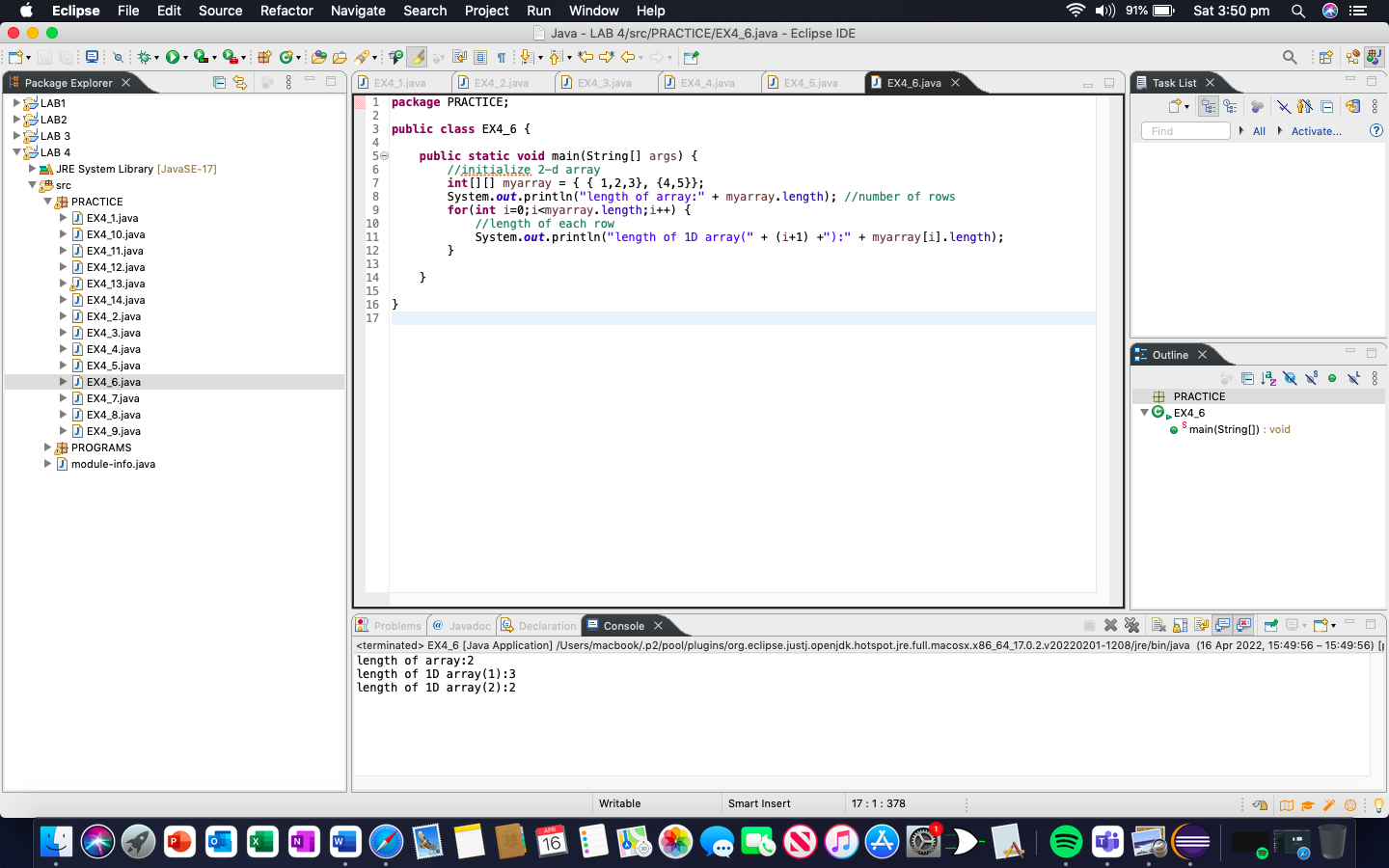
System.***out***.println("length of 1D array(" + (i+1) +"):" + myarray[i].length);

}

}

}

OUTPUT



Example 4.7: The following Java program shows the usage of length property to print the 2d array.

**package** PRACTICE;

**public** **class** EX4\_7 {

**public** **static** **void** main(String[] args) {

//two dimensional array definition

**int**[][] myarray = **new** **int**[3][3];

//printing the 2-d array

System.***out***.println("The two-dimensional array:");

**for** (**int** i=0; i<myarray.length; i++) {

**for**(**int** j=0; j<myarray[i].length;j++) {

myarray[i][j] = j+i;

System.***out***.print(myarray[i][j] + " ");

}

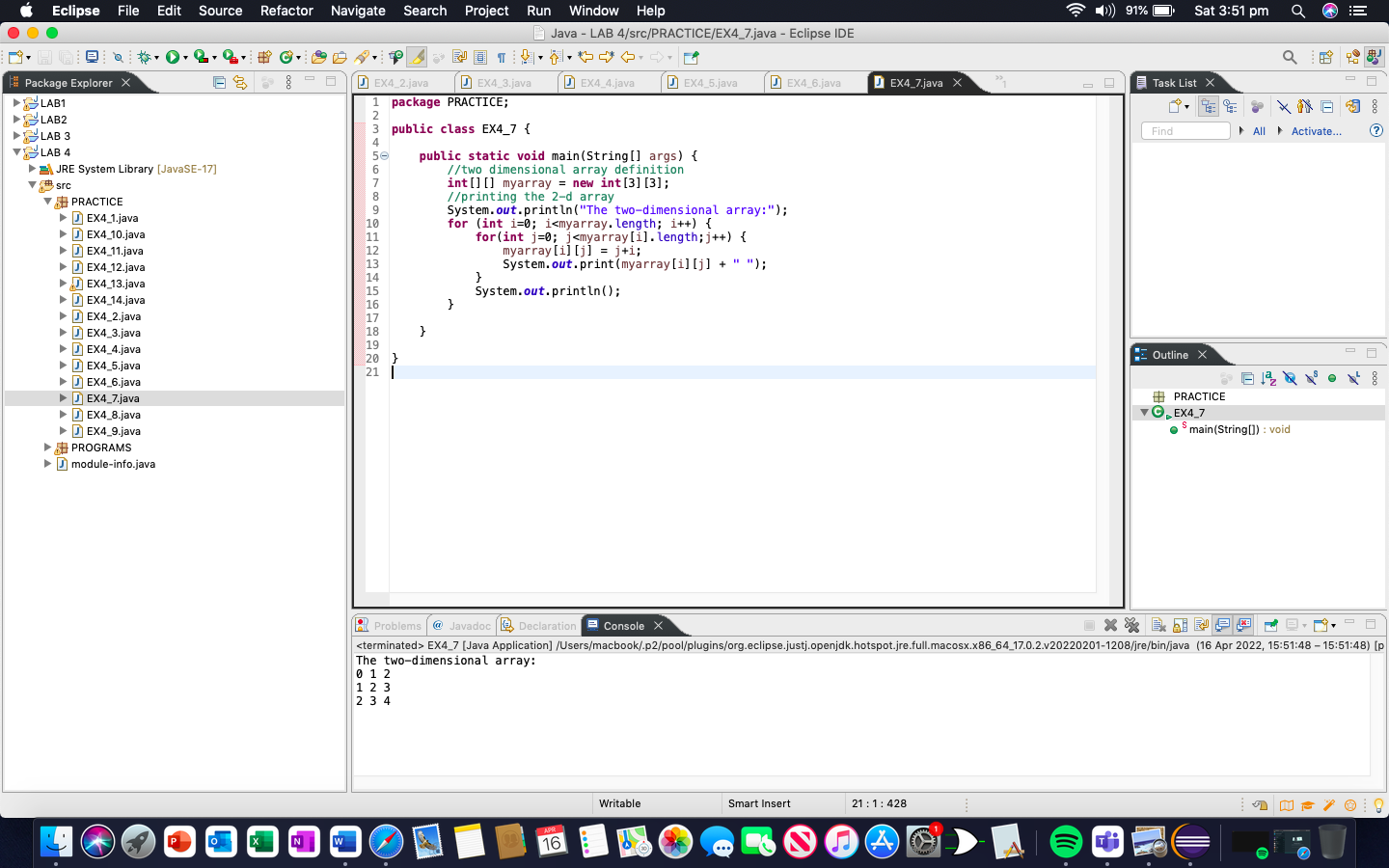
System.***out***.println();

}

}

}

OUTPUT



Example 4.8: The example below shows the initialization of the 3d array while declaration.

