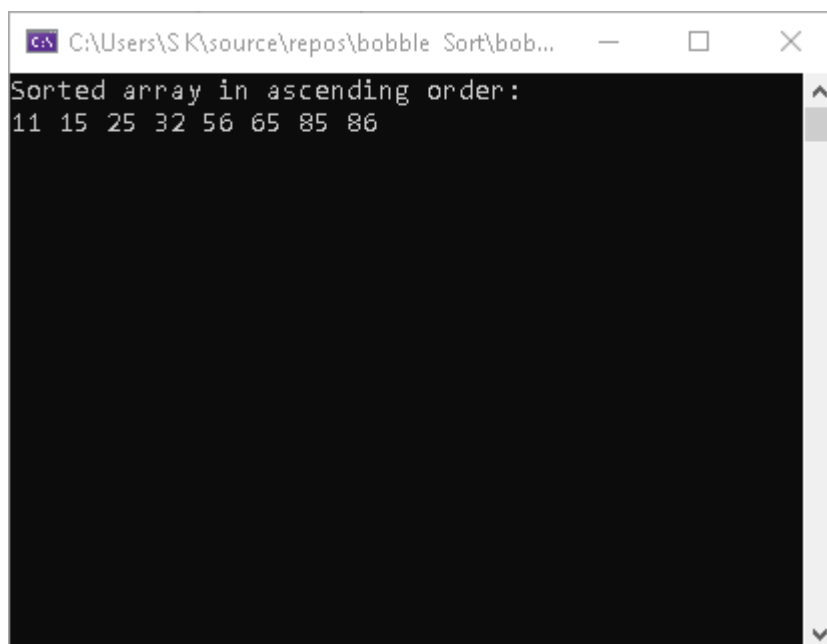


1. By using the bubble sort algorithm, write C# code to sort an integer array of 10 elements in ascending

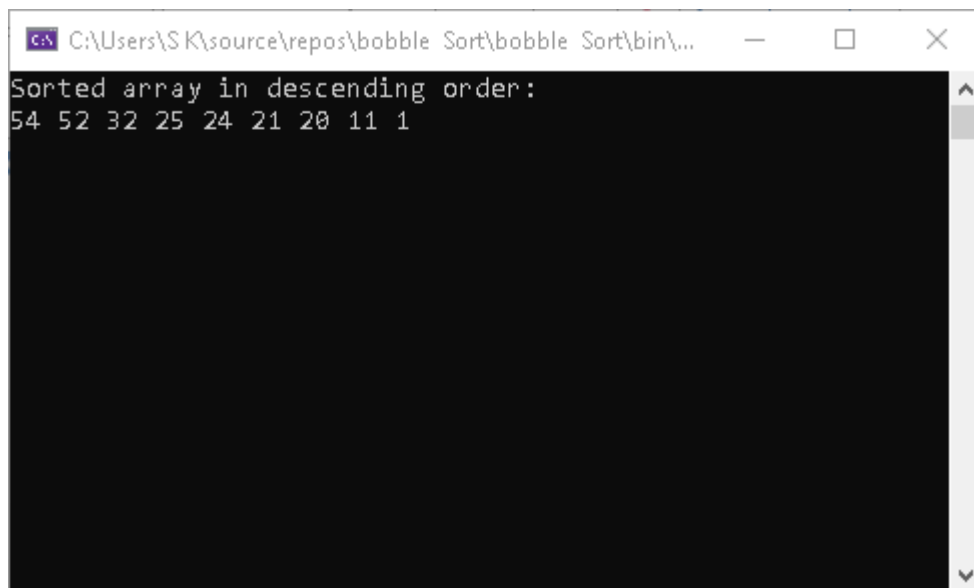
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace SortingArray
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] arr = { 32,25,11,15,56,86,85,65 };
            int temp;
            for (int j = 0; j <= arr.Length - 2; j++)
            {
                for (int i = 0; i <= arr.Length - 2; i++)
                {
                    if (arr[i] > arr[i + 1])
                    {
                        temp = arr[i + 1];
                        arr[i + 1] = arr[i];
                        arr[i] = temp;
                    }
                }
            }
            Console.WriteLine("Sorted array in ascending order:");
            foreach (int p in arr)
                Console.Write(p + " ");
            Console.Read();
        }
    }
}
```



The screenshot shows a Windows console window with the title bar "C:\Users\S K\source\repos\bubble Sort\bob...". The console output displays the sorted array in ascending order: "Sorted array in ascending order:" followed by the numbers "11 15 25 32 56 65 85 86" on the next line. The console has a black background and white text.

2. **Modify the C# code in exercise 1 in order to sort the array in descending order.**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace SortingArray
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] arr = { 20, 52, 25, 24, 1, 11, 21, 32, 54 };
            int temp;
            for (int j = 0; j <= arr.Length - 2; j++)
            {
                for (int i = 0; i <= arr.Length - 2; i++)
                {
                    if (arr[i] < arr[i + 1])
                    {
                        temp = arr[i + 1];
                        arr[i + 1] = arr[i];
                        arr[i] = temp;
                    }
                }
            }
            Console.WriteLine("Sorted array in descending order:");
            foreach (int p in arr)
                Console.Write(p + " ");
            Console.Read();
        }
    }
}
```

