**Experiment – 2**

Q1) Write a program that generate 6\*6 two-dimensional matrix, filled with 0’s and 1’s , display the matrix, check every raw and column have an odd number’s of 1’s

Ans:

**Program:**

package Exp\_2;

import java.util.Scanner;

public class ZeroesOnes {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int[][] arr = new int[6][6];

        System.out.println("Enter the array: ");

        for (int i =0;i<6;i++){

            for(int j=0;j<6;j++){

                arr[i][j] = sc.nextInt();

            }

        }

        for (int i =0;i<6;i++){

            int no1 = 0;

            for(int j=0;j<6;j++){

                if(arr[i][j]==1) no1++;

            }

            if(no1 % 2 != 0){

                System.out.println("Row number "+ (i+1) + " has odd number of ones. " +

                        "Number of ones are : " + no1);

            }

        }

        for (int i =0;i<6;i++){

            int no2 = 0;

            for(int j=0;j<6;j++){

                if(arr[j][i]==1) no2++;

            }

            if(no2 % 2 != 0){

                System.out.println("Column number "+ (i+1) + " has odd number of ones. " +

                        "Number of ones are : " + no2);

            }

        }

        sc.close();

    }

}

**Output:**

A screenshot of a computer

AI-generated content may be incorrect.

Q2) Write a generic method that returns the minimum elements and their indices in a two dimensional array.

Ans:

**Program:**

package Exp\_2;

import java.util.Scanner;

public class Min2D {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter number of rows and columns: ");

        int n = sc.nextInt();

        int m = sc.nextInt();

        int[][] arr = new int[n][m];

        System.out.println("Enter array elements:");

        for(int i=0;i<n;i++){

            for(int j=0;j<m;j++){

                arr[i][j] = sc.nextInt();

            }

        }

        for(int i=0;i<n;i++){

            int min = arr[i][0];

            int idxRow = 0;

            int idxCol = 0;

            for(int j=0;j<m;j++){

                if(min>arr[i][j]){

                    min = arr[i][j];

                    idxRow = i;

                    idxCol = j;

                }

            }

            System.out.println("Minimum element in row "+i+" is "+min+" at index ("+idxRow+","+idxCol+")");

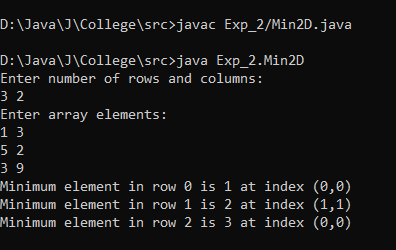
        }

        sc.close();

    }

}

**Output:**



Q3) Write a method that returns a new array by eliminating the duplicate values in the array.

Ans:

**Program:**

package Exp\_2;

import java.util.Scanner;

public class EliminateDuplicateElements {

public static int[] removeDuplicates(int[] arr) {

int n = arr.length;

int[] temp = new int[n];

int a = 0;

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

boolean isDuplicate = false;

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

if (temp[j] == arr[i]) {

isDuplicate = true;

break;

}

}

if (!isDuplicate) {

temp[a] = arr[i];

a++;

}

}

int[] result = new int[a];

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

result[i] = temp[i];

}

return result;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

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

int n = sc.nextInt();

int[] arr = new int[n];

System.out.println("Enter array elements: ");

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

arr[i] = sc.nextInt();

}

int[] uniqueArr = removeDuplicates(arr);

System.out.println("Array after removing duplicates: ");

for (int num : uniqueArr) {

System.out.print(num + " ");

}

sc.close();

}

}

**Output:**

A computer screen with white text

AI-generated content may be incorrect.

Q4) Write a program to add, subtract or multiply two 3\*3 integer arrays as per choice of user.

Sample Input:

Array 1:

1 2 3

4 5 6

7 8 9

Array 2:

5 6 7

1 2 0

4 3 2

Symbol: +

Sample Output:

6 8 10

5 7 6

11 11 11

Ans:

**Program:**

package Exp\_2;

import java.util.Scanner;

public class AddMultiplySubtract {

    public static void printArray(int[][] arr){

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

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

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

            }

            System.out.print("\n");

        }

    }

    public static void addMatrix(int[][] arr, int[][] brr){

        int n = arr.length;

        int m = arr[0].length;

        int[][] crr = new int[n][m];

        for(int i=0;i<n;i++){

            for(int j=0;j<m;j++){

                crr[i][j] = arr[i][j] + brr[i][j];

            }

        }

        printArray(crr);

    }

    public static void subtractMatrix(int[][] arr, int[][] brr){

        int n = arr.length;

        int m = arr[0].length;

        int[][] crr = new int[n][m];

        for(int i=0;i<n;i++){

            for(int j=0;j<m;j++){

                crr[i][j] = arr[i][j] - brr[i][j];

            }

        }

        printArray(crr);

    }

    public static void multiplyMatrix(int[][] arr, int[][] brr){

        int n = arr.length;

        int m = arr[0].length;

        int[][] crr = new int[n][m];

        for(int i=0;i<n;i++){

            for(int j=0;j<m;j++){

                crr[i][j] = 0;

                for(int k=0;k<n;k++){

                    crr[i][j] += arr[i][k] \* brr[k][j];

                }

            }

        }

        printArray(crr);

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int n=3,m=3;