**package** PRACTICE;

**public** **class** EX4\_8 {

**public** **static** **void** main(String[] args) {

//initialize 3-d array

**int**[][][] intArray = { { {1,2,3},{4,5,6},{7,8,9} } };

System.***out***.println("3-d array is given below :");

//printl the elements of array

**for**(**int** i=0;i<1;i++) {

**for**(**int** j=0;j<3;j++) {

**for**(**int** z=0; z<3;z++) {

System.***out***.println("intArray [" + i + "][" + j + "][" + z + "] = " + intArray[i][j][z]);

}

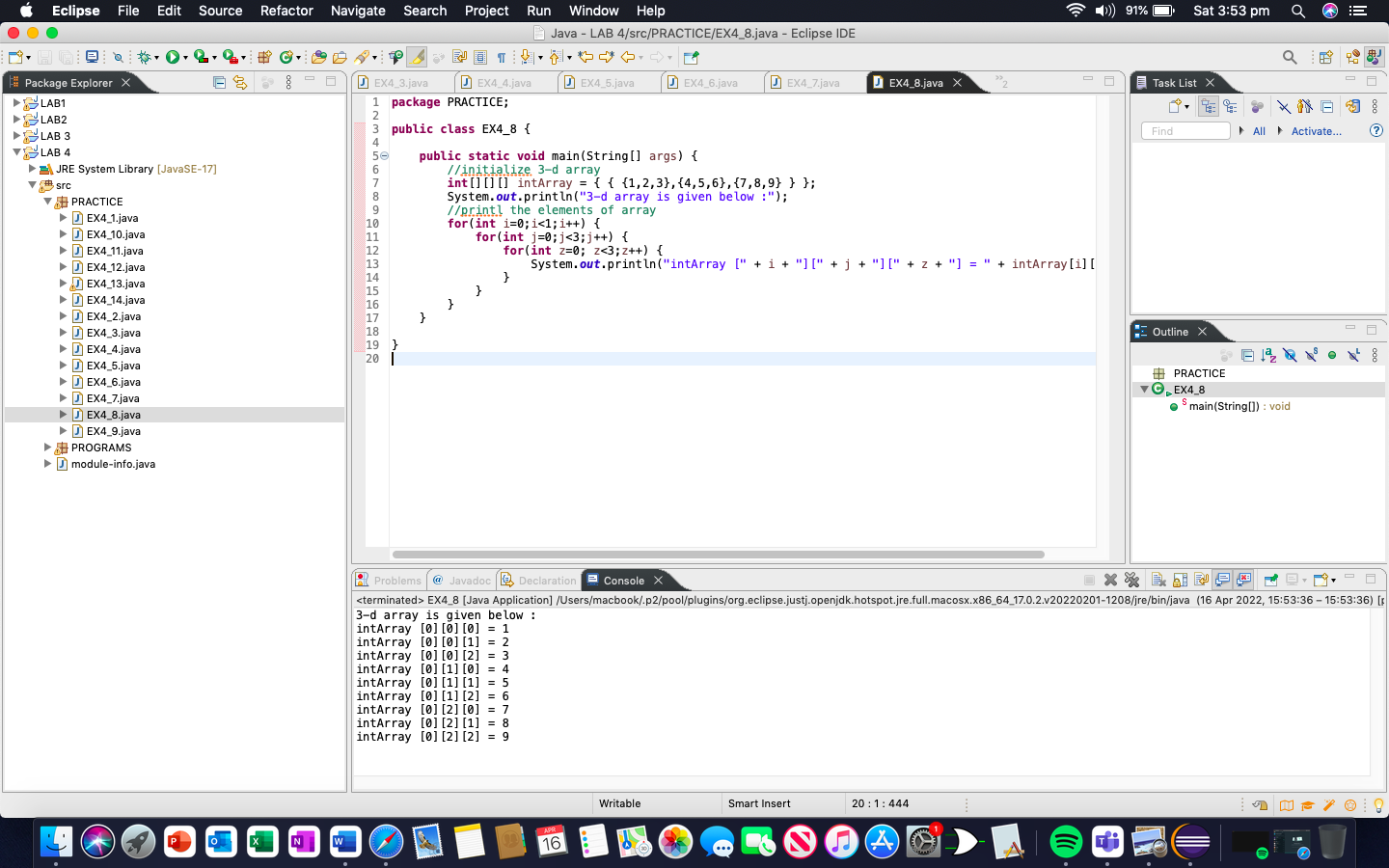
}

}

}

}

OUTPUT



Example 4.9: The program below uses for loops to access the array elements and print them to the console.

**package** PRACTICE;

**public** **class** EX4\_9 {

**public** **static** **void** main(String[] args) {

//initailized 3-d array

**int**[][][] myarray = { { {1,2,3}, {4,5,6} } , { { 1,4,9},{16,25,36} } , { { 1,8,27}, {64,125,216} } };

System.***out***.println("3x2x3 array is given below");

//print the 3-d array

**for**(**int** i=0; i<3; i++) {

**for**(**int** j=0; j<2; j++) {

**for**(**int** k=0; k<3; k++) {

System.***out***.print(myarray[i][j][k] + "\t");

