C# applications which uses Array and Iterations

Application 1 :

C# application which demonstrates types of array and its memory allocation techniques.

```
/*
 Aray is linear data structure which holds homogeneous elements in indexed format.
 There are three types of array as
 1. Single dimentional array
 2. Multi dimentional array
 3. Jagged array
 */

using System;

namespace MarvellousArray
{
    class Marvellous
    {
        static void Main(string[] args)
        {
            // Below syntax demonstrate creation of single dimensional array

            // First way
            int[] arr1 = new int[5];
            arr1[0] = 11;
            arr1[1] = 21;
            arr1[2] = 51;
            arr1[3] = 101;
            arr1[4] = 151;

            // Second way
            int[] arr2 = new int[5]{10, 20, 30, 40, 50};

            // Third way
            int[] arr3 = {11, 22, 33, 44, 55};

            // In this case arr3 and arr4 referring to same memory locations (Shallow copy)
            int[] arr4 = arr3;

            // Creating multi dimensional array (2D)

            // First way
            int[,] brr1 = new int[3, 4] {{0, 1, 2, 3},{4, 5, 6, 7},{8, 9, 10, 11}};

            // Second way
            int[,] brr2 = new int[2, 2];
            brr2[0, 0] = 10;
            brr2[0, 1] = 20;
            brr2[1, 0] = 30;
```

```
            brr2[1, 0] = 30;

            // Third way
            int[,] brr3 = { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 } };

            //Jagged array - Every row of different size
            int[][] crr = new int[2][];

            crr[0] = new int[3];
            crr[1] = new int[4];
        }
    }
}
```

Application 2 :

C# application which demonstrate the ways in which we can perform iterations.

```
/*
 There are four ways in which we can perform iteration in C# as
 1. for loop
 2. while loop
 3. do-while loop
 4. foreeach loop
 */

// We can demonstate every loop with array

using System;

namespace MarvellousArray
{
    class Marvellous
    {
        static void Main(string[] args)
        {
            // Create one dimensional array
            int[] arr1 = new int[5];
            arr1[0] = 11;
            arr1[1] = 21;
            arr1[2] = 51;
            arr1[3] = 101;
            arr1[4] = 151;

            Console.Write("Contents of array using For loop :");
                for (int i = 0; i < arr1.Length; i++)
                {
                        Console.Write(" " + arr1[i]);
                }
            Console.WriteLine("");
```

```csharp
Console.Write("Contents of array using For-each loop :");
    foreach (int i in arr1)
    {
            Console.Write(" " + i);
    }

Console.WriteLine("");

Console.Write("Contents of array using while loop :");
int j = 0;
while (j < arr1.Length)
{
    Console.Write(" " + arr1[j]);
    j++;
}

Console.WriteLine("");

Console.Write("Contents of array using Do-while loop :");
int k = 0;
do
{
    Console.Write(" " + arr1[k]);
    k++;
} while (k < arr1.Length);

Console.WriteLine();

// Creating multi dimensional array (2D)
int[,] brr1 = new int[3, 4] {{0, 1, 2, 3},{4, 5, 6, 7},{8, 9, 10, 11}};

Console.Write("Contents of multi dimensional array are\n");

    // We can iterate multi dimensional array using nested loops as

for(int i=0;i<3;i++)
{
    for(int x=0;x<4;x++)
    {
        Console.Write(brr1[i,x]+" ");
    }
}
}
}
```