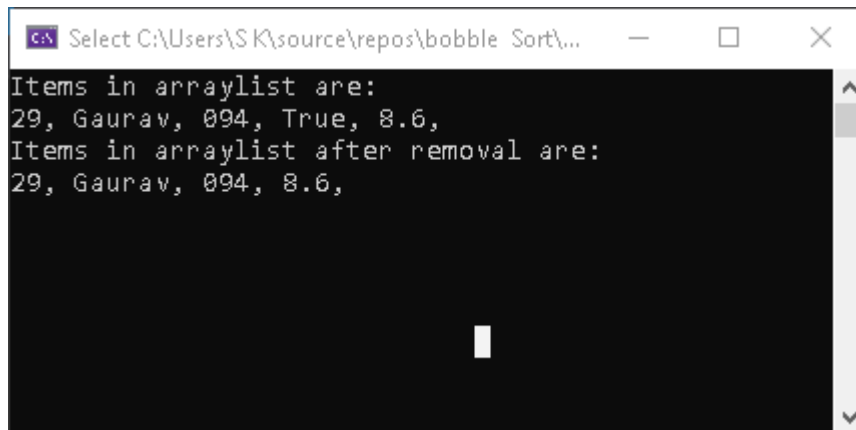


The screenshot shows a Windows command prompt window with the title bar "C:\Users\S\K\source\repos\bobble Sort\bobble Sort\bin\...". The window contains the following text:

```
Sorted array in descending order:
54 52 32 25 24 21 20 11 1
```

### 3. Create an ArrayList, add items, remove items, print all the items.

```
using System;
using System.Collections;
namespace ArrayList
{
    class Program
    {
        static void Main(string[] args)
        {
            var arlist1 = new ArrayList();
            arlist1.Add(29);
            arlist1.Add("Gaurav");
            arlist1.Add("094");
            arlist1.Add(true);
            arlist1.Add(8.6);
            Console.WriteLine("Items in arraylist are: ");
            foreach (var item in arlist1)
                Console.Write(item + ", ");
            arlist1.Remove(true);
            Console.WriteLine();
            Console.WriteLine("Items in arraylist after removal are: ");
            foreach (var item in arlist1)
                Console.Write(item + ", ");
            Console.ReadLine();
        }
    }
}
```



### 4. Create List<>, add items, remove items and print items.

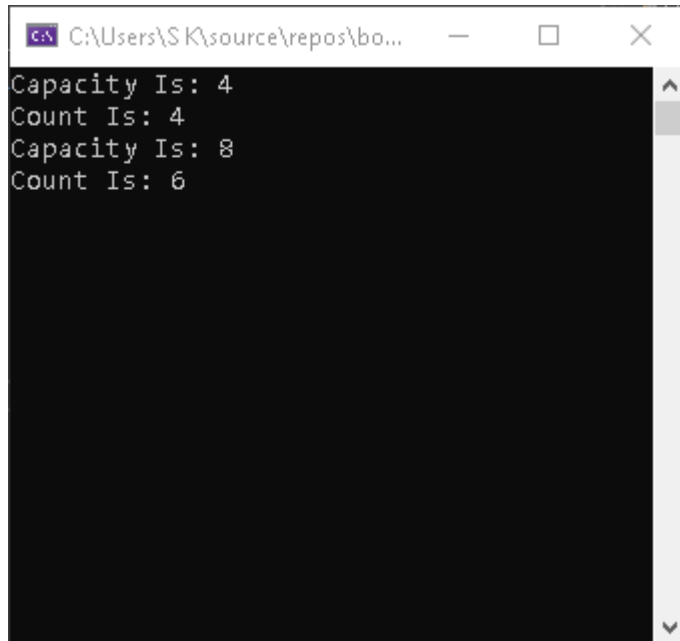
```
using System;
using System.Collections.Generic;
namespace List
{
    class Program
    {
        static void Main(string[] args)
        {
            List<int> firstlist = new List<int>();
            firstlist.Add(3);
            firstlist.Add(22);
            firstlist.Add(54);
            firstlist.Add(92);
        }
    }
}
```

```

        Console.WriteLine("Capacity Is: " + firstlist.Capacity);
        Console.WriteLine("Count Is: " + firstlist.Count);
        firstlist.Add(54);
        firstlist.Add(16);

        Console.WriteLine("Capacity Is: " + firstlist.Capacity);
        Console.WriteLine("Count Is: " + firstlist.Count);
        Console.ReadLine();
    }
}