}

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

}

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

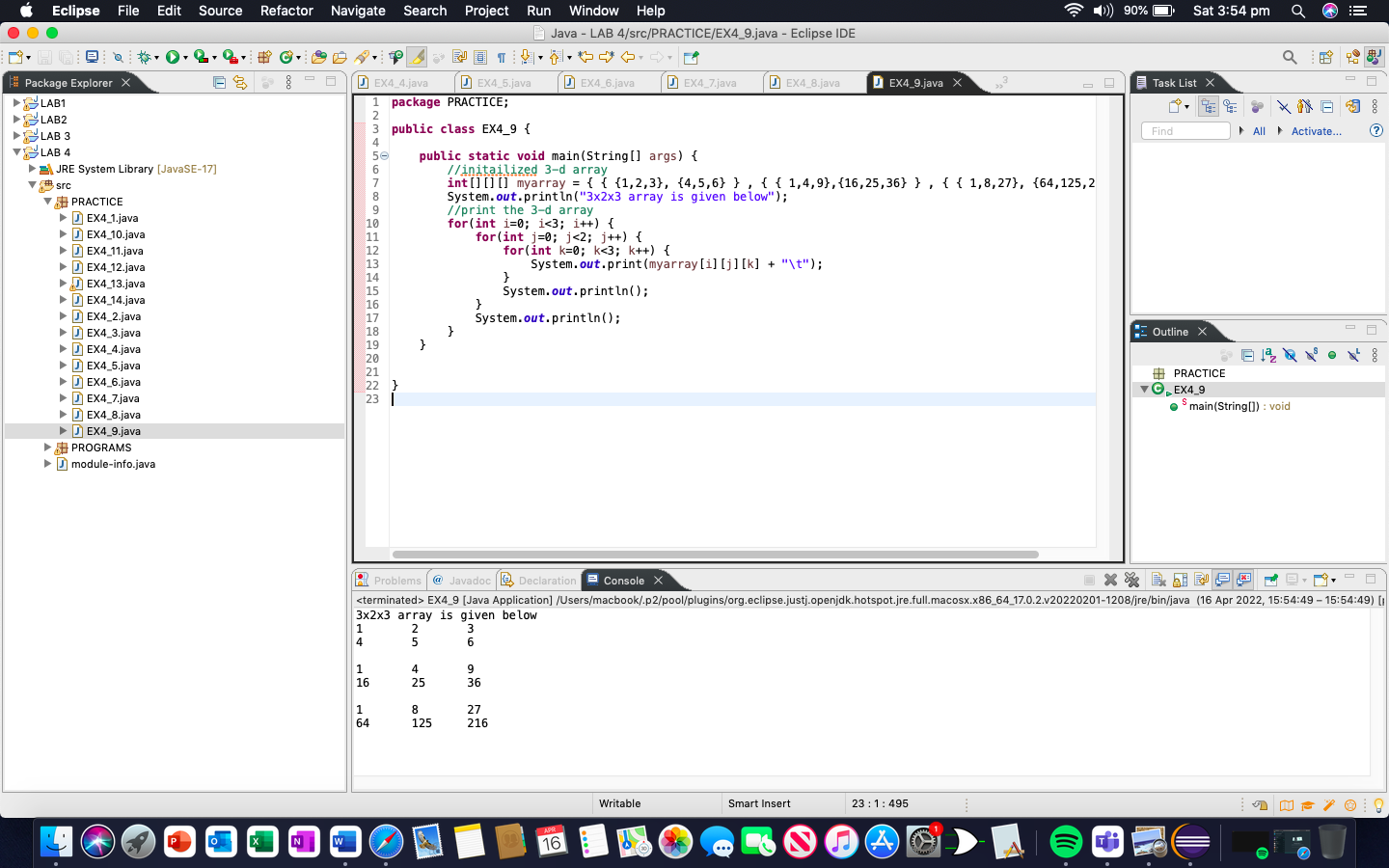
}

}

}

}

OUTPUT



Example 4.10: This program also uses enhanced for loop to traverse through the array and display its elements.

**package** PRACTICE;

**public** **class** EX4\_10 {

**public** **static** **void** main(String[] args) {

//initialize 3-d array

**int**[][][] intArray = {

{{10,20,30},{20,40,60}},

{ {10,30,50,70},{50},{80,90}}

};

System.***out***.println("Multidimensional Array (3-d) is as follow:");

//use for..each loop to iterate through elements of 3d array

**for**(**int**[][] array\_2D: intArray) {

**for**(**int**[] array\_1D: array\_2D) {

**for**(**int** elem: array\_1D) {

System.***out***.print(elem + "\t");

}

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

}

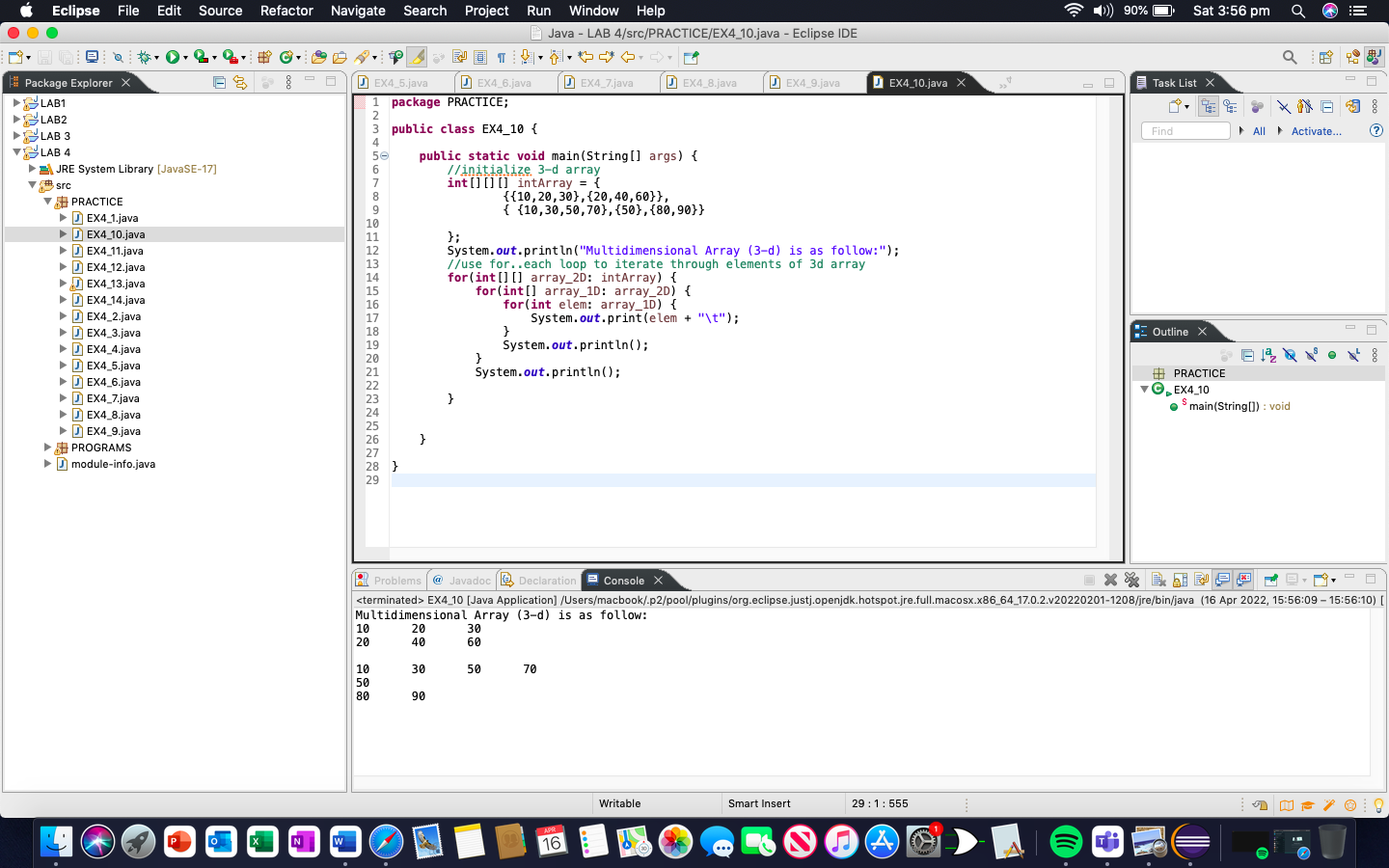
System.***out***.println();

}

}

}

OUTPUT



Example 4.11: The following program will print the elements in the array with their index numbers.

**package** PRACTICE;

**public** **class** EX4\_11 {

**public** **static** **void** main(String[] args) {

**int**[] Array = {27, 37, 18, 20, 54, 67, 35, 23, 19, 38};

System.***out***.printf("%s%Bs%n " , "Index", "Value");

