

2.2 Модуль CoctailSort

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Gubskiy_CoxtailSort
{
    public class CoctailSort
    {
        public static void Swap(ref int e1, ref int e2)
        {
            var temp = e1;
            e1 = e2;
            e2 = temp;
        }

        public static int[] SortArray(int[] array)
        {
            for (var i = 0; i < array.Length / 2; i++)
            {
                var swapFlag = false;

                for (var j = i; j < array.Length - i - 1; j++)
                {
                    if (array[j] > array[j + 1])
                    {
                        Swap(ref array[j], ref array[j + 1]);
                        swapFlag = true;
                    }
                }

                for (var j = array.Length - 2 - i; j > i; j--)
                {
                    if (array[j - 1] > array[j])
                    {
                        Swap(ref array[j - 1], ref array[j]);
                        swapFlag = true;
                    }
                }

                if (!swapFlag)
                {
                    break;
                }
            }

            return array;
        }
    }
}
```

Рисунок 9 – Модуль CoctailSort

На данном рисунке представлена реализация метода коктейльной сортировки, а именно, весь необходимый для реализации внутри программы код.

Коктейльная сортировка - это улучшение и вариация барботажной сортировки, также известная как «двусторонняя барботажная сортировка». Коктейльная сортировка - это сортировка от низкого до высокого, а затем от высокого к низкому (выберите самые большие и самые маленькие элементы), чем Эффективность сортировки пузырьков несколько лучше, потому что сортировка пузырьков сравнивается только в одном направлении (от низкого до высокого), и только один элемент перемещается за цикл.

2 Алгоритм анализа :

1. Сначала отсортируйте массив слева направо в пузырьковом порядке (в порядке возрастания), затем самый большой элемент перейдет в крайнее правое положение.
2. Затем отсортируйте массив справа налево в пузырьковом порядке (в порядке убывания), затем наименьший элемент перейдет в крайнее левое положение.
3. Цикл 1, 2 шага, последовательное изменение направления пузырьков и продолжение сужения диапазона несортированных элементов до конца последнего элемента [1]