```



```

C:\Users\S K\source\repos\bo...
Capacity Is: 4
Count Is: 4
Capacity Is: 8
Count Is: 6

```

## 5. Create sorted list and perform basic operations.

```

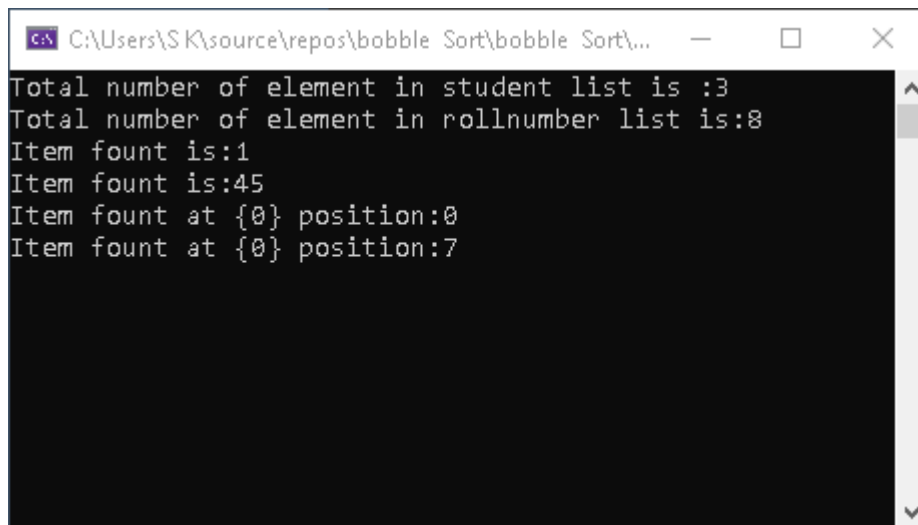
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace SortedList
{
    class Program
    {
        static void Main(string[] args)
        {
            List<string> student = new List<string>();
            List<int> rollnumber = new List<int>();
            student.Add("Aman");
            student.Add("Dev");
            student.Add("Ram");
            rollnumber.Add(1);
            rollnumber.Add(24);
            rollnumber.Add(13);
            rollnumber.Add(42);
            rollnumber.Add(50);
            rollnumber.Add(25);
            rollnumber.Add(30);
            rollnumber.Add(45);
            int a = student.Count();

```

```

        Console.WriteLine("Total number of element in student list is : " + a);
        int b = rollnumber.Count();
        Console.WriteLine("Total number of element in rollnumber list is:" + b);
        int i = rollnumber.Find(item => item < 60);
        Console.WriteLine("Item fount is:" + i);
        int x = rollnumber.FindLast(item => item < 60);
        Console.WriteLine("Item fount is:" + x);
        int y = rollnumber.FindIndex(item => item < 60);
        Console.WriteLine("Item fount at {0} position:" + y);
        int j = rollnumber.FindLastIndex(item => item < 60);
        Console.WriteLine("Item fount at {0} position:" + j);
        Console.ReadLine();
    }
}
}

```



```

C:\Users\SK\source\repos\bobble Sort\bobble Sort\...
Total number of element in student list is :3
Total number of element in rollnumber list is:8
Item fount is:1
Item fount is:45
Item fount at {0} position:0
Item fount at {0} position:7

