using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleStruct

{

struct Marsh : IComparable<Marsh>

{

string StartPointName;

string FinalPointName;

string Number;

public Marsh(string startPointName, string finalPointName, string number)

{

StartPointName = startPointName;

FinalPointName = finalPointName;

Number = number;

}

public override string ToString()

{

var line = String.Format(

"Название начального пункта: {0}\n" +

"Номер конечного пункта: {1}\n" +

"Номер маршрута: {2}\n",

StartPointName, FinalPointName, Number);

return line;

}

public static void Input(List<Marsh> listMarsh)

{

Console.Write("Введите название начального пункта: ");

var startPointName = Console.ReadLine();

Console.Write("Введите название конечного пункта: ");

var finalPointName = Console.ReadLine();

Console.Write("Введите номер маршрута: ");

var number = Console.ReadLine();

var newItem = new Marsh(startPointName, finalPointName, number);

listMarsh.Add(newItem);

}

public static void Output(List<Marsh> listMarsh)

{

foreach (var item in listMarsh) Console.WriteLine(item.ToString());

}

public static void ReverseBubbleSortByStartPointName(List<Marsh> listMarsh)

{

Marsh temp;

for (int i = 0; i < listMarsh.Count; i++)

{

for (int j = i + 1; j < listMarsh.Count; j++)

{

if (listMarsh[i].CompareTo(listMarsh[j]) < 0)

{

temp = listMarsh[i];

listMarsh[i] = listMarsh[j];

listMarsh[j] = temp;

}

}

}

}

public int CompareTo(Marsh other)

{

return this.StartPointName.CompareTo(other.StartPointName);

}

}

}