Tutorial – 2

1. Write a C# Sharp program that stores elements in an array and prints them.

```csharp
using System;

class Q1
{
    static void Main()
    {
        //stoeres elements in an array  and prints them
        int size;
        Console.Write("Enter the size of the array: ");
        try
        {
            size = Convert.ToInt32(Console.ReadLine());
            if (size <= 0)
            {
                Console.WriteLine("Size must be a positive integer.
Defaulting to 10.");
                size = 10;
            }
        }
        catch (FormatException)
        {
            Console.WriteLine("Invalid input! Defaulting size to 10.");
            size = 10;
        }
        int[] arr = new int[size];
        for (int i = 0; i < size; i++) {
            Console.Write($"Element {i + 1}: ");
            try
            {
                arr[i] = Convert.ToInt32(Console.ReadLine());
            }
            catch (FormatException)
            {
```

```
                Console.WriteLine("Invalid input! Please enter a valid
    integer.");
                i--; // Decrement i to repeat this iteration
            }
        }
        Console.WriteLine("All elements stored successfully.");
        foreach (int element in arr)
        {
            Console.WriteLine($"Stored element: {element}");
        }
    }
    }
```

Output:

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q1
Enter the size of the array: 5
Element 1: 50
Element 2: 40
Element 3: 30
Element 4: 8
Element 5: 6
All elements stored successfully.
Stored element: 50
Stored element: 40
Stored element: 30
Stored element: 8
Stored element: 6
```

2. Write a program of sorting an array. Declare single dimensional array and accept 5 integer values from the user. Then sort the input in ascending order and display output.

```
using System;

class Q2
{
    static void Main()
```

```csharp
{
    //sorting element
    int size;
    Console.Write("Enter the size of the array: ");
    try { size = Convert.ToInt32(Console.ReadLine()); }
    catch
    {
        Console.WriteLine("Invalid input! Defaulting size to 10.");
        size = 10;
    }
    if (size <= 0)
    {
        Console.WriteLine("Size must be a positive integer. Defaulting
to 10.");
        size = 10;
    }
    int[] arr = new int[size];
    Console.WriteLine("Enter elements in the array:");
    for (int i = 0; i < arr.Length; i++)
    {
        try
        {
            Console.Write($"Element {i + 1}: ");
            arr[i] = Convert.ToInt32(Console.ReadLine());
        }
        catch (FormatException)
        {
            Console.WriteLine("Invalid input! Please enter a valid
integer.");
            i--; // Decrement i to repeat this iteration
        }
    }
    //soring arry without built-in sort method
    for (int i = 0; i < arr.Length - 1; i++)
    {
        for (int j = i + 1; j < arr.Length; j++)
        {
            if (arr[i] > arr[j])
            {
                // Swap arr[i] and arr[j]
```

```csharp
                    int temp = arr[i];
                    arr[i] = arr[j];
                    arr[j] = temp;
                }
            }
        }
        Console.WriteLine("Sorted array:");

        for (int i = 0; i < arr.Length; i++)
        {
            Console.Write(arr[i] + " ");
        }

    }

}
```

Output

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q2
Enter the size of the array: 5
Enter elements in the array:
Element 1: 12
Element 2: 32
Element 3: 85
Element 4: 4
Element 5: 36
Sorted array:

4 12 32 36 85
```

3. Write a C# Sharp program to read n values in an array and display them in reverse order.

```csharp
using System;
using System.Runtime.InteropServices;
```

```csharp
class Q3
{
    static void Main()
    {
        //revers arrary order
        int size;
        Console.Write("Enter the size of the array: ");

        try
        {
            size = Convert.ToInt32(Console.ReadLine());
            if (size <= 0)
            {
                Console.WriteLine("Size must be a positive integer.
Defaulting to 10.");
                size = 10;
            }
        }
        catch (FormatException)
        {
            Console.WriteLine("Invalid input! Defaulting size to 10.");
            size = 10;
        }
        int[] arr = new int[size];
        Console.WriteLine("Enter elements in the array:");
        for (int i = 0; i < arr.Length; i++)
        {
            try
            {
                Console.Write($"Element {i + 1}: ");
                arr[i] = Convert.ToInt32(Console.ReadLine());
            }
            catch (FormatException)
            {
                Console.WriteLine("Invalid input! Please enter a valid
integer.");
                i--; // Decrement i to repeat this iteration
            }
        }
        Console.WriteLine("Reversed array:");
```

```csharp
        int[] reversedArr = new int[size];
        for (int i = arr.Length - 1; i >= 0; i--)
        {
            reversedArr[arr.Length - 1 - i] = arr[i];
            Console.Write(arr[i] + " ");
        }

    }
}
```

