using System;

using System.Collections.Generic;

namespace ConsoleVector

{

 public enum Direction { Left, Right };//Направление сдвига

 class Vector

 {

 public static double Max(double[] massif)//макс. значение элементов массива

 {

 double max = massif[0];

 foreach(var item in massif)

 if (max < item) max = item;

 return max;

 }

 public static void Reorder(double[] massif)//Перестановка значений в обратном порядке

 {

 Array.Reverse(massif);

 }

 public static void ShiftLeft(double[] massif)//влево на одну позицию

 {

 var n = massif.Length;

 var firstItem = massif[0];

 for (int i = 1; i < n; i++) massif[i - 1] = massif[i];

 massif[n - 1] = firstItem;

 }

 public static void ShiftLeft\_K(int k, double[] massif)//влево на k позиций

 {

 var n = massif.Length;

 double item = 0;

 for (int i = 0; i < k; i++)

 {

 for (int j = 0; j < n; j++)

 if (j == 0) item = massif[j];

 else massif[j - 1] = massif[j];

 massif[n - 1] = item;

 }

 }

 public static void ShiftRight(double[] massif)//вправо на одну позицию

 {

 var n = massif.Length;

 var lastItem = massif[n - 1];

 for (int i = n - 2; i >= 0; i--) massif[i + 1] = massif[i];

 massif[0] = lastItem;

 }

 public static void ShiftRight\_K(int k, double[] massif)//вправо на k позиций

 {

 var n = massif.Length;

 double item = 0;

 for (int i = 0; i < k; i++)

 {

 for (int j = n - 1; j >= 0; j--)

 if (j == n - 1) item = massif[j];

 else massif[j + 1] = massif[j];

 massif[0] = item;

 }

 }

 public static void Shift\_K(Direction d, int k, double[] massif)//в заданном направлении на заданное число позиций

 {

 switch (d)

 {

 case Direction.Left:

 Vector.ShiftLeft\_K(k, massif);

 break;

 case Direction.Right:

 Vector.ShiftRight\_K(k, massif);

 break;

 }

 }

 }

}