///////////////////////Program1////////////////////

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace CSharp020716\_E1

{

class Program

{

static void Main(string[] args)

{

int[] nums = new int[4];

Console.WriteLine("Length of the num is " + nums.Length);

for (int i=0; i < nums.Length; i++)

nums[i] = i \* i;

Console.Write("Here as nums: ");

for (int i=0; i < nums.Length; i++)

Console.Write(nums[i] + " ");

Console.WriteLine();

Console.ReadLine();

}

}

}

///////////////////////Program2////////////////////

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace CSharp020716\_E2

{

class Program

{

static void Main(string[] args)

{

int[, ,] nums = new int[3, 5, 6];

Console.WriteLine("Length of nums Complet array is : "+nums.Length);

Console.ReadLine();

}

}

}

///////////////////////Program3////////////////////

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace CSharp020716\_E3

{

class Program

{

static void Main(string[] args)

{

int i, j;

int []nums1 = new int[4];

int []nums2 = new int[4];

for (i = 0; i < nums1.Length; i++)

nums1[i] = i;

Console.Write("Original Contents: ");

for (i = 0; i < nums1.Length; i++)

Console.Write(nums1[i] + " ");

Console.WriteLine();

if(nums2.Length >= nums1.Length)

for(i=0,j=nums1.Length-1;i<nums1.Length;i++,j--)

nums2[j]=nums1[i];

Console.Write("Reverse Contents: ");

for (i = 0; i < nums1.Length; i++)

Console.Write(nums2[i] + " ");

Console.WriteLine();

Console.ReadLine();

}

}

}

///////////////////////Program4////////////////////

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace CSharp020716\_E4

{

class Jagged

{

static void Main(string[] args)

{

int[][] network\_nodes = new int[3][];

network\_nodes[0] = new int[3];

network\_nodes[1] = new int[5];

network\_nodes[2] = new int[2];

int i, j;

for (i = 0; i < network\_nodes.Length; i++)

for (j = 0; j < network\_nodes[i].Length; j++)

network\_nodes[i][j] = i \* j + 70;

Console.Write("Total Numbers of network nodes: " + network\_nodes.Length+"\n");

for (i = 0; i < network\_nodes.Length; i++)

{

for (j = 0; j < network\_nodes[i].Length; j++)

{

Console.Write("CPU usage at node " + i + " CPU " + j + ":");

Console.Write(network\_nodes[i][j] + "% \n");

}

Console.WriteLine();

}

Console.WriteLine();

Console.ReadLine();

}

}

}