**for**(**int** counter=0; counter<Array.length; counter++) {

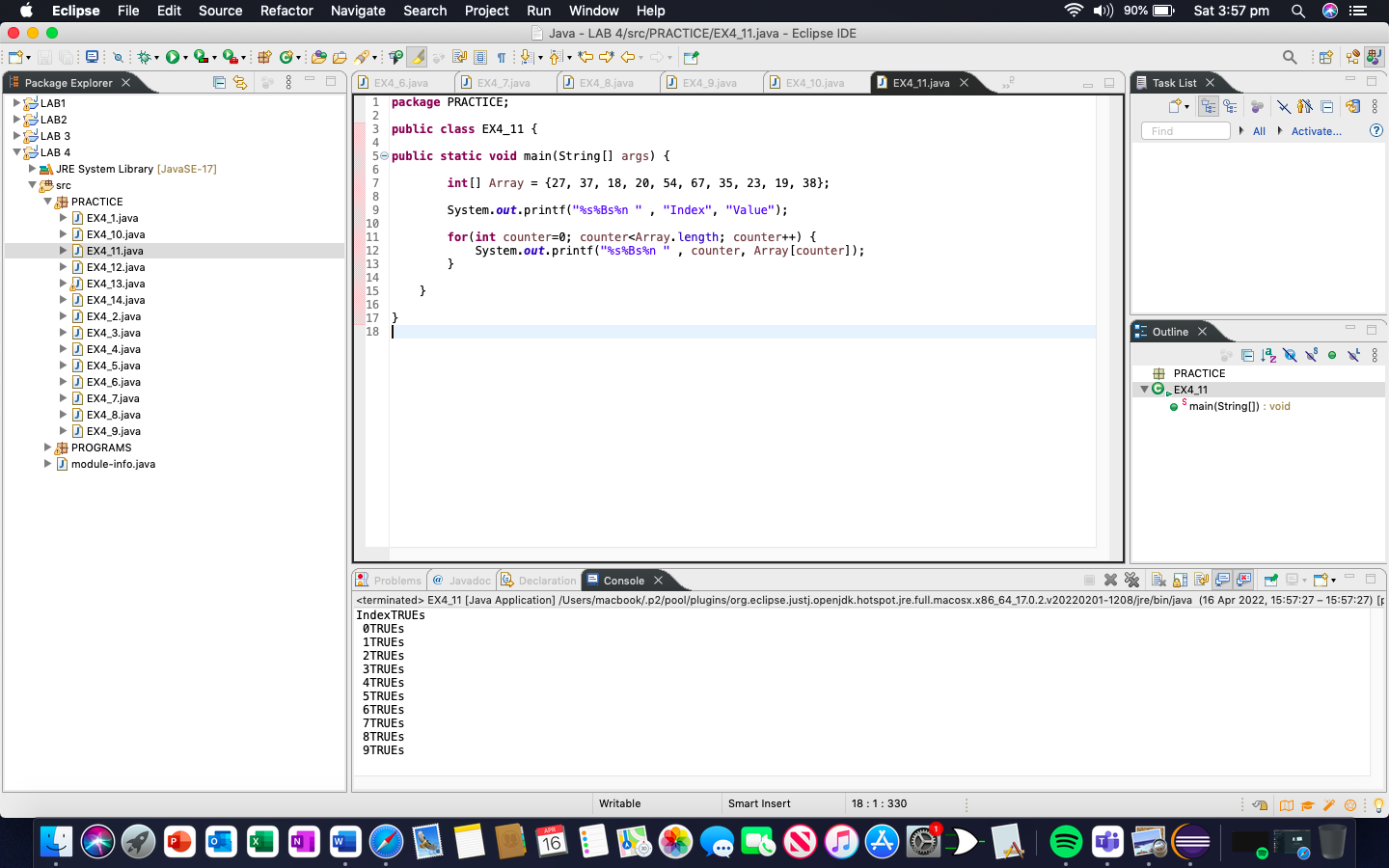
System.***out***.printf("%s%Bs%n " , counter, Array[counter]);

}

}

}

OUTPUT



Example 4.12: The following program will create the bar chart

**package** PRACTICE;

**public** **class** EX4\_12 {

**public** **static** **void** main(String[] args) {

**int**[] array = {0,0,0,0,0,0,1,2,4,2,1};

System.***out***.println("Grade Distribution");

**for**(**int** counter=0; counter<array.length; counter++) {

**if**(counter==10) {

System.***out***.printf("%5d: ", 100);

}**else** {

System.***out***.printf("%02d-%02d: ", counter\*10, counter\*10+9);

}

**for**(**int** stars=0; stars<array.length; stars++) {

System.***out***.print("\*");

}

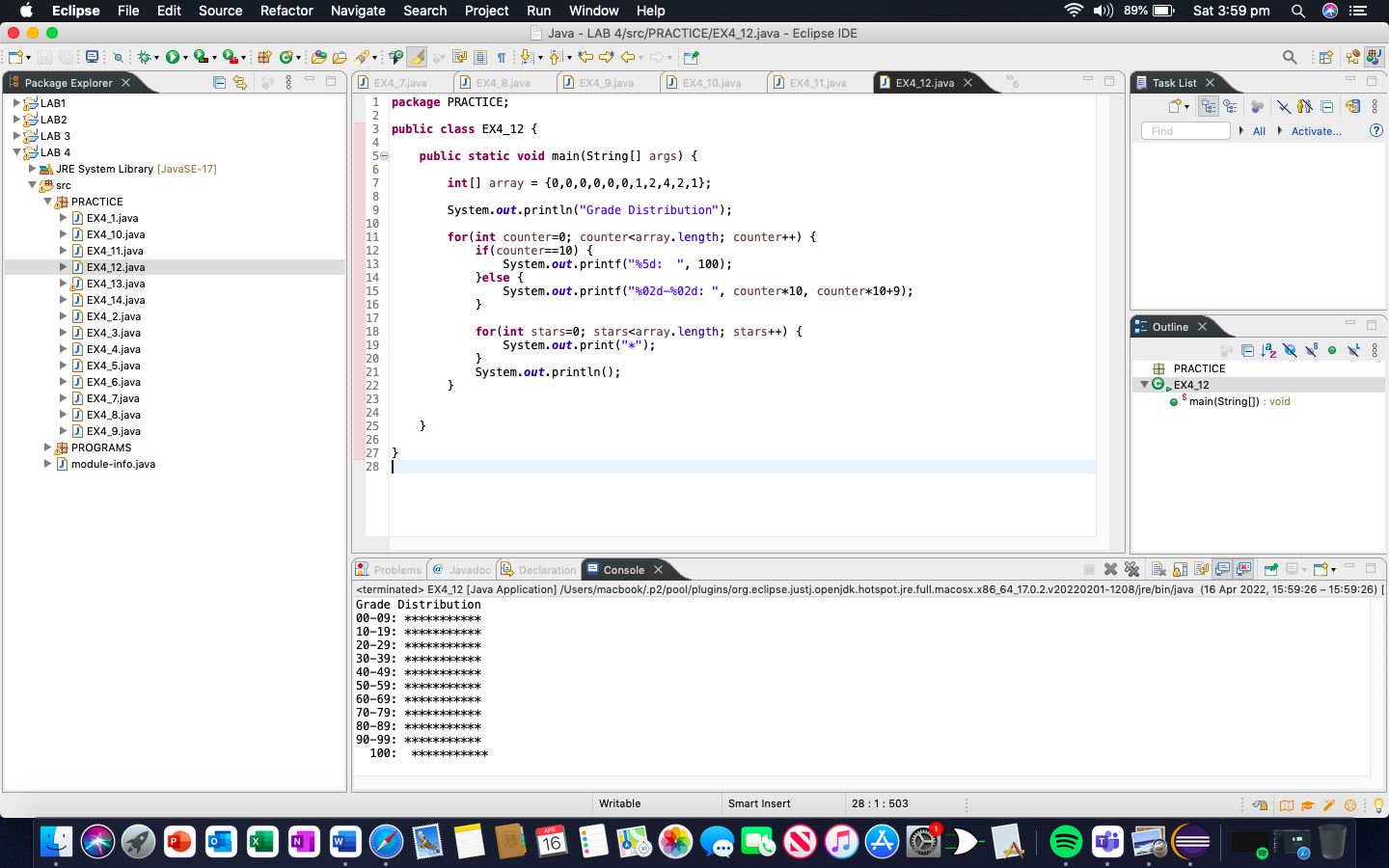
System.***out***.println();

}

}

}

OUTPUT



Example 4.13: Write a program which can passing a reference

**package** PRACTICE;

**import** java.util.Arrays;

**public** **class** EX4\_13 {

**public** **static** **void** main(String[] args) {

**int**[] array = {1,2,3,4,5};

System.***out***.printf("Effect of passing reference to entrie array:%n" +

"The values of the orignal array are:%n");

**for**(**int** value:array) {

System.***out***.printf(" %d ", value);

}

*modifyArray*(array);

System.***out***.printf("%n%nThe values of the modified array are:%n");

**for**(**int** value:array) {

System.***out***.printf(" %d ", value);

}

System.***out***.printf("%n%nEffects of passing array element value:%n" +

"array[3] before modifyElement:%d%n", array[3]);

*modifyElement*(array[3]);

System.***out***.printf("array[3] after modifyElement: %d%n", array[3]);

}

**private** **static** **void** modifyElement(**int** i) {

}

**public** **static** **void** modifyArray(**int**[] array2) {

**for** (**int** counter=0; counter<array2.length;counter++) {

array2[counter] \*=2;

}

}

}

OUTPUT



Example 4.14: Write a program that can have Array class methods and System.arraycopy

