# **Name: Abdurrahman Qureshi**

# **Roll No: 210451**

Practical No: 13

1. **WAP in java to search an element in a 1D Array**

**CODE:**

import java.io.\*;

public class EXP13SearchNInArray{

static void SearchingInArray() throws Exception{

InputStreamReader r=new InputStreamReader(System.in);

BufferedReader br=new BufferedReader(r);

System.out.print("Enter the size of the array: ");

int n = Integer.parseInt(br.readLine());

int[] a = new int[n];

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

System.out.print("Enter value for position [" + (i+1) + "] : ");

a[i] = Integer.parseInt(br.readLine());

//br.readLine();}

System.out.print("Enter element to be searched: ");

int ToBeSearched = Integer.parseInt(br.readLine());

//br.readLine();

int flag = 0;

for (int j : a) {

if (j == ToBeSearched) {

flag++;

} else {

continue; }}

if (flag != 0) {

System.out.println("Element exists..."); }

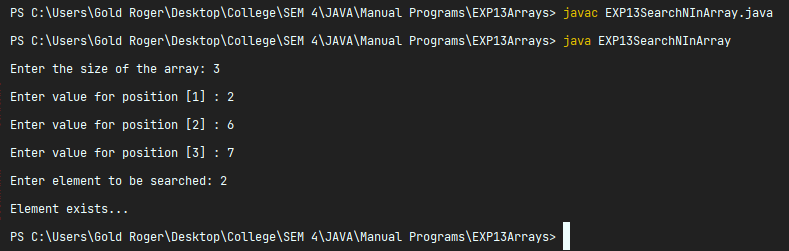
else{

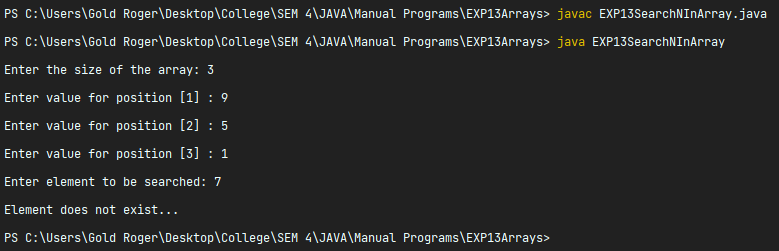
System.out.println("Element does not exist..."); }}

public static void main(String[] args) throws Exception{

SearchingInArray();}}

**OUTPUT:**

****

****

**2) WAP in java to sort a 1D array**

**CODE:**

import java.io.\*;

public class EXP13SortingArray{

static void SortingInArray() throws Exception{

InputStreamReader r=new InputStreamReader(System.in);

BufferedReader br=new BufferedReader(r);

System.out.print("Enter the size of the array: ");

int n = Integer.parseInt(br.readLine());

int[] a = new int[n];

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

System.out.print("Enter value for position [" + (i+1) + "] : ");

a[i] = Integer.parseInt(br.readLine());}

for (int i = 0 ; i < a.length-1 ; i++){

for (int j = i+1 ; j < a.length ; j++){

if (a[i] > a[j]) {

int temp = a[i];

a[i] = a[j];

a[j] = temp; }}}

System.out.println("After Sorting :");

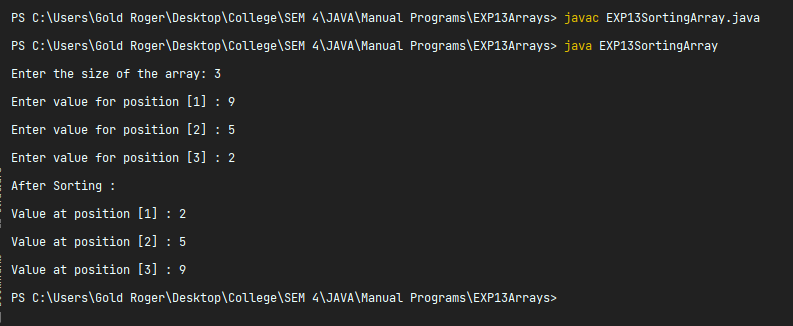
for (int i = 0 ; i < a.length ; i++){

System.out.println("Value at position [" + (i+1) + "] : " + a[i]); }}

public static void main(String[] args) throws Exception{

SortingInArray();}}

**OUTPUT:**



1. **WAP in java to calculate the sum and average of a 1D array**

**CODE:**

import java.io.\*;

public class EXP13ArraySumAvg{

static void SumAvgArray() throws Exception {

InputStreamReader r=new InputStreamReader(System.in);