```

## 6. Create Dictionary and hashtable, add few items, remove items, print items.

```

using System;
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace Dictionary_HashTable
{
    class Program
    {
        static void Main(string[] args)
        {
            Dictionary<int, string> My_dict1 = new Dictionary<int, string>();
            My_dict1.Add(1123, "Welcome");
            My_dict1.Add(1124, "to");
            My_dict1.Add(1125, "My World");
            foreach (KeyValuePair<int, string> ele1 in My_dict1)
            {
                Console.WriteLine("{0} and {1}", ele1.Key, ele1.Value);
            }
            Console.WriteLine();
            My_dict1.Remove(1123);
            foreach (KeyValuePair<int, string> ele in My_dict1)

```

```

        {
            Console.WriteLine("{0} and {1}", ele.Key, ele.Value);
        }
        Console.WriteLine(" ");
        Console.WriteLine("Total number of key/value pairs present in My_dict1:{0}",
My_dict1.Count);
        My_dict1.Clear();
        Console.WriteLine("Total number of key/value " + "pairs present in My_dict1
after clear:{0} ", My_dict1.Count);
        Console.WriteLine();
        Hashtable my_hashtable = new Hashtable();
        my_hashtable.Add("A1", "Welcome");
        my_hashtable.Add("A2", "to");
        my_hashtable.Add("A3", "My World");
        my_hashtable.Remove("A2");
        Console.WriteLine("Key and Value pairs :");
        foreach (DictionaryEntry ele1 in my_hashtable)
        {
            Console.WriteLine("{0} and {1} ", ele1.Key, ele1.Value);
        }
        Console.WriteLine("Total number of elements present" + " in my_hashtable:{0} ",
my_hashtable.Count);
        my_hashtable.Clear();
        Console.WriteLine("Total number of elements present in" + " my_hashtable:{0}",
my_hashtable.Count);
        Console.ReadLine();
    }
}
}

```

```

1124 and to
1125 and C#

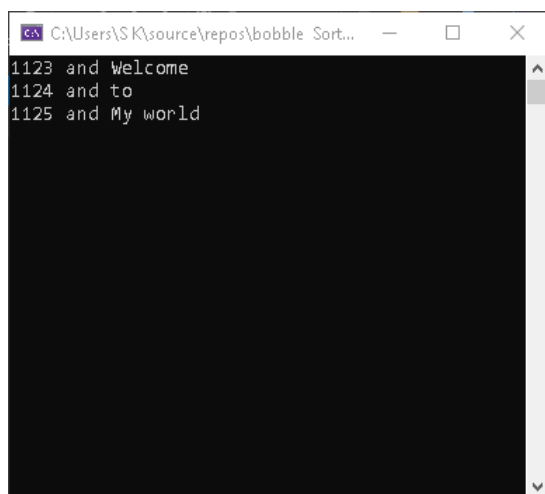
Total number of key/value pairs present in My_dict1:2
Total number of key/value pairs present in My_dict1 after clear:0

Key and Value pairs :
A3 and C#
A1 and Welcome
Total number of elements present in my_hashtable:2
Total number of elements present in my_hashtable:0

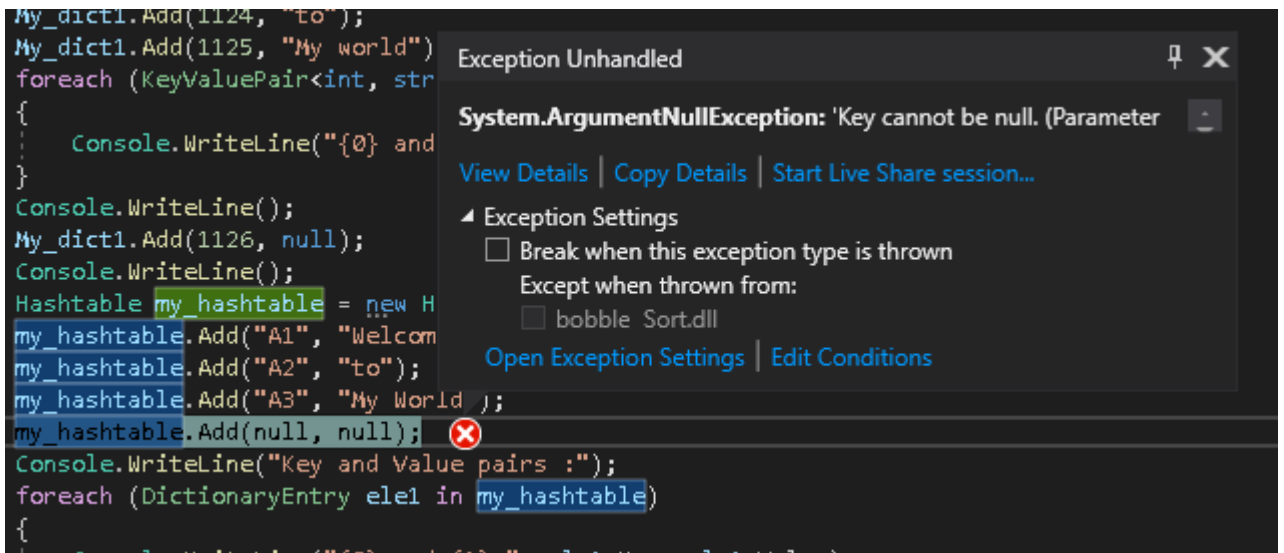
```