**package** PRACTICE;

**import** java.util.Arrays;

**public** **class** EX4\_14 {

**public** **static** **void** main(String[] args) {

//sort double array into asscending order

**double**[] doubleArray = {8.4, 9.3, 0.2, 7.9, 3.4};

Arrays.*sort*(doubleArray);

System.***out***.printf("%ndoubleArray :");

**for**(**double** value: doubleArray) {

System.***out***.printf("%.1f", value);

}

//filled 10-element arrays with 7s

**int**[] filledIntArray = **new** **int**[10];

Arrays.*fill*(filledIntArray, 7);

*displayArray*(filledIntArray, "filledIntArray");

//copy array intArray into array intArray2

**int**[] intArray = {1,2,3,4,5,6};

**int**[] intArraycopy = **new** **int**[intArray.length];

System.*arraycopy*(intArray, 0, intArraycopy, 0, intArray.length);

*displayArray*(intArraycopy, "intArraycopy");

//compare intArray and intArraycopy for equality

**boolean** b = Arrays.*equals*(intArray, intArraycopy);

System.***out***.printf("%n%nintArray &s intArraycopy%n",

(b ? "==":"!="));

//compare intArray and filledIntArray for equality

b = Arrays.*equals*(intArray, filledIntArray);

System.***out***.printf("intArray &s filledIntArray%n",

(b ? "==":"!="));

//search intArray for value 5

**int** location = Arrays.*binarySearch*(intArray, 5);

**if**(location>=0) {

System.***out***.printf("Found 5 at element %d int intArray%n", location);

}**else** {

System.***out***.println("5 not found in intArray");

}

//search intArray for value 876

location = Arrays.*binarySearch*(intArray, 876);

**if**(location>=0) {

System.***out***.printf("Found 876 at element %d int intArray%n", location);

}**else** {

System.***out***.println("876 not found in intArray");

}

}

**public** **static** **void** displayArray(**int**[] array, String description) {

System.***out***.printf("%n%s: ",description);

**for**(**int** value: array) {

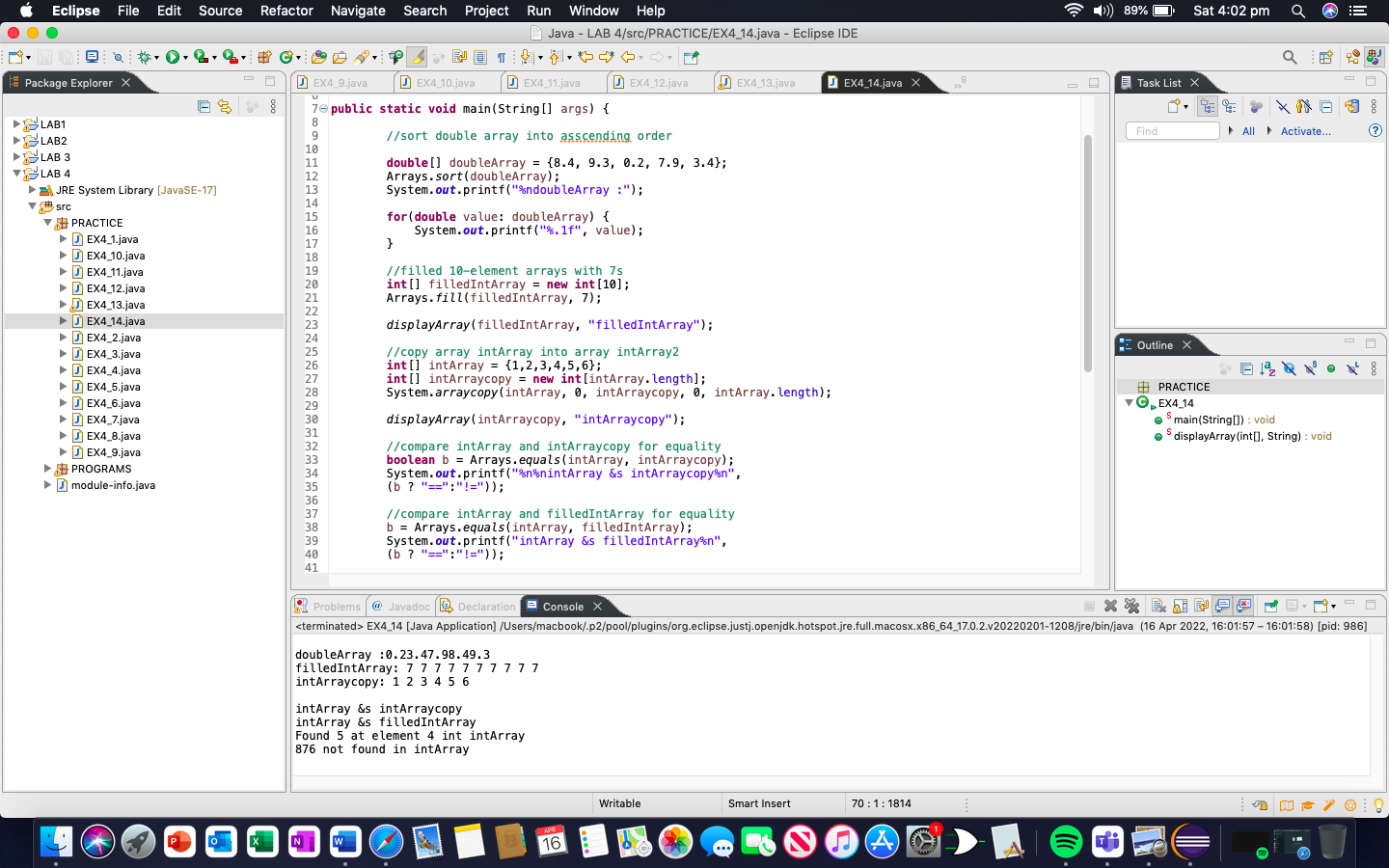
System.***out***.printf("%d ", value);

}

}

}

OUTPUT



PROGRAM:

PROGRAM1:

**package** PROGRAMS;

**public** **class** PROGRAM1 {

**public** **static** **void** main(String[] args) {

**int**[][] matrix1 = {

{1,4,5},

{3,3,4},

{3,6,8}

};

**int**[][] matrix2 = {

{2,6,5},

{4,6,2},

{8,2,4}

};

**int**[][] matrix = **new** **int**[3][3];

**for**(**int** i=0;i<3;i++) {

**for**(**int** j=0;j<3;j++) {

matrix[i][j] = 0;

**for**(**int** k=0;k<3;k++) {

matrix[i][j] += matrix1[i][k] \* matrix2[k][j];

}

System.***out***.print(matrix[i][j] + " ");