BufferedReader br=new BufferedReader(r);

System.out.print("Enter the size of the array: ");

int n = Integer.parseInt(br.readLine());

int[] a = new int[n];

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

System.out.print("Enter value for position [" + (i+1) + "] : ");

a[i] = Integer.parseInt(br.readLine());}

int sum = 0;

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

sum += a[i]; }

int avg = (sum / a.length);

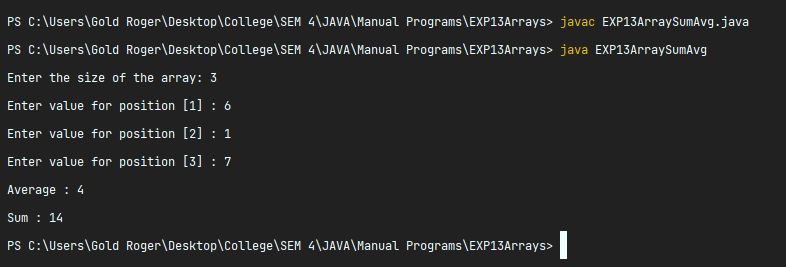
System.out.println("Average : " + avg);

System.out.println("Sum : " + sum); }

public static void main(String[] args) throws Exception{

SumAvgArray();}}

**OUTPUT:**



1. **WAP in java to find smallest and largest in a 1D array**

**CODE:**

import java.io.\*;

public class EXP13LarSmaArray{

static void LargestSmallestArray() throws Exception {

InputStreamReader r = new InputStreamReader(System.in);

BufferedReader br = new BufferedReader(r);

System.out.print("Enter the size of the array: ");

int n = Integer.parseInt(br.readLine());

int[] a = new int[n];

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

System.out.print("Enter value for position [" + (i + 1) + "] : ");

a[i] = Integer.parseInt(br.readLine());}

int largest = a[0];

for (int i = 0 ; i < a.length-1 ; i++){

for (int j = i+1 ; j < a.length ; j++){

if (a[i] > a[j]) {

int temp = a[i];

a[i] = a[j];

a[j] = temp; }}}

int smallest = a[0];

for (int i = 1 ; i < a.length ; i++){

if (a[i] > largest) {

largest = a[i]; }}

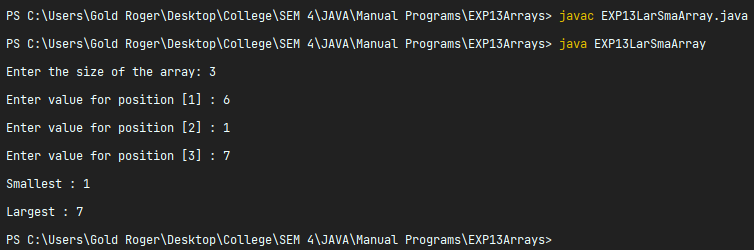
System.out.println("Smallest : " + smallest);

System.out.println("Largest : " + largest); }

public static void main(String[] args) throws Exception{

LargestSmallestArray();}}

**OUTPUT:**



1. **WAP in java to count occurrences of an element in a 1D array**

**CODE:**

import java.io.\*;

public class EXP13CountingOccurances{

static void CountingElement() throws Exception{

InputStreamReader r=new InputStreamReader(System.in);

BufferedReader br=new BufferedReader(r);

System.out.print("Enter the size of the array: ");

int n = Integer.parseInt(br.readLine());

int[] a = new int[n];

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

System.out.print("Enter value for position [" + (i+1) + "] : ");

a[i] = Integer.parseInt(br.readLine());}

System.out.print("Enter element to be counted: ");

int ToBeSearched = Integer.parseInt(br.readLine());

//br.readLine();

int flag = 0;

for (int j : a) {

if (j == ToBeSearched) {

flag++;