Output

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q3
Enter the size of the array: 5
Enter elements in the array:
Element 1: 12
Element 2: 65
Element 3: 22
Element 4: 21
Element 5: 58
Reversed array:
58 21 22 65 12
Reversed array using a new array:
58 21 22 65 12
```

4. Write a C# Sharp program to copy the elements of one array into another array.

```csharp
using System;
using System.Reflection;

class Q4
{
    static void Main()
    {
        // copy one arry to another
        int size;
          Console.Write("Enter the size of the array: ");
        size = Convert.ToInt32(Console.ReadLine());
        int[] arr1 = new int[size];
```

```csharp
            int[] arr2 = new int[size];
            Console.WriteLine("Enter the elements of the array:");
            for (int i = 0; i < size; i++)
            {
                Console.Write("Element {0}: ", i + 1);
                arr1[i] = Convert.ToInt32(Console.ReadLine());
            }
            // Copying arr1 to arr2
            for (int i = 0; i < size; i++)
            {
                arr2[i] = arr1[i];
            }
            // Displaying the copied array
            Console.WriteLine("Copied array elements:");
            for (int i = 0; i < size; i++)
            {
                Console.WriteLine("Element {0}: {1}", i + 1, arr2[i]);
            }
        }
    }
```

Output

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q4
Enter the size of the array: 5
Enter the elements of the array:
Element 1: 12
Element 2: 15
Element 3: 13
Element 4: 48
Element 5: 21
Copied array elements:
Element 1: 12
Element 2: 15
Element 3: 13
Element 4: 48
Element 5: 21
```

5. Write a C# Sharp program to count duplicate elements in an array.

```csharp
using System;
using System.ComponentModel;

class Q5
{
    static void Main()
    {
        int size;
        Console.Write("Enter the size of the array: ");
        size = Convert.ToInt32(Console.ReadLine());
        int[] arr = new int[size];
        Console.WriteLine("Enter the elements of the array:");
        for (int i = 0; i < size; i++)
        {
            Console.Write("Element {0}: ", i + 1);
            arr[i] = Convert.ToInt32(Console.ReadLine());
        }
        int[] count = new int[size];
        for (int i = 0; i < size; i++)
        {
            count[i] = 1;
            for (int j = i + 1; j < size; j++)
            {
                if (arr[i] == arr[j])
                {
                    count[i]++;
                    arr[j] = -1;
                }
            }
        }
        Console.WriteLine("Duplicate elements and their counts:");

        for (int i = 0; i < size; i++)
        {
            if (arr[i] != -1 && count[i] > 1)
            {
                Console.WriteLine("Element {0} occurs {1} times", arr[i],
count[i]);
```

```
            }
        }


    }
}
```

Output

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q5
Enter the size of the array: 5
Enter the elements of the array:
Element 1: 12
Element 2: 13
Element 3: 85
Element 4: 12
Element 5: 22
Duplicate elements and their counts:
Element 12 occurs 2 times
```

---

6. Write a C# Sharp program to find the maximum and minimum elements in an array.

```csharp
using System;
using System.ComponentModel;

class Q6
{
    static void Main()
    {
        int size;
        Console.Write("Enter the size of the array: ");
        size = Convert.ToInt32(Console.ReadLine());
        int[] arr = new int[size];
        Console.WriteLine("Enter the elements of the array:");
        for (int i = 0; i < size; i++)
        {
            Console.Write("Element {0}: ", i + 1);
            arr[i] = Convert.ToInt32(Console.ReadLine());
        }
```

```csharp
        // Initialize max and min with the first element
        int max = arr[0];
        int min = arr[0];
        // Find max and min

