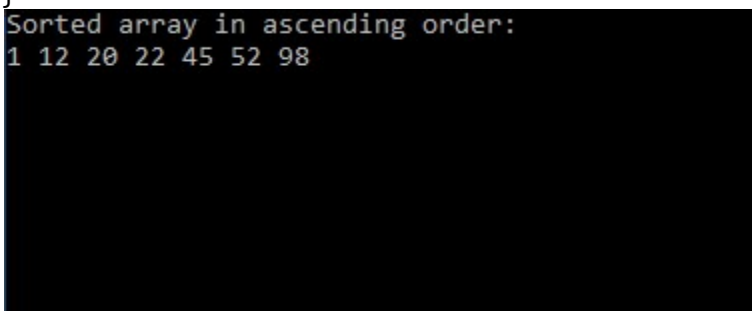
1. **By using the bubble sort algorithm, write C# code to sort an integer array of 10 elements in ascending.**

```
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 = {20, 52, 98, 1, 22, 12, 45};
            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();
        }
    }
}
```

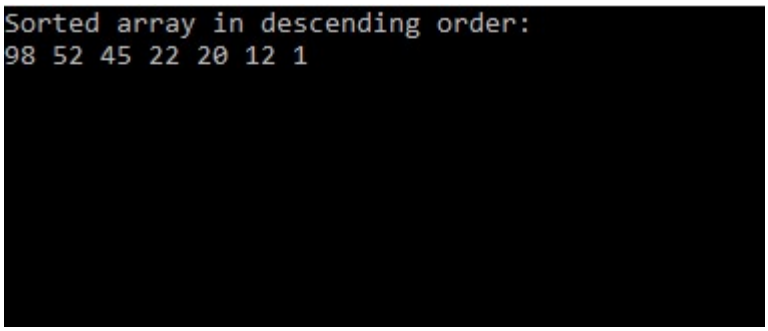


2. **Modify the C# code in exercise 1 in order to sort the array in descending order.**

```
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, 98, 1, 22, 12, 45};
            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();
        }
    }
}
```



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("Nandini");
            arlist1.Add("943");
            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();
        }
    }
}
```

```
Items in arraylist are:
29, Nandini, 943, True, 8.6,
Items in arraylist after removal are:
29, Nandini, 943, 8.6,
```

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(21);
            firstlist.Add(56);
            firstlist.Add(9);

            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();
        }
    }
}
```

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("Ram");
            student.Add("Richa");
            student.Add("Ajay");

            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();
        }
    }
}
```

```
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, "C#");

            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", "C#");

            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();
        }
    }
}
```

```
1123 and Welcome
1124 and to
1125 and C#

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, "C#");


            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", "C#");

            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();
        }
    }
}
```