} else {

continue; }}

if (flag == 0 ){

System.out.println("Element does not exist..."); }

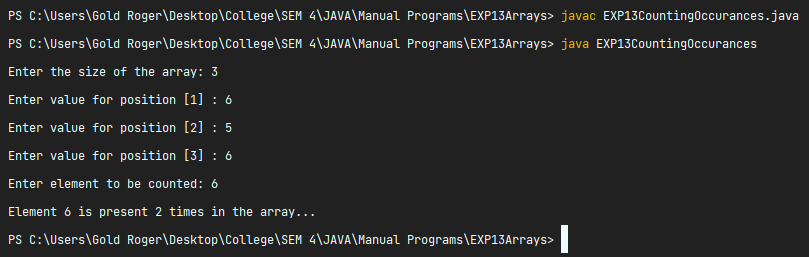
else {

System.out.println("Element " + ToBeSearched + " is present " + flag + " times in the array..."); }}

public static void main(String[] args) throws Exception{

CountingElement();}}

**OUTPUT:**



1. **WAP in java to transpose a (mXn) matrix**

**CODE:**

public class EXP13MatrixTranspose {

static void MatTranspose() throws Exception {

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

int[][] transposeMat = new int[3][3];

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

for (int j = 0; j < ogMat.length; j++) {

transposeMat[i][j] = ogMat[j][i]; }}

System.out.println("Original Matrix : ");

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

for (int j = 0; j < ogMat.length; j++) {

System.out.print(ogMat[i][j] + "\t");}

System.out.println();}

System.out.println("Transposed Matrix : ");

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

for (int j = 0; j < transposeMat.length; j++) {

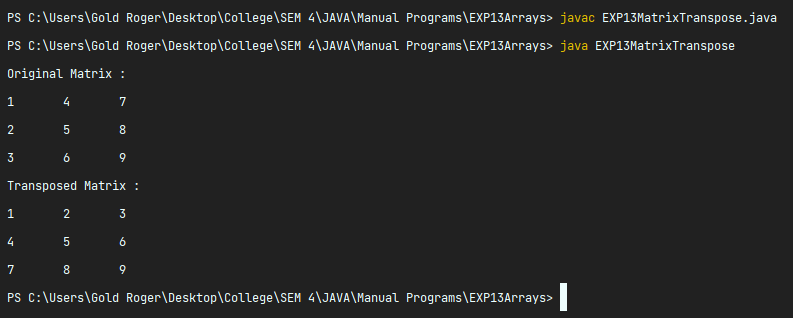
System.out.print(transposeMat[i][j] + "\t");}

System.out.println();}}

public static void main(String[] args) throws Exception {

MatTranspose();}}

**OUTPUT:**



1. **WAP in java to multiply two a (mXn) matrix**

**CODE:**

public class EXP13ArrayMul {

static void MulArray() throws Exception {

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

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

int[][] arrayC = new int[3][3];

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

for (int j = 0; j < arrayA.length; j++) {

arrayC[i][j] = arrayA[i][j] \* arrayB[i][j]; }}

System.out.println("Matrix A : ");

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

for (int j = 0; j < arrayA.length; j++) {

System.out.print(arrayA[i][j] + "\t");}

System.out.println();}

System.out.println("Matrix B : ");

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

for (int j = 0; j < arrayB.length; j++) {

System.out.print(arrayB[i][j] + "\t");}

System.out.println();}

System.out.println("Matrix after multiplication : ");

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

for (int j = 0; j < arrayC.length; j++) {

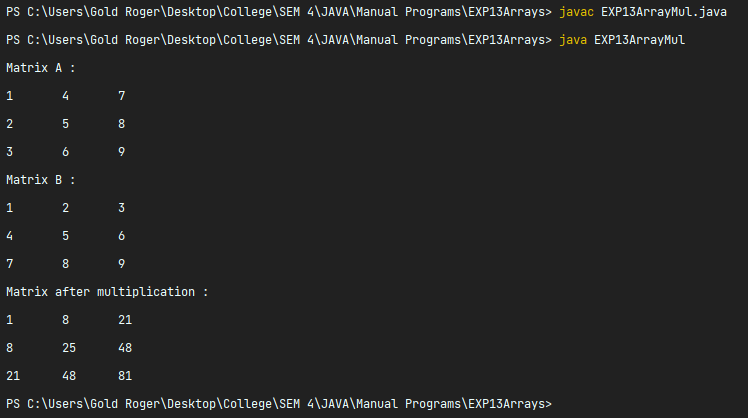
System.out.print(arrayC[i][j] + "\t");}

System.out.println();}}

public static void main(String[] args) throws Exception {

MulArray();}}

**OUTPUT:**



1. **WAP in java to add two a (mXn) matrix**

**CODE:**

public class EXP13ArrayAdd {

static void addArray() throws Exception {

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

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

int[][] arrayC = new int[3][3];

