Advance Programming (Object Oriented Programming Lab)
Assignment-3
Due Date -1 Week

1. Create an array of integers with a size of 10. Fill the array with random integer valuesbetween 1 and 100 (inclusive).
    a) Calculate and display the sum of all the elements in the array
    b) Find and display the maximum and minimum values in the array
    c) Calculate and display the average of the array elements
    d) Search for a specific integer value entered by the user. Display whether the valueis present in the array or not

2. Create two arrays of integers, each with a size of 5.
    a. Populate the arrays with random integer values between 1 and 500 (inclusive)
    b. Compare the two arrays and find the number of elements that are common in both arrays
    c. Calculate and display the sum of the common elements
    d. Determine and display the average of the non-common elements.

---

3. Create a 3x3 matrix (2D array) of integers and do the following operations
    a. Fill the matrix with random integer values between 1 and 50 (inclusive) anddisplay the matrix.
    b. Calculate and display the sum of the main diagonal elements (elements where therow index is equal to the column index).
    c. Determine and display the product of the secondary diagonal elements (elementswhere the row index + column index = size of the matrix - 1).

4. Write a program to create an array and sort the array by using any sorting algorithmhaving average case time complexity $O(n^2)$. Display all the intermediate steps of the sorting procedure

---

5. Design a class **Student** that represents a student's information including their name andan array of test scores (at least 5 scores).
    a) Implement methods to:
        a. Calculate and return the average test score.
        b. Determine and return the highest and lowest test scores.
        c. Display the student's information along with their test scores.
    b) Design a class **StudentGradingSystem** that maintains an array of **Student** objects.
    c) Implement methods to:
        a. Add a new student to the system.
        b. Display the student with the highest average score.
        c. Display the student with the lowest average score.
        d. Display the student(s) with the highest test score.
        e. Display the student(s) with the lowest test score

    Demonstrate the use of the **StudentGradingSystem** class by adding students and performing therequired operations.