EXPERIMENT - 1

Student Name: Sandeep Kumar **UID: 20BCS4885**

Branch: BE - CSE **Section/Group: 603/A**

Semester: 6th semester **Subject: Web Programming Using .Net**

Aim:

Implement a recursive algorithm for finding the maximum element in an arbitrary (unsorted) array.

Code:

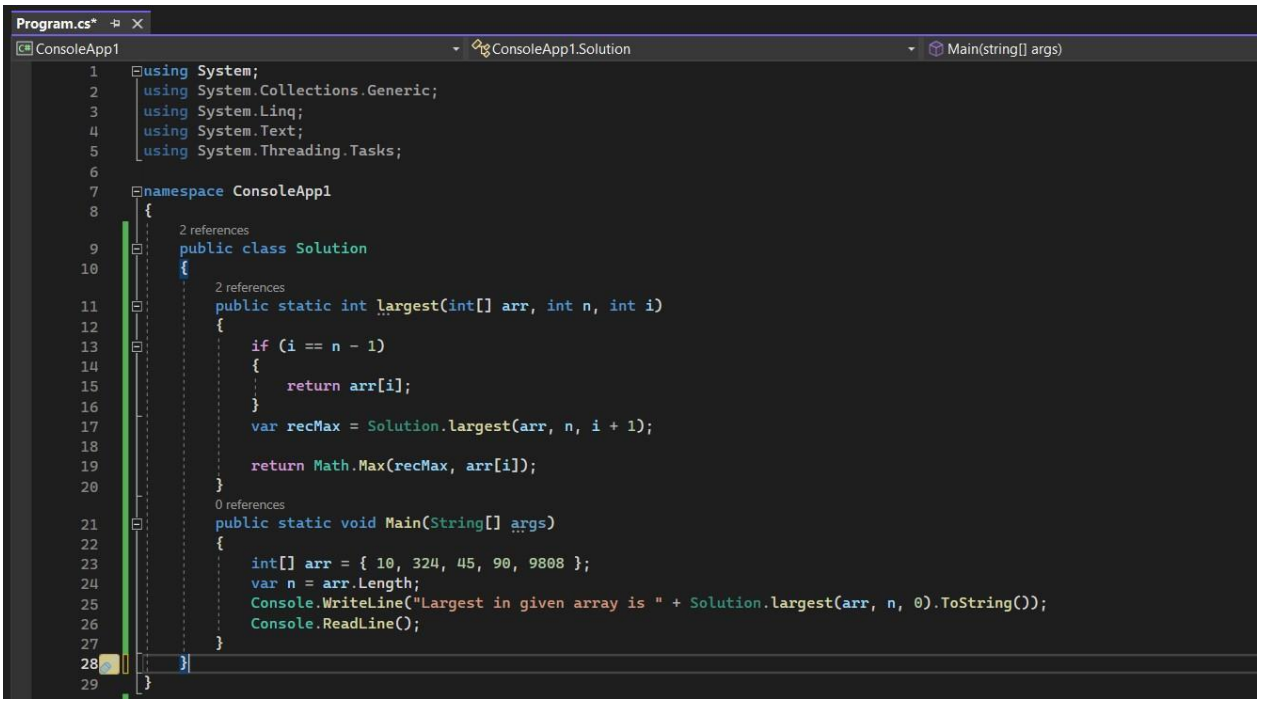
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ConsoleApp1
{
   public class Solution
   {
      public static int largest(int[] arr, int n, int i)
      {
         if (i == n - 1)
         {
            return arr[i];
         }
         var recMax = Solution.largest(arr, n, i + 1);

         return Math.Max(recMax, arr[i]);
      }
      public static void Main(String[] args)
      {
         int[] arr = { 10, 324, 45, 90, 9808 };
         var n = arr.Length;
         Console.WriteLine("Largest in given array is " +Solution.largest(arr, n, 0).ToString());
         Console.ReadLine();
      }
   }
}
```

Time Complexity: O(N), where N is the size of the given array.

Output:

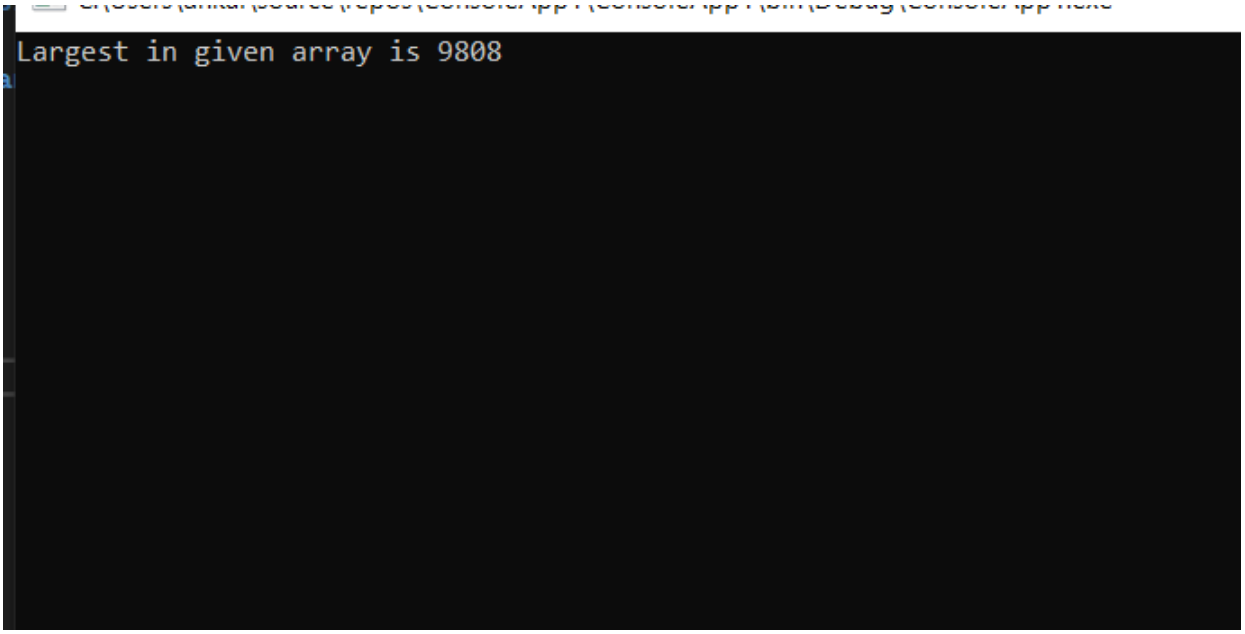


```
Program.cs*
ConsoleApp1 | ConsoleApp1.Solution | Main(string[] args)

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ConsoleApp1
{
    2 references
    public class Solution
    {
        2 references
        public static int largest(int[] arr, int n, int i)
        {
            if (i == n - 1)
            {
                return arr[i];
            }
            var recMax = Solution.largest(arr, n, i + 1);

            return Math.Max(recMax, arr[i]);
        }
        0 references
        public static void Main(String[] args)
        {
            int[] arr = { 10, 324, 45, 90, 9808 };
            var n = arr.Length;
            Console.WriteLine("Largest in given array is " + Solution.largest(arr, n, 0).ToString());
            Console.ReadLine();
        }
    }
}
```



```
Largest in given array is 9808
```