## 7. Try to add null values/keys to both of them and note out output/errors

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace Dictionary_HashTable
{
    class Program
    {
        static void Main(string[] args)
        {
            Dictionary<int, string> My_dict1 = new Dictionary<int, string>();
            My_dict1.Add(1123, "Welcome");
            My_dict1.Add(1124, "to");
            My_dict1.Add(1125, "My world");
            foreach (KeyValuePair<int, string> ele1 in My_dict1)
            {
                Console.WriteLine("{0} and {1}", ele1.Key, ele1.Value);
            }
            Console.WriteLine();
            My_dict1.Add(1126, null);
            Console.WriteLine();
            Hashtable my_hashtable = new Hashtable();
            my_hashtable.Add("A1", "Welcome");
            my_hashtable.Add("A2", "to");
            my_hashtable.Add("A3", "My World");
            my_hashtable.Add(null, null);
            Console.WriteLine("Key and Value pairs :");
            foreach (DictionaryEntry ele1 in my_hashtable)
            {
                Console.WriteLine("{0} and {1} ", ele1.Key, ele1.Value);
            }
            Console.ReadLine();
        }
    }
}
```



```
C:\Users\SJK\source\repos\bobble Sort...
1123 and Welcome
1124 and to
1125 and My world
```



8. Write a program in C# Sharp to calculate the sum of elements in an array.

**Test Data :**

Input 5 elements in the array :

element - 0 : 5

element - 1 : 7

element - 2 : 3

element - 3 : 2

element - 4 : 9

*Expected Output :*

The sum of the elements of the array is 26

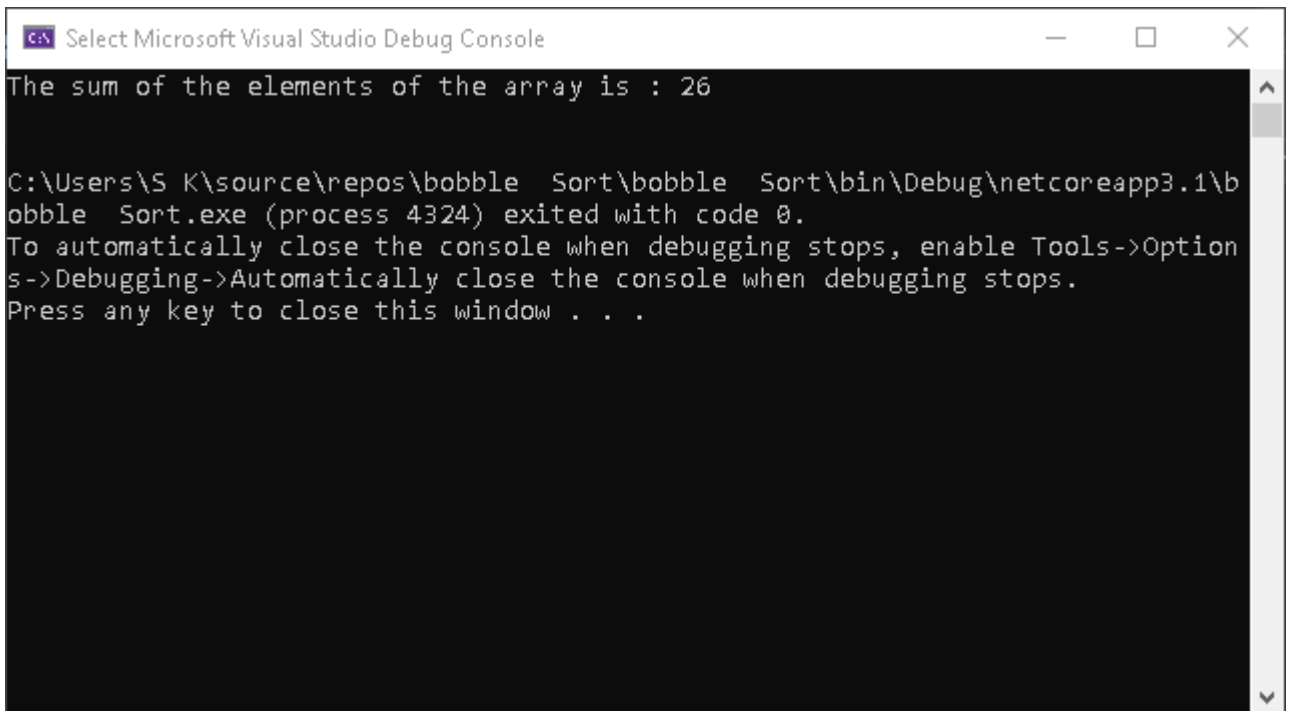
```
using System;
public class Exercise5
{
    public static void Main()
    {
        int[] b = { 5, 7, 3, 2, 9 };
        int n, sum = 0;