        Console.WriteLine("Finding maximum and minimum elements in the
    array:");
        for (int i = 1; i < size; i++)
        {
            if (arr[i] > max)
            {
                max = arr[i];
            }
            if (arr[i] < min)
            {
                min = arr[i];
            }
        }
        // Display the results
        Console.WriteLine("Maximum element in the array: " + max);
        Console.WriteLine("Minimum element in the array: " + min);

    }
}
```

Output

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q6
Enter the size of the array: 5
Enter the elements of the array:
Element 1: 54
Element 2: 89
Element 3: 22
Element 4: 11
Element 5: 8
Finding maximum and minimum elements in the array:
Maximum element in the array: 89
Minimum element in the array: 8
```

7. Write a program in C# Sharp to separate odd and even integers into separate arrays.

```csharp
using System;
using System.ComponentModel;

class Q7
{
    static void Main()
    {
        //Write a program in C# Sharp to separate odd and even integers
into separate arrays
        int size;
        Console.Write("Enter the size of the array: ");
        size = Convert.ToInt32(Console.ReadLine());
        int[] arr = new int[size];
        int evenCount = 0, oddCount = 0;
        Console.WriteLine("Enter the elements of the array:");
        for (int i = 0; i < size; i++)
        {
            Console.Write("Element {0}: ", i + 1);
            arr[i] = Convert.ToInt32(Console.ReadLine());
            if (arr[i] % 2 == 0)
            {
                evenCount++;
            }
            else
            {
                oddCount++;
            }
        }
        // Create separate arrays for even and odd integers
        int[] evenArr = new int[evenCount];
        int[] oddArr = new int[oddCount];
        evenCount = 0; // Reset count for filling even array
        oddCount = 0; // Reset count for filling odd array
        Console.WriteLine("Separating odd and even integers into separate
arrays:");
        for (int i = 0; i < size; i++)
        {
            if (arr[i] % 2 == 0)
```

```csharp
            {
                evenArr[evenCount++] = arr[i]; // Fill even array
            }
            else
            {
                oddArr[oddCount++] = arr[i]; // Fill odd array
            }
        }
        // Display the even array
        Console.WriteLine("Even integers:");
        for (int i = 0; i < evenArr.Length; i++)
        {
            Console.Write(evenArr[i] + " ");
        }
        Console.WriteLine();
        // Display the odd array
        Console.WriteLine("Odd integers:");
        for (int i = 0; i < oddArr.Length; i++)
        {
            Console.Write(oddArr[i] + " ");
        }

    }
}
```

Output

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q7
Enter the size of the array: 5
Enter the elements of the array:
Element 1: 45
Element 2: 654
Element 3: 78
Element 4: 21
Element 5: 30
Separating odd and even integers into separate arrays:
Even integers:
654 78 30
Odd integers:
45 21
```

8. Write a C# Sharp program to sort array elements in descending order.

```csharp
using System;
using System.ComponentModel;

class Q8
{
    static void Main()
    {
        int size;
        Console.Write("Enter the size of the array: ");
        size = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Enter the elements of the array:");
        int[] arr = new int[size];
        for (int i = 0; i < size; i++)
        {
            Console.Write("Element {0}: ", i + 1);
            arr[i] = Convert.ToInt32(Console.ReadLine());
        }
        // Sort the array in descending order
        Console.WriteLine("Sorting array elements in descending order:");
        for (int i = 0; i < size - 1; i++)
        {
            for (int j = i + 1; j < size; j++)
            {
                if (arr[i] < arr[j]) // Change to '>' for ascending order
                {
                    // Swap elements
                    int temp = arr[i];
                    arr[i] = arr[j];
                    arr[j] = temp;
                }
            }
        }
        // Display the sorted array
        Console.WriteLine("Sorted array in descending order:");
        for (int i = 0; i < size; i++)
        {
```

```
            Console.Write(arr[i] + " ");
        }

    }
}
```

Output

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q8
Enter the size of the array: 5
Enter the elements of the array:
Element 1: 12
Element 2: 54
Element 3: 32
Element 4: 80
Element 5: 5
Sorting array elements in descending order:
Sorted array in descending order:
80 54 32 12 5
```