}

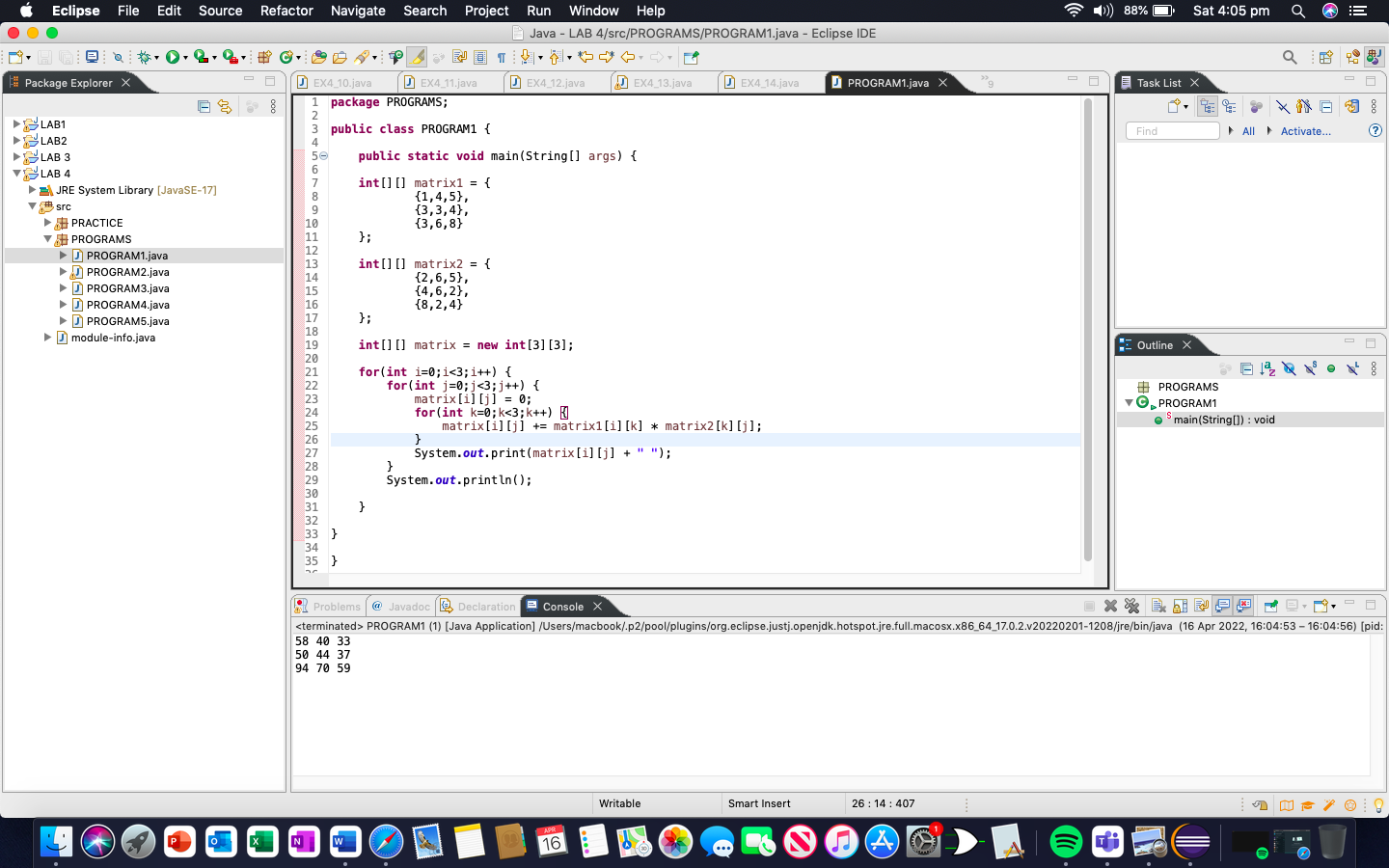
System.***out***.println();

}

}

}

OUTPUT



PROGRAM2:

package PROGRAMS;

import java.util.ArrayList;

import java.util.List;

public class PROGRAM2 {

public static void main(String[] args) {

int[][]matrix = {{1,2,3},{2,4,6},{6,2,9}};

System.out.println("Matrix 1");

List<Integer> flatArray = new ArrayList();

for (int[] row : matrix) {

for (int col : row) {

flatArray.add(col);

}

}

for(int i=0;i<matrix.length;i++) {

for(int j=0;j<matrix[i].length;j++) {

System.out.print(matrix[i][j] + " ");

}

System.out.println();

}

int max = matrix[0][0];

int min = matrix[0][0];

int duplicate\_count = 0;

for(int i=0;i<matrix.length;i++) {

boolean dupFound = false;

for(int j=0;j<matrix[i].length;j++) {

if(max<matrix[i][j]) {

max = matrix[i][j];

}else if(min>matrix[i][j]) {

min = matrix[i][j];

}else if(flatArray.get(i) == flatArray.get(j)) {

dupFound = true;

}

}

if(dupFound) {

duplicate\_count+=1;

}

}

System.out.println("largest number " + max + " smallest number " + min + " equal number " + duplicate\_count + "\n");

int[][] matrix2 = {{4,6,2},{2,6,5},{1,8,7}};

System.out.println("matrix2\n");

List<Integer> flatArray2 = new ArrayList();

for (int[] row : matrix2) {

for (int col : row) {

flatArray2.add(col);

}

}

for(int i=0; i<matrix2.length; i++) {

for(int j=0; j<matrix2[i].length; j++) {

System.out.print(matrix2[i][j] + " ");

}

System.out.println();

}

int max2 = matrix2[0][0];

int min2 = matrix2[0][0];

duplicate\_count = 0;

for(int i=0; i<matrix2.length; i++) {

boolean dupFound = false;

for(int j=0; j<matrix2[i].length; j++) {

if(max2<matrix2[i][j]) {

max2 = matrix2[i][j];

}else if(min2>matrix2[i][j]) {

min2 = matrix2[i][j];

}else if(flatArray2.get(i) == flatArray2.get(j)) {

dupFound = true;

}

}

if(dupFound) {

duplicate\_count+=1;

}

}

System.out.println("\nlargest number " + max2 + " smallest number " + min2 + " equal number " + duplicate\_count + "\n");

int[][] matrix3 = {{12,54,67},{78,20,34},{12,79,90}};

System.out.println("matrix3\n");

List<Integer> flatArray3 = new ArrayList();

for (int[] row : matrix3) {

for (int col : row) {

flatArray3.add(col);

}

}

for(int i=0; i<matrix3.length; i++) {

for(int j=0; j<matrix3[i].length; j++) {

System.out.print(matrix3[i][j] + " ");

}

System.out.println();

}

int max3 = matrix3[0][0];

int min3 = matrix3[0][0];

duplicate\_count= 0;

for(int i=0; i<matrix3.length; i++) {

boolean dupFound = false;

for(int j=0; j<matrix3[i].length; j++) {

if(max3<matrix3[i][j]) {

max3 = matrix3[i][j];

}else if(min3>matrix3[i][j]) {

min3 = matrix3[i][j];

}else if(flatArray3.get(i) == flatArray3.get(j)) {

dupFound = true;

}

}

if(dupFound) {

duplicate\_count+=1;