        n = b.Length;

        for (int i = 0; i < n; i++)
        {
            sum += b[i];
        }

        Console.WriteLine("The sum of the elements of the array is : {0}\n\n", sum);
    }
}
```





```
Select Microsoft Visual Studio Debug Console

The sum of the elements of the array is : 26

C:\Users\S K\source\repos\bobble Sort\bobble Sort\bin\Debug\netcoreapp3.1\bobble Sort.exe (process 4324) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

9. **Write a program in C# Sharp to create a function to calculate the sum of the individual digits of a given number.**

Test Data :

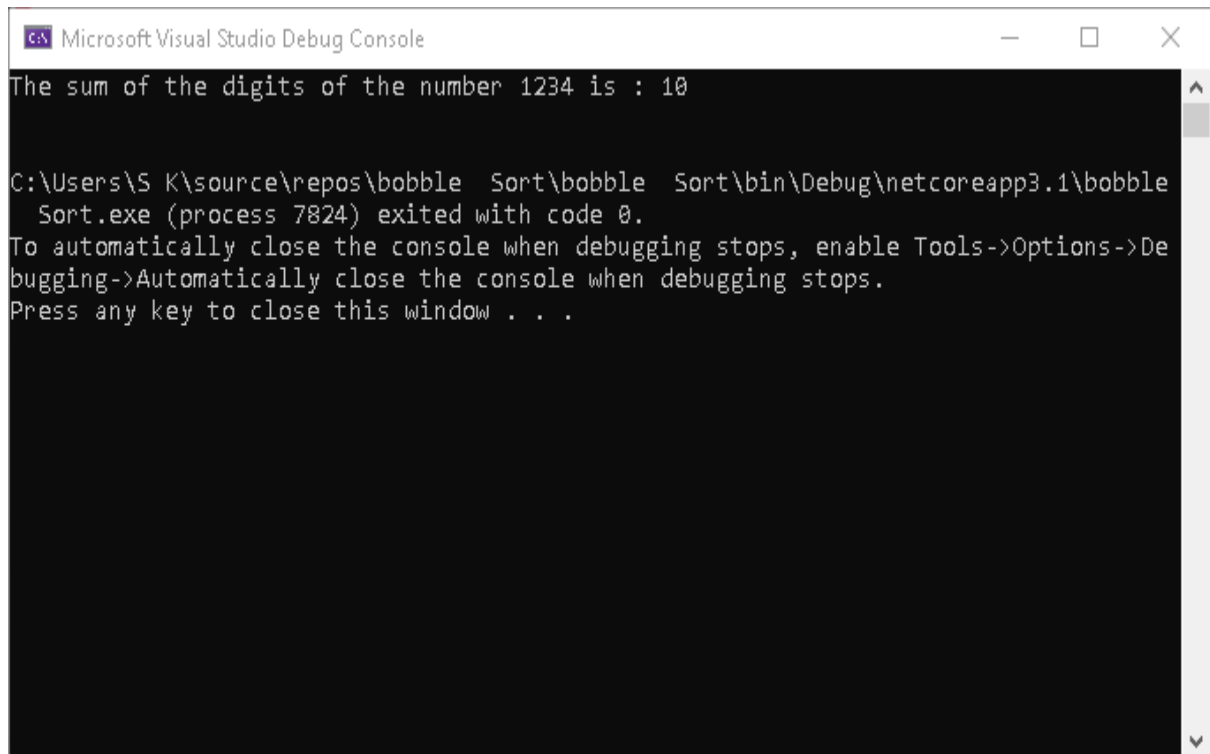
Enter a number: 1234

*Expected Output :*

The sum of the digits of the number 1234 is : 10

```
using System;
public class Excercise6
{
    public static int SumCal(int n)
    {
        string n1 = Convert.ToString(n);
        int sum = 0;
        for (int i = 0; i < n1.Length; i++)
            sum += Convert.ToInt32(n1.Substring(i, 1));
        return sum;
    }

    public static void Main()
    {
        int num;
        num = 1234;
        Console.WriteLine("The sum of the digits of the number {0} is : {1} \n", num, SumCal(num));
    }
}
```



```
Microsoft Visual Studio Debug Console

The sum of the digits of the number 1234 is : 10