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

for (int j = 0; j < arrayA.length; j++) {

arrayC[i][j] = arrayA[i][j] + arrayB[i][j]; }}

System.out.println("Matrix A : ");

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

for (int j = 0; j < arrayA.length; j++) {

System.out.print(arrayA[i][j] + "\t");}

System.out.println();}

System.out.println("Matrix B : ");

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

for (int j = 0; j < arrayB.length; j++) {

System.out.print(arrayB[i][j] + "\t");}

System.out.println();}

System.out.println("Matrix after Addition : ");

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

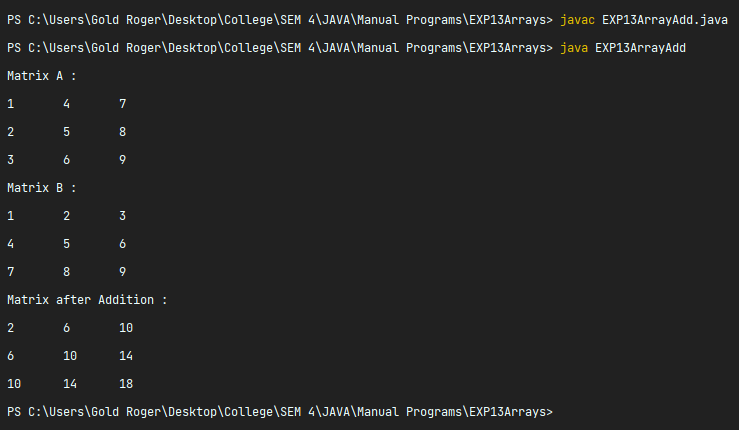
for (int j = 0; j < arrayC.length; j++) {

System.out.print(arrayC[i][j] + "\t");}

System.out.println();}}

public static void main(String[] args) throws Exception {addArray();}}

**OUTPUT:**



1. **WAP in java to calculate sum of diagonals of (mXn) matrix**

**CODE:**

public class EXP13SumOfDiagonals {

static void DiagonalsAdd() throws Exception {

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

int sumOfDiagonals = 0;

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

for (int j = 0; j < arrayA.length; j++) {

if (i == j) {

sumOfDiagonals += arrayA[i][j]; }}}

System.out.println("Matrix A : ");

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

for (int j = 0; j < arrayA.length; j++) {

System.out.print(arrayA[i][j] + "\t");}

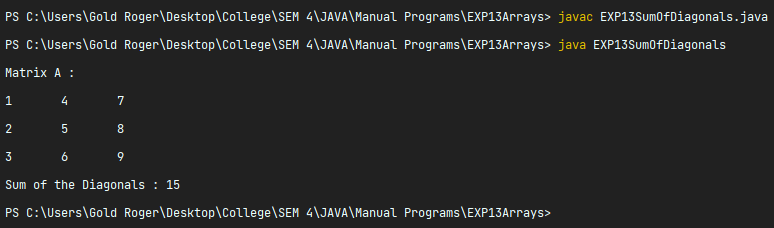
System.out.println();}

System.out.println("Sum of the Diagonals : " + sumOfDiagonals); }

public static void main(String[] args) throws Exception {

DiagonalsAdd();}}

**OUTPUT:**



1. **WAP in java to calculate sum and average of (mXn) matrix**

**CODE:**

public class EXP13SumAvgMxN {

static void SumAvgMxN() throws Exception {

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

int sum = 0;

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

for (int j = 0; j < arrayA.length; j++) {

sum += arrayA[i][j]; }}

System.out.println("Matrix A : ");

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

for (int j = 0; j < arrayA.length; j++) {

System.out.print(arrayA[i][j] + "\t");}

System.out.println();}

int avg = (sum / 9);

System.out.println("Sum : " + sum);

System.out.println("Average : " + avg); }

public static void main(String[] args) throws Exception {

SumAvgMxN();}}

**OUTPUT:**