}

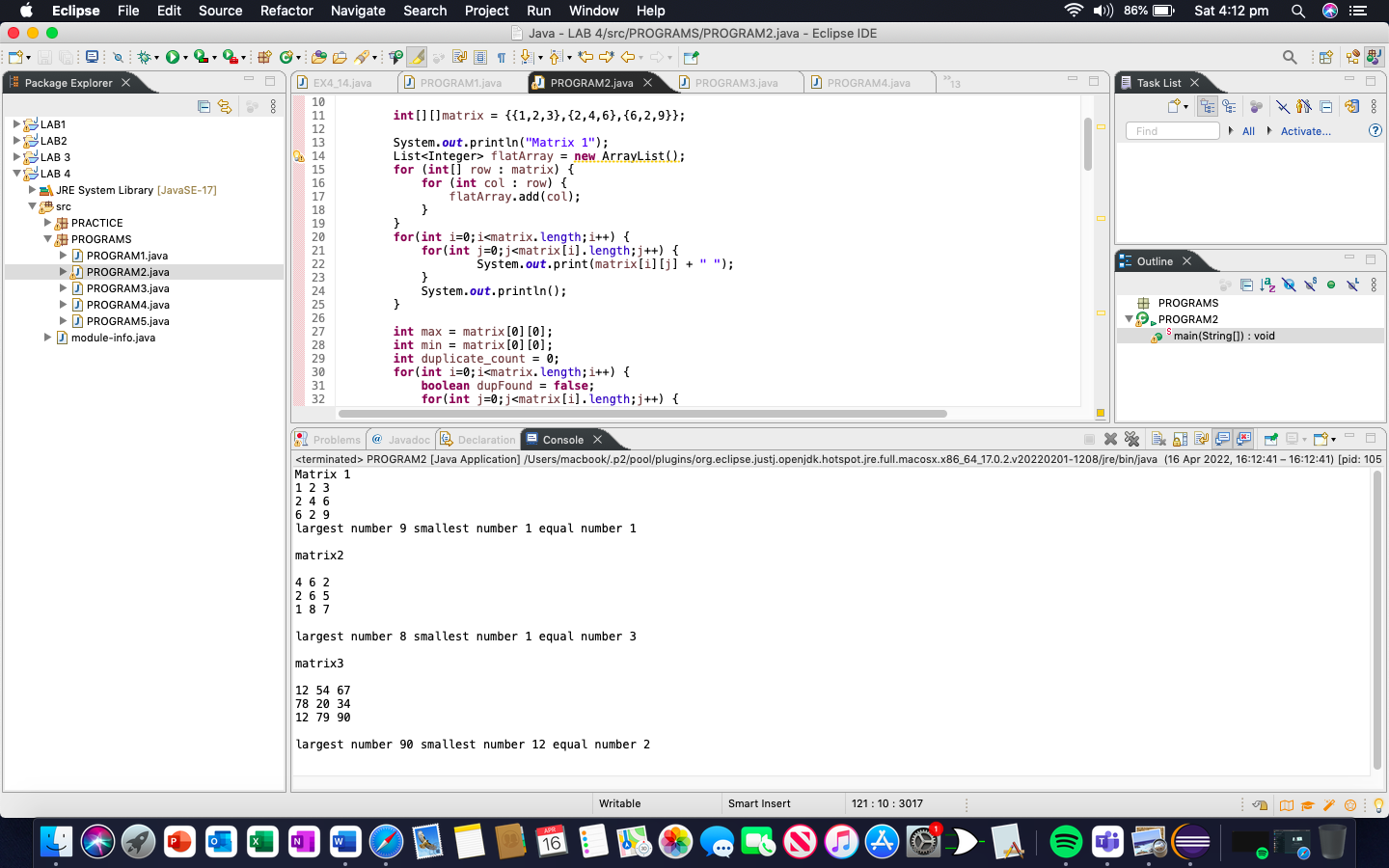
}

System.out.println("\nlargest number " + max3 + " smallest number " + min3 + " equal number " + duplicate\_count + "\n");

}

}

OUTPUT



PROGRAM3:

**package** PROGRAMS;

**public** **class** PROGRAM3 {

**public** **void** add(String n, **int** t, **int** s) {

String[][] name = {{"xyz", "rst", "sh", "sg", "tre"}, {"asd", "hd", "fs","df", "ed"}};

**int** len = name.length;

String[][] name2 = **new** String[len][len];

System.*arraycopy*(name, 0, name2, 0, name.length);

name2[t][s] = n;

**for**(**int** i=0; i<name2.length;i++) {

**for**(**int** j=0;j<name2[i].length;j++) {

System.***out***.print(name2[i][j] + " ");

}

System.***out***.println("\n");

}

}

**public** **void** insert(**int** n, **int** t, **int** s) {

**int**[][] array = {{1,2,3,6,78},{43,56,45,76,10}};

**int** Index\_position = n;

**int** Index\_position2 = s;

**int** new\_value = t;

**for**(**int** i=array.length-1; i > Index\_position; i--){

**for**(**int** j=array[i].length-1; j> Index\_position2; j--) {

array[i][j] = array[i-1][j-1];

}

}

array[Index\_position][Index\_position2] = new\_value;

**for**(**int** i=0; i<array.length;i++) {

**for**(**int** j=0;j<array[i].length;j++) {

System.***out***.print(array[i][j] + " ");

}

System.***out***.println("\n");

}

}

**public** **void** copy() {

**int**[][] Array = {{1,5,2,7,3},{9,0,2,6,4}};

**int**[][] Array2 = **new** **int**[Array.length][Array.length];

System.*arraycopy*(Array, 0, Array2, 0, Array.length);

**for**(**int** i=0; i<Array2.length;i++) {

**for**(**int** j=0;j<Array2[i].length;j++) {

System.***out***.print(Array2[i][j] + " ");

}

System.***out***.println("\n");

}

}

**public** **void** delete(**int** n, **int** s) {

**int**[][] array1 = {{1,5,9,5,3},{7,3,0,1,2}};

**for**(**int** i=0; i<array1.length; i++) {

**for**(**int** j=0; j<array1[i].length; j++) {

**if**(array1[i][j]==array1[n][s]) {

**continue**;

}

System.***out***.print(array1[i][j] + " ");

}

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

}

}

**public** **static** **void** main(String[] args) {

PROGRAM3 obj = **new** PROGRAM3();

System.***out***.println("----------ADD METHOD----------\n");

obj.add("shaheer", 0,1);

System.***out***.println("----------INSERT METHOD----------\n");

obj.insert(0, 2, 3);

System.***out***.println("----------COPY METHOD----------\n");

obj.copy();

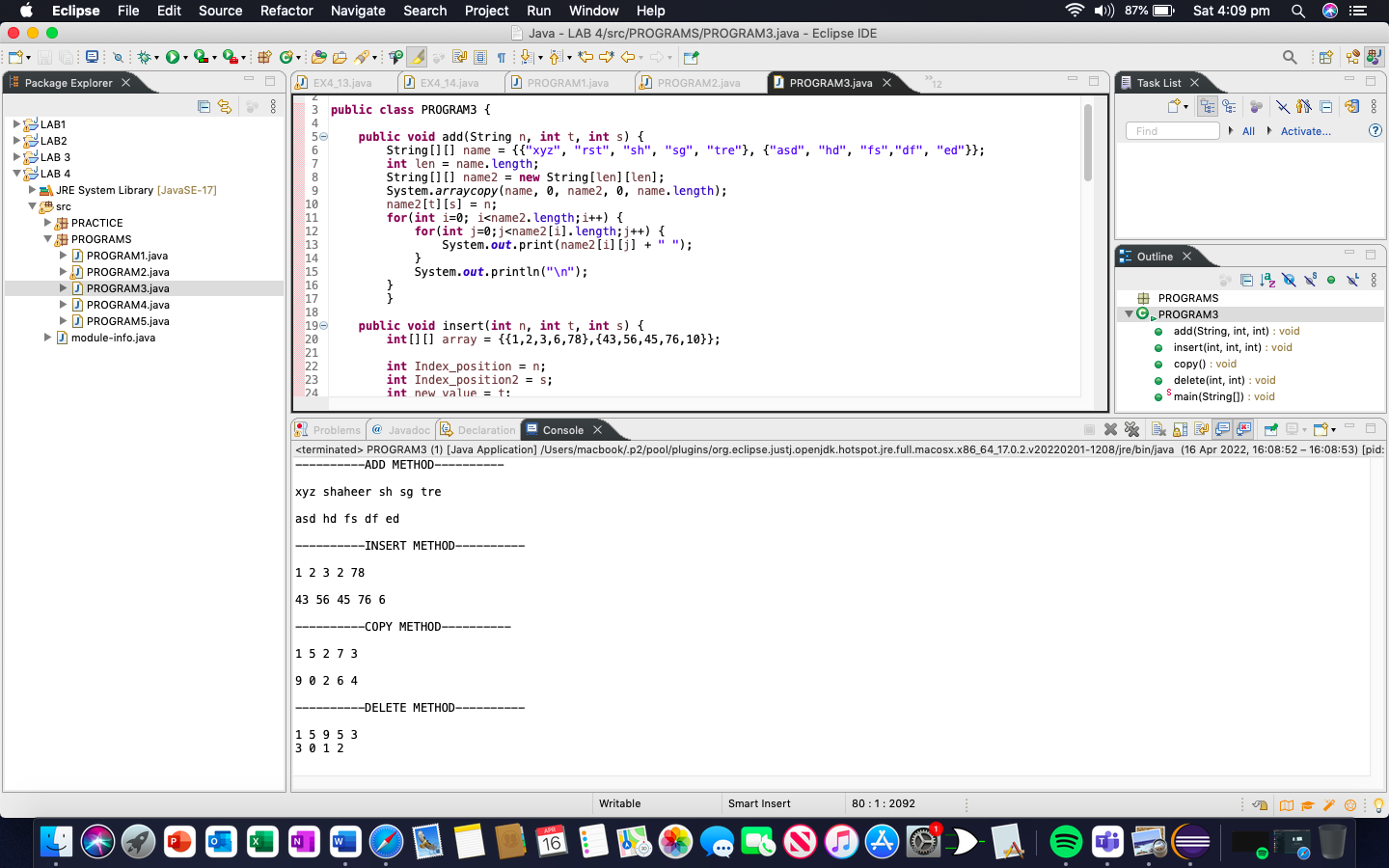
System.***out***.println("----------DELETE METHOD----------\n");

obj.delete(1, 0);