C:\Users\S K\source\repos\bobble Sort\bobble Sort\bin\Debug\netcoreapp3.1\bobble Sort.exe (process 7824) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

10. **Write a program in C# Sharp to create a recursive function to find the factorial of a given number.**

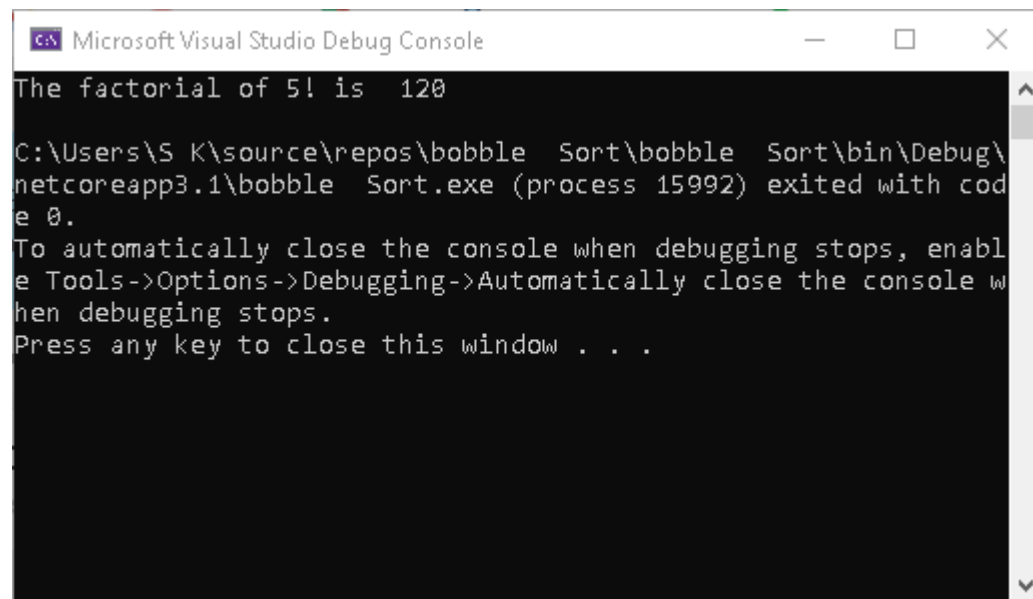
Test Data :

Enter a number: 5

*Expected Output :*

The factorial of 5! is 120

```
using System;
class Excercise7
{
    static void Main()
    {
        decimal f;
        int num = 5;
        f = Factorial(num);
        Console.WriteLine("The factorial of {0}! is {1}", num, f);
    }
    static decimal Factorial(int n1)
    {
        if (n1 == 0)
        {
            return 1;
        }
        else
        {
            return n1 * Factorial(n1 - 1);
        }
    }
}
```



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "Microsoft Visual Studio Debug Console". The console output is as follows:

```
The factorial of 5! is 120  
  
C:\Users\S K\source\repos\bobble Sort\bobble Sort\bin\Debug\  
netcoreapp3.1\bobble Sort.exe (process 15992) exited with cod  
e 0.  
To automatically close the console when debugging stops, enabl  
e Tools->Options->Debugging->Automatically close the console w  
hen debugging stops.  
Press any key to close this window . . .
```