C# applications for Method Calling techniques

Application 1 :

C# application which demonstrates different ways of method calling techniques

```
/*
 There are three ways in which we can call functions in C# as
 1. Call by value
 2. Call by address
 3. Call by reference
 */

using System;

namespace Function_calling_Techniques
{
   class MarvellousDemo
   {
      public void FunValue(int no)
      {
         Console.WriteLine("Call by Value : {0}\n",no);
         no++;
      }

      public unsafe void FunAddress(int* ptr)
      {
         Console.WriteLine("Call by Address : {0}\n", *ptr);
         (*ptr)++;
      }

      public void FunReference(ref int no)
      {
         Console.WriteLine("Call by Reference : {0}\n", no);
         no++;
      }
   }

   class Marvellous
   {
      static unsafe void Main(string[] args)
      {
         int value1 = 10, value2 = 10, value3 = 10;

         MarvellousDemo dobj = new Demo();

         Console.WriteLine("Value before call {0}\n”,value1);

         dobj.FunValue(value1);

         Console.WriteLine("Value after call {0}\n", value1);
```

```
            Console.WriteLine("Value before call {0}\n", value2);

            dobj.FunAddress(&value2);

            Console.WriteLine("Value after call {0}\n", value2);

            Console.WriteLine("Value before call {0}\n", value3);

            dobj.FunReference(ref value3);

            Console.WriteLine("Value after call {0}\n", value3);
        }
    }
}
```

Application 2 :

C# application which demonstrates the use of out keyword while passing parameters to the function.

```
/*
C# provides out keyword to pass arguments as out-type.
It is like reference-type, except that it does not require variable to initialise before passing.
 */
using System;

namespace Out_Parameter
{
    class MarvellousDemo
    {
        public void fun(out int no)
        {
            no = 11;
        }
    }

    class Marvellous
    {
        static void Main(string[] args)
        {
            int value1, value2 = 10;

            MarvellousDemo dobj = new Demo();

            //Console.WriteLine("Before call {0}", value1);

            dobj.fun(out value1);

            Console.WriteLine("After call {0}", value1);
```

```
            Console.WriteLine("Before call {0}", value2);

            dobj.fun(out value2);

            Console.WriteLine("After call {0}", value2);
        }
    }
}
```

Application 3 :

C# application which demonstrates the way of passing array to the method.

```
// Passing array to the function

using System;

namespace Function_and_Array
{
    class MarvellousDemo
    {
        static void Display(int[] arr)
        {
            Console.WriteLine("Elements of array are :");

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

        static void Main(string[] args)
        {
            int[] arr1 = {10, 20, 30, 40};
            int[] arr2 = { 11, 21, 51};

            Display(arr1);
            Display(arr2);
        }
    }
}
```