9. Write a C# Sharp program to delete an element at the desired position from an array.

```csharp
using System;
using System.ComponentModel;

class Q9
{
    static void Main()
    {
        int size;
        Console.Write("Enter the size of the array: ");
        size = Convert.ToInt32(Console.ReadLine());
        int[] arr = new int[size];
        Console.WriteLine("Enter the elements of the array:");
        for (int i = 0; i < size; i++)
        {
            Console.Write("Element {0}: ", i + 1);
            arr[i] = Convert.ToInt32(Console.ReadLine());
```

```csharp
        }
        // Display the original array
        Console.WriteLine("Original array:");
        for (int i = 0; i < size; i++)
        {
            Console.Write(arr[i] + " ");
        }
        Console.WriteLine();
        // Ask for the position to delete
        Console.Write("Enter the position of the element to delete (1 to
{0}): ", size);
        int position = Convert.ToInt32(Console.ReadLine());
        // Validate position
        if (position < 1 || position > size)
        {
            Console.WriteLine("Invalid position! Please enter a position
between 1 and {0}.", size);
            return;
        }
        // Adjust position for zero-based index
        position--; // Convert to zero-based index
        // Create a new array with one less size
        int[] newArr = new int[size - 1];
        // Copy elements to the new array, skipping the element at the
specified position
        for (int i = 0, j = 0; i < size; i++)
        {
            if (i != position) // Skip the element at the specified
position
            {
                newArr[j++] = arr[i];
            }
        }
        // Display the new array
        Console.WriteLine("Array after deleting element at position {0}:",
position + 1);
        for (int i = 0; i < newArr.Length; i++)
        {
            Console.Write(newArr[i] + " ");
        }
```

```
        }
    }
}
```

Output

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q9
Enter the size of the array: 5
Enter the elements of the array:
Element 1: 21
Element 2: 5
Element 3: 84
Element 4: 2
Element 5: 9
Original array:
21 5 84 2 9
Enter the position of the element to delete (1 to 5): 2
Array after deleting element at position 2:
21 84 2 9
```

---

10. Write a C# Sharp program for adding two matrices of the same size.

```csharp
using System;
using System.ComponentModel;

class Q10
{
    static void Main()
    {
        //Write a C# Sharp program for adding two matrices of the same
size.
        int rows, cols;
        Console.Write("Enter the number of rows: ");
        rows = Convert.ToInt32(Console.ReadLine());
        Console.Write("Enter the number of columns: ");
        cols = Convert.ToInt32(Console.ReadLine());
        int[,] matrix1 = new int[rows, cols];
        int[,] matrix2 = new int[rows, cols];
        int[,] sumMatrix = new int[rows, cols];
```

```csharp
        Console.WriteLine("Enter elements of first matrix:");
        for (int i = 0; i < rows; i++)
        {
            for (int j = 0; j < cols; j++)
            {
                Console.Write($"Element [{i},{j}]: ");
                matrix1[i, j] = Convert.ToInt32(Console.ReadLine());
            }
        }
        Console.WriteLine("Enter elements of second matrix:");
        for (int i = 0; i < rows; i++)
        {
            for (int j = 0; j < cols; j++)
            {
                Console.Write($"Element [{i},{j}]: ");
                matrix2[i, j] = Convert.ToInt32(Console.ReadLine());
            }
        }
        // Adding the two matrices
        Console.WriteLine("Sum of the two matrices:");
        for (int i = 0; i < rows; i++)
        {
            for (int j = 0; j < cols; j++)
            {
                sumMatrix[i, j] = matrix1[i, j] + matrix2[i, j];
                Console.Write(sumMatrix[i, j] + "\t");
            }
            Console.WriteLine();
        }
    }
}
```

Output

```
C:\Users\dharm\source\repos\24SOECE13043_Dharmraj_sodha\LAB2>Q10
Enter the number of rows: 2
Enter the number of columns: 2
Enter elements of first matrix:
Element [0,0]: 51
Element [0,1]: 32
Element [1,0]: 14
Element [1,1]: 6
Enter elements of second matrix:
Element [0,0]: 32
Element [0,1]: 44
Element [1,0]: 56
Element [1,1]: 20
Sum of the two matrices:
83      76
70      26
```