REPORTE DE PRÁCTICA

| Nombre Completo | Machado Sanchez Javier |
| --- | --- |
| Número de la Unidad a Evaluar | 1 |
| Tema a Evaluar | Manejando HashSet |
| Fecha de Entrega | 04 de Septiembre del 2023 |

**Programa Fuente**

| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace P7\_Manejando\_HashSet  {  internal class Program  {  static void Main(string[] args)  {  //Machado Sanchez Javier - 22211600  Console.Title = "Manejando HashSet";  bool error;  //Declaración de variables para los conjuntos  HashSet<int> Conjunto\_A = new HashSet<int>();  HashSet<int> Conjunto\_B = new HashSet<int>();  HashSet<int> Conjunto\_C = new HashSet<int>();  HashSet<int> Conjunto\_D = new HashSet<int>();  //Captura de datos del conjunto A  do  {  try  {  Console.WriteLine("\n\t\t.: CONJUNTO A :." +  "\n\t| Ingrese los promedios del conjunto |");  Capturar(Conjunto\_A, 5);  error = false;  }  catch (FormatException e)  {  Console.Clear();  Console.WriteLine("\n\t\t.: ALGO HA SALIDO MAL :.");  Console.Write("\n\tMensaje de error: {0}" +  "\n\n\tSe reiniciará la captura de datos." +  "\n\tPresione la tecla <INTRO> para continuar...", e.Message);  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  error = true;  }  catch (ArgumentOutOfRangeException e)  {  Console.Clear();  Console.WriteLine("\n\t\t.: ALGO HA SALIDO MAL :.");  Console.Write("\n\tMensaje de error: {0}" +  "\n\n\tSe reiniciará la captura de datos." +  "\n\tPresione la tecla <INTRO> para continuar...", e.Message);  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  error = true;  }  } while (error == true);  Console.Write("\n\n\tLos datos del conjunto han sido capturados correctamente." +  "\n\tPresione la tecla <INTRO> para continuar...");  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  //Captura de datos del conjunto B  do  {  try  {  Console.WriteLine("\n\t\t.: CONJUNTO B :." +  "\n\t| Ingrese los promedios del conjunto |");  Capturar(Conjunto\_B, 3);  error = false;  }  catch (FormatException e)  {  Console.Clear();  Console.WriteLine("\n\t\t.: ALGO HA SALIDO MAL :.");  Console.Write("\n\tMensaje de error: {0}" +  "\n\n\tSe reiniciará la captura de datos." +  "\n\tPresione la tecla <INTRO> para continuar...", e.Message);  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  error = true;  }  catch (ArgumentOutOfRangeException e)  {  Console.Clear();  Console.WriteLine("\n\t\t.: ALGO HA SALIDO MAL :.");  Console.Write("\n\tMensaje de error: {0}" +  "\n\n\tSe reiniciará la captura de datos." +  "\n\tPresione la tecla <INTRO> para continuar...", e.Message);  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  error = true;  }  } while (error == true);  Console.Write("\n\n\tLos datos del conjunto han sido capturados correctamente." +  "\n\tPresione la tecla <INTRO> para continuar...");  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  //Captura de datos del conjunto C  do  {  try  {  Console.WriteLine("\n\t\t.: CONJUNTO C :." +  "\n\t| Ingrese los promedios del conjunto |");  Capturar(Conjunto\_C, 6);  error = false;  }  catch (FormatException e)  {  Console.Clear();  Console.WriteLine("\n\t\t.: ALGO HA SALIDO MAL :.");  Console.Write("\n\tMensaje de error: {0}" +  "\n\n\tSe reiniciará la captura de datos." +  "\n\tPresione la tecla <INTRO> para continuar...", e.Message);  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  error = true;  }  catch (ArgumentOutOfRangeException e)  {  Console.Clear();  Console.WriteLine("\n\t\t.: ALGO HA SALIDO MAL :.");  Console.Write("\n\tMensaje de error: {0}" +  "\n\n\tSe reiniciará la captura de datos." +  "\n\tPresione la tecla <INTRO> para continuar...", e.Message);  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  error = true;  }  } while (error == true);  Console.Write("\n\n\tLos datos del conjunto han sido capturados correctamente." +  "\n\tPresione la tecla <INTRO> para continuar...");  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  //Captura de datos del conjunto D  do  {  try  {  Console.WriteLine("\n\t\t.: CONJUNTO D :." +  "\n\t| Ingrese los promedios del conjunto |");  Capturar(Conjunto\_D, 7);  error = false;  }  catch (FormatException e)  {  Console.Clear();  Console.WriteLine("\n\t\t.: ALGO HA SALIDO MAL :.");  Console.Write("\n\tMensaje de error: {0}" +  "\n\n\tSe reiniciará la captura de datos." +  "\n\tPresione la tecla <INTRO> para continuar...", e.Message);  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  error = true;  }  catch (ArgumentOutOfRangeException e)  {  Console.Clear();  Console.WriteLine("\n\t\t.: ALGO HA SALIDO MAL :.");  Console.Write("\n\tMensaje de error: {0}" +  "\n\n\tSe reiniciará la captura de datos." +  "\n\tPresione la tecla <INTRO> para continuar...", e.Message);  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  error = true;  }  } while (error == true);  Console.Write("\n\n\tLos datos del conjunto han sido capturados correctamente." +  "\n\tPresione la tecla <INTRO> para continuar...");  while (Console.ReadKey().Key != ConsoleKey.Enter) ;  Console.Clear();  //Declaración de conjuntos para resultados y cálculos  HashSet<int> Union = new HashSet<int>(Conjunto\_A);  Union.UnionWith(Conjunto\_C);  HashSet<int> Interseccion = new HashSet<int>(Conjunto\_B);  Interseccion.IntersectWith(Conjunto\_D);  HashSet<int> Diferencia = new HashSet<int>(Conjunto\_C);  Diferencia.ExceptWith(Conjunto\_B);  HashSet<int> Diferencia\_Simetrica = new HashSet<int>(Conjunto\_D);  Diferencia\_Simetrica.SymmetricExceptWith(Conjunto\_A);  //Impresión de resultados  Console.WriteLine("\n\t\t.: IMPRESIÓN DE RESULTADOS :.");  Console.Write("\n Conjunto A:");  Mostrar(Conjunto\_A);  Console.Write("\n Conjunto B:");  Mostrar(Conjunto\_B);  Console.Write("\n Conjunto C:");  Mostrar(Conjunto\_C);  Console.Write("\n Conjunto D:");  Mostrar(Conjunto\_D);  Console.Write("\n Unión A con C:");  Mostrar(Union);  Console.Write("\n Intersección B con D:");  Mostrar(Interseccion);  Console.Write("\n Diferencia C con B:");  Mostrar(Diferencia);  Console.Write("\n Diferencia Simétrica D con A:");  Mostrar(Diferencia\_Simetrica);  Console.Write("\n\n\tCálculos mostrados correctamente." +  "\n\tPresione la tecla <Esc> para salir...");  while (Console.ReadKey().Key != ConsoleKey.Escape) ;  }  //Función estática para mostrar los conjuntos con un formato estándar  static void Mostrar(HashSet<int> conjunto)  {  Console.Write(" {");  foreach (int i in conjunto)  {  Console.Write(" {0},", i);  }  Console.Write("\b }");  }  //Función estática para capturar los elementos de los conjuntos  static void Capturar(HashSet<int> conjunto, int j)  {  int aux = 0;  for (int i = 0; i < j; i++)  {  Console.Write("\n\tPromedio {0}: ", i + 1);  aux = int.Parse(Console.ReadLine());  if (aux >= 0 && aux <= 100)  {  conjunto.Add(aux);  }  else { throw new ArgumentOutOfRangeException(); }  }  }  }  } |
| --- |

**Ventana de Entrada de datos**

|  |
| --- |

**Ventana de Resultados**

|  |
| --- |

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

| En esta actividad se me hizo mucho más sencillo desarrollar el programa debido a que es muy parecido al anterior. La diferencia es que acá decidí añadir una función para hacer la captura de datos y así hacer un código más limpio y óptimo. También aproveche para crear un try catch en cada captura de datos para controlar los errores en caso de que el usuario ponga un dato incorrecto(como decimales o palabras) o se pase del rango de calificaciones(números negativos o mayores a 100).  Sin embargo, como la clase HashSet es nueva para mí, realizaré programas sencillos a lápiz y papel para así desarrollar más la lógica de la clase HashSet y sus métodos; preparándome por sí las utilizamos en algún examen. |
| --- |