}

}

OUTPUT



PROGRAM4:

**package** PROGRAMS;

**public** **class** PROGRAM4 {

**public** **static** **void** main(String[] args) {

**int**[] DATA = {1,4,9,2,8,10};

**int** K=1;

**int** LOC=1;

**int** max = DATA[1];

**for**(K=0; K<DATA.length; K++) {

**if**(K>DATA[K]) {

LOC+=1;

max = K;

}

**if**(max<DATA[K]) {

LOC=K;

max=DATA[K];

}

}

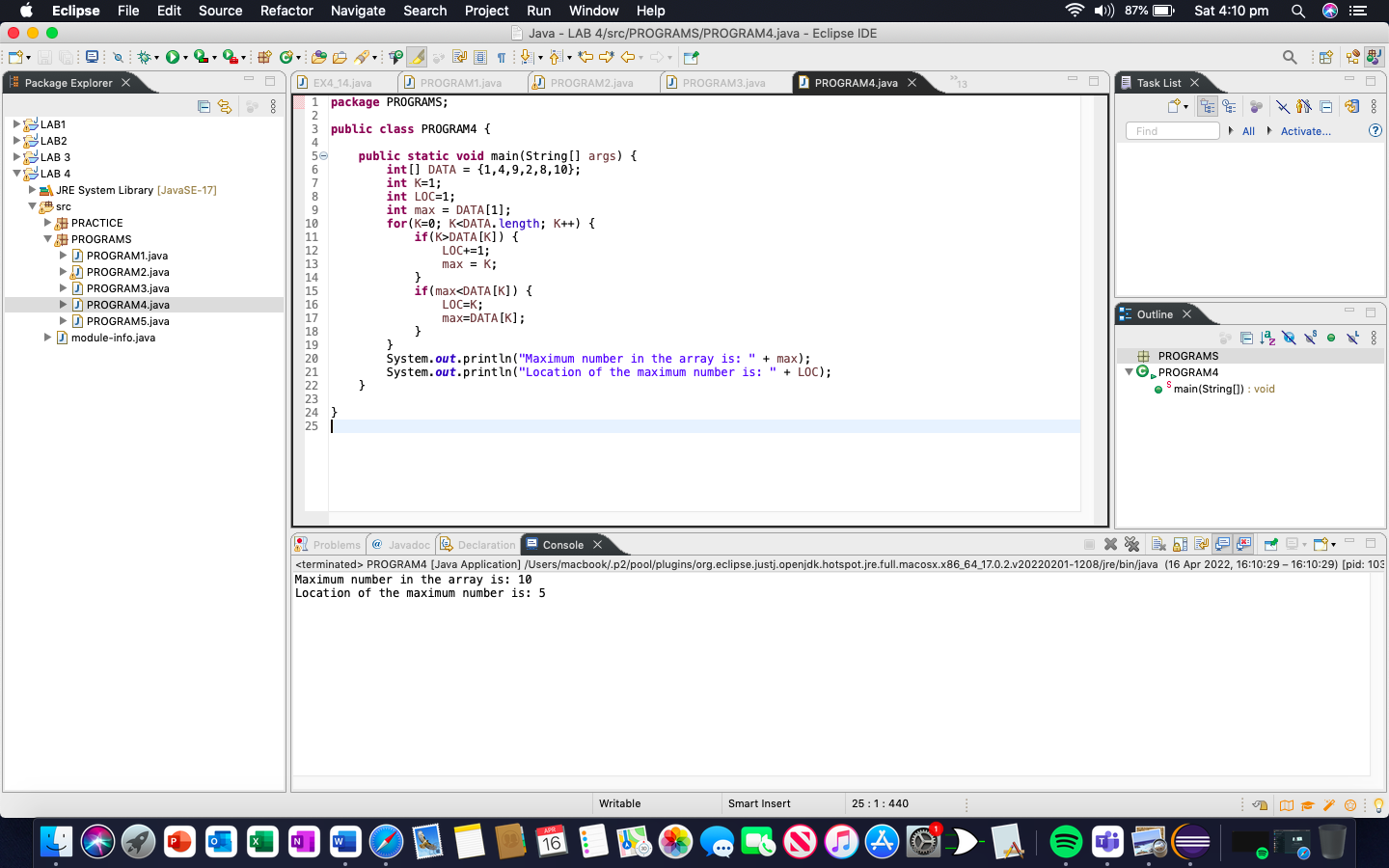
System.***out***.println("Maximum number in the array is: " + max);

System.***out***.println("Location of the maximum number is: " + LOC);

}

}

OUTPUT



PROGRAM5:

ALGORITHM FOR 1D:

(Linear search in 1D array) A non-empty array Array with N numerical values is given. This algorithm traverses the locations of Array. The variable i is used as a counter.  
STEP 1. [Initialize] Set i: = 0, N: = values   
STEP 2. [Increment counter.] Set i: = i+1  
STEP 3. [Test Counter.] If i = N, then:

Write: i and Exit.

STEP 5. [Repeat loop.] Go to Step 2.

ALGORITHM FOR 2D:

(Linear search in 2D array) A non-empty array Array with N numerical values is given. This algorithm traverses the locations of Array. The variable i AND j is used as a counter.  
STEP 1. [Initialize] Set i: = 0, j: = 0, N: = values   
STEP 2. [Increment counter.] Set i: = i+1, j: = j+1  
STEP 3. [Test Counter.] If i & j = N, then:

Write: i and j and Exit.

STEP 5. [Repeat loop.] Go to Step 2.

ALGORITHM FOR 3D:

(Linear search in 3D array) A non-empty array Array with N numerical values is given. This algorithm traverses the locations of Array. The variable i, j AND k is used as a counter.  
STEP 1. [Initialize] Set i: = 0, j: = 0, k: = 0, N: = values   
STEP 2. [Increment counter.] Set i: = i+1, j: = j+1, k: = k+1  
STEP 3. [Test Counter.] If i & j & k= N, then:

Write: i, j and k and Exit.

STEP 5. [Repeat loop.] Go to Step 2.

CODE:

**package** PROGRAMS;

**public** **class** PROGRAM5 {

**public** **void** ONE(**int** n) {

**int**[] array = {1,3,5,3};

**for**(**int** i=0;i<array.length;i++) {

**if**(array[i]==n) {

System.***out***.println(n + ": Location is " + i + "\n");

}

}

}

**public** **void** TWO(**int** n) {

**int**[][] array = {{1,5,2},{2,7,1}};

**for**(**int** i=0; i<array.length; i++) {

**for**(**int** j=0; j<array[i].length; j++) {

**if**(array[i][j]==n) {

System.***out***.println(n + ": is at " + j + " Location of inner array and " + i + " Location of outer array\n");

}

}

}

}

**public** **void** THREE(**int** n) {

**int**[][][] array = {{{1,3,5},{8,78,30},{12,45,34}},{{13,76,40},{15,73,90},{37,29,10}}};

**for**(**int** i=0;i<array.length;i++) {

**for**(**int** j=0; j<array[i].length; j++) {

**for**(**int** k=0; k<array[i][j].length; k++) {

**if**(array[i][j][k]==n) {

System.***out***.println(n + ": is at " + k + " Location of inner most array and at " + j +

" Location of inner array and at " + i + " Location of outer array\n");

}

}

}

}

}

**public** **static** **void** main(String[] args) {

PROGRAM5 obj = **new** PROGRAM5();

System.***out***.println("----------1D ARRAY----------\n");

obj.ONE(5);

System.***out***.println("----------2D ARRAY----------\n");

obj.TWO(1);

System.***out***.println("----------3D ARRAY----------\n");

obj.THREE(3);

}

}

OUTPUT

