REPORTE DE PRÁCTICA

| Nombre Completo | Machado Sanchez Javier |
| --- | --- |
| Número de la Unidad a Evaluar | 5 |
| Tema a Evaluar | Ordenamiento Radix Forma Ascendente |
| Fecha de Entrega | 02 de Noviembre del 2023 |

**Programa Fuente**

| using System;  using System.Collections.Generic;  namespace P36\_Ordenamiento\_Radix\_Forma\_Larga  {  internal class Program  {  public static int[] Numeros = new int[30];  public static IList<IList<int>> Digitos = new List<IList<int>>(30);  public static int tamanio\_maximo = 0;  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  //Declaración de método para ordenar  public static void OrdenamientoRadix()  {  for(int i = 0; i < tamanio\_maximo; i++)  {  for(int j = 0; j < Numeros.Length; j++)  {  int digito = (int)(Numeros[j] % Math.Pow(10, i + 1) / Math.Pow(10,i));  Digitos[digito].Add(Numeros[j]);  }  int indice = 0;  for (int y = 0; y < Digitos.Count; y++)  {  IList<int> selDigito = Digitos[y];  for(int I = 0; I < selDigito.Count; I++)  {  Numeros[indice++] = selDigito[I];  }  }  LimpiarDigitos();  }  }  //Fin declaración de método para ordenar  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  //Inicio declaración de método para limpiar dígitos  public static void LimpiarDigitos()  {  for(int i = 0; i < Digitos.Count; i++)  {  Digitos[i].Clear();  }  }  //Fin declaración de método para limpiar dígitos  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  //Inicio declaración método para mostrar  public static void Mostrar(int[] Numeros)  {  for (int i = 1; i <= Numeros.Length; i++)  {  if (i == (Numeros.Length / 2 + 1)) Console.Write("\n");  Console.Write($"[{Numeros[i - 1]}] ");  }  }  //Fin declaración de método mostrar  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  static void Main(string[] args)  {  Console.Title = "P36 Radix Sort Ascendente";  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  //Declaración de variables locales  int Numero; char opcion;  Random NumerosRandom = new Random();    //Ciclo for para inicializar la lista Dígitos  for (int i = 0; i < 30; i++)  {  Digitos.Add(new List<int>());  }  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  do //Do while para controlar menú  {  //Despliegue de menú  Console.Write("\n\t\t.: MENÚ DE OPCIONES :." +  "\n\n\t[ 1 ] - Capturar Números" +  "\n\t[ 2 ] - Ordenar Números Ascendentemente" +  "\n\t[ 3 ] - Salir del Programa" +  "\n\n\tIngrese el número de la opción deseada: ");  if (char.TryParse(Console.ReadLine(), out opcion) != false) //Try catch para controlar errores de formato  {  Console.Write($"\n\n\tLa opción {opcion} se ha capturado correctamente" +  $"\n\n\tPresione la tecla <INTRO> para continuar...");  while (Console.ReadKey().Key != ConsoleKey.Enter) ; Console.Clear();  //Switch case para controlar opciones  switch (opcion)  {  case '1':  {  Console.WriteLine("\n\t\t.: CAPTURA DE DATOS :." +  "\n\n\t| La captura de datos se ha realizado por medio de números random |");  for (int i = 0; i < Numeros.Length; i++) //Ciclo para capturar todos los números  {  Numero = NumerosRandom.Next(0, 1000);  Console.Write($"\n\t{i + 1}.- Ingrese un numero: {Numero}");  Console.WriteLine($"\n\tSe ha capturado el número {Numero} correctamente.");  Numeros[i] = Numero;  if(tamanio\_maximo < Numeros[i].ToString().Length)  {  tamanio\_maximo = Numeros[i].ToString().Length;  }  }  Console.Write("\n\tCaptura de datos terminada, presione la tecla <INTRO> para continuar...");  while (Console.ReadKey().Key != ConsoleKey.Enter) ; Console.Clear();  }  break;  case '2':  {  Console.WriteLine("\n\t\t.: ORDENANDO DATOS ASCENDENTEMENTE :." +  "\n\n\t| Este es el arreglo original sin ordenar |\n\t");  Mostrar(Numeros); //Llamada a método para mostrar  Console.WriteLine("\n\n\t| Este es el arreglo ordenado ascendentemente |\n\t");  OrdenamientoRadix();//llamada a método para ordenar  Mostrar(Numeros);  Console.Write("\n\n\tOrdenamiento terminado, presione la tecla <INTRO> para continuar...");  while (Console.ReadKey().Key != ConsoleKey.Enter) ; Console.Clear();  }  break;  case '3':  {  Console.Write("\n\t\t:. SALIDA :." +  "\n\n\tGracias por utilizar nuestro programa" +  "\n\n\tPresione la tecla <Esc> para salir...");  while (Console.ReadKey().Key != ConsoleKey.Escape) ;  }  break;  default:  {  Console.Write("\n\t\t.: OPCIÓN INVALIDA :." +  "\n\n\tPor favor ingrese una opción existente" +  "\n\n\tPresione la tecla <INTRO> para continuar...");  while (Console.ReadKey().Key != ConsoleKey.Enter) ; Console.Clear();  }  break;  }  }  else  {  Console.Write("\n\n\tLa opción no se ha podido capturar, asegúrese de escribirla correctamente" +  "\n\n\tPresione la tecla <INTRO> para continuar...");  while (Console.ReadKey().Key != ConsoleKey.Enter) ; Console.Clear();  }  } while (opcion != '3');  }  }  } |
| --- |

**Ventana de Entrada de datos**

|  |
| --- |

**Ventana de Resultados**

|  |
| --- |

**EXPLICA BREVEMENTE QUÉ TE PARECIÓ EL PROBLEMA, QUE SE TE DIFICULTO Y COMO LO SOLUCIONASTE.**

| Tuve un problema a la hora de correr el programa. Resulta que me lanzaba una excepción del tipo ArgumentOutOfRange al querer ingresar valores, me decía que el problema era el index, es decir, la posición.  Esto sucedía en la primera vuelta ya que miré que al momento del error el contador de los for valía 0, así que eso me hizo pensar que el problema estaba en la declaración, porque era como si la lista no tuviera ningún tamaño.  Me di cuenta que la lista era de tipo lista, ya que estaba IList<List<int>>, de ahí pensé que quizá y no estaba inicializando List<int>, por lo que el index no existía o era null debido a que no se había agregado nada aún. Le pregunté sobre esto a ChatGPT y me confirmó que era así, también me dió un ciclo for que se encargaba de inicializar las listas internas hasta el número del arreglo a utilizar.  De esta manera el problema se solucionó. |
| --- |