        System.out.println("Enter array 1 elements: ");

        int[][] arr = new int[n][m];

        for(int i=0;i<n;i++){

            for(int j=0;j<m;j++){

                arr[i][j] = sc.nextInt();

            }

        }

        System.out.println("Enter array 2 elements: ");

        int[][] brr = new int[n][m];

        for(int i=0;i<n;i++){

            for(int j=0;j<m;j++){

                brr[i][j] = sc.nextInt();

            }

        }

        System.out.println("Enter operator (+,-,\*): ");

        String ch = sc.next();

        if(ch.equals("+")){

            addMatrix(arr, brr);

        }

        else if(ch.equals("-")){

            subtractMatrix(arr, brr);

        }

        else if(ch.equals("\*")){

            multiplyMatrix(arr,brr);

        }

        sc.close();

    }

}

**Output:**

A computer screen shot of a program

AI-generated content may be incorrect.

Q5) Write a program to sort an array of 10 elements using selection sort.

Ans:

**Program:**

package Exp\_2;

import java.util.Scanner;

public class SelectionSort {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int[] arr = new int[10];

        System.out.println("Enter array elements: ");

        for (int i = 0; i < 10; i++) {

            arr[i] = sc.nextInt();

        }

        for(int i=0;i<9;i++){ // taken 9 as no need to sort last element

            int minIdx = i;

            for(int j=i+1;j<10;j++){

                if (arr[j] < arr[minIdx]) {

                    minIdx = j;

                }

            }

            int temp = arr[minIdx];

            arr[minIdx] = arr[i];

            arr[i] = temp;

        }

        System.out.println("Sorted Array:");

        for (int i = 0; i < 10; i++) {

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

        }

        sc.close();

    }

}

**Output:**

A screen shot of a computer program

AI-generated content may be incorrect.

Q6) Write a program that prompts the user to enter a string and displays the number of vowels and consonants in the string.

Ans:

**Program:**

package Exp\_2;

import java.util.Scanner;

public class VowelsConsonants {

    public static void findVowelsConsonants(String s){

        int noOfVowels = 0;

        int noOfConsonants = 0;

        int n = s.length();

        s = s.toLowerCase();

        s = s.trim();

        for(int i=0;i<n;i++){

            char ch = s.charAt(i);

            if(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u'){

                noOfVowels += 1;

            }

            else{

                noOfConsonants += 1;

            }

        }

        System.out.println("Number of vowels: "+noOfVowels);

        System.out.println("Number of consonants: "+noOfConsonants);

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter a string");

        String s = sc.nextLine();

        findVowelsConsonants(s);

        sc.close();

    }

}

**Output:**

A screen shot of a computer code

AI-generated content may be incorrect.

Q7) Write a program that prompts the user to enter two strings and displays the largest common prefix of the two strings

Ans:

**Program:**

package Exp\_2;

import java.util.Scanner;

public class LargestCommonPrefix {

    public static String LCP(String s1, String s2){

        String lcp = "";

        int n = s1.length();

        int m = s2.length();

        int i=0;

        int j=0;

        if(n>m){

            while(i<m){

                if(s1.charAt(i) == s2.charAt(j)){

                    lcp = lcp + s1.charAt(i);

                }

                else{

                    break;

                }

                i++;

                j++;

            }

        }

        else if(n<m){

            while(j<n){

                if(s1.charAt(i) == s2.charAt(j)){

                    lcp = lcp + s2.charAt(j);

                }

                else{break;}

                i++;

                j++;

            }

        }

        else{

            while(i!=n){

                if(s1.charAt(i) == s2.charAt(j)){

                    lcp = lcp + s1.charAt(i);

                }

                else{break;}

                i++;

                j++;

            }

        }

        return lcp;

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter first string: ");

        String s1 = sc.next();

        System.out.print("Enter second string: ");

        String s2 = sc.next();

        String lcp = LCP(s1,s2);

        if(lcp.isEmpty()) System.out.println("No common prefix");

        else   System.out.println("Largest common prefix: "+lcp);

        sc.close();

    }

}

**Output:**

A screen shot of a computer program

AI-generated content may be incorrect.

Q8) Some websites impose certain rules for passwords. Write a method that checks whether a string is a valid password. Suppose the password rules are as follows: A password must have at least eight characters. A password consists of only letters and digits. A password must contain at least two digits. Write a program that prompts the user to enter a password and displays Valid Password if the rules are followed or Invalid Password otherwise.

Ans:

**Program:**

package Exp\_2;

import java.util.Scanner;

public class Password {

    public static void checkPass(String s){

        if(s.length()<8){

            System.out.println("Invalid password! Password should contain atleast 8 characters ");

            return;

        }

        int countDigits = 0;

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

            char ch = s.charAt(i);

            if(Character.isDigit(ch)){

                countDigits += 1;

            }

        }

        if(countDigits<2){

            System.out.println("Invalid password! Password should contain atleast 2 digits");

            return;

        }

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

            char ch = s.charAt(i);

            if(!Character.isLetterOrDigit(ch)){

                System.out.println("Invalid password! Password should contain only a digit or an alphabet  ");

                return;

            }

        }

        System.out.println("Your password is valid ");

        return;

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter your password: ");

        String s = sc.next();

        checkPass(s);

        sc.close();

    }

}

**Output:**

A computer screen shot of white text

AI-generated content may be incorrect.