ГУ ВПО «Белорусско-Российский университет»

Кафедра «Автоматизированные системы управления»

Лабораторная работа № 2

**Сравнение эффективности разных видов сортировки в статических структурах данных**

по дисциплине «Типы и структуры данных»

Вариант а, д

Выполнил студент гр. ПМР-211

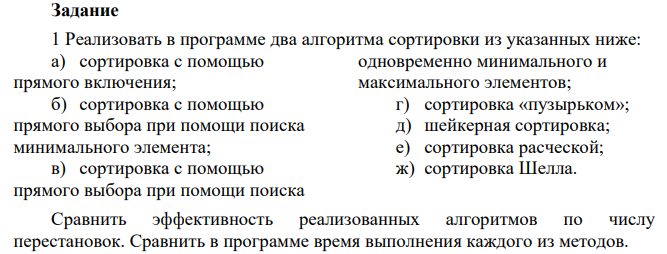
Азарёнок А.Р.а ааааааааааааааа

Проверила ст.преподаватель ааа

Беккер И. А. аааааааааааааааааа.

Могилев 2023

Цель работы: сформировать знания и умения организации сортировки с выполнением анализа алгоритма с учетом рассмотрения лучшего, среднего, худшего случаев.



Листинг программы:

using System;

using System.Diagnostics;

namespace lab2

{

internal class Program

{

static void Main(string[] args)

{

Console.Write("Введите размер массива: ");

int len = int.Parse(Console.ReadLine());

Console.Write("Введите метод создания массива (random, manual, example): ");

string method = Console.ReadLine().ToLower();

int[] array\_1 = get\_arr(method, len);

int[] array\_2 = new int[array\_1.Length];

array\_1.CopyTo(array\_2, 0);

Stopwatch stopwatch = new Stopwatch();

show\_arr("начальный", array\_1);

Console.WriteLine();

//замер времени выполнения метода прямого включения

stopwatch.Start();

InsertionSort(array\_1);

show\_arr("прямое включение", array\_1);

stopwatch.Stop();

Console.WriteLine($"Затраченное время в милисекундах: {stopwatch.Elapsed.TotalMilliseconds}\n");

//замер времени выполнения метода шейкера

stopwatch.Restart();

ShakerSort(array\_2);

show\_arr("шейкер", array\_2);

stopwatch.Stop();

Console.WriteLine($"Затраченное время в милисекундах: {stopwatch.Elapsed.TotalMilliseconds}\n");

}

//сортировка с помощью прямого включения

static void InsertionSort(int[] arr)

{

int count = 0;

for (int i = 1; i < arr.Length; i++)

{

//находим j-й индекс в отсортрованной части массива

int temp = arr[i];

int arr\_index = -1;

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

{

if (temp <= arr[j])

{

arr\_index = j;

break;

}

}

//смещение части массива [j, i-1] на один вправо

int arr\_first = arr[arr\_index];

int temp\_;

for (int j = arr\_index; j < i; j++)

{

temp\_ = arr[j + 1];

arr[j + 1] = arr\_first;

arr\_first = temp\_;

count++;

}

//вставка необходимого числа в массив

arr[arr\_index] = temp;

}

Console.WriteLine($"Количество перестановок: {count}");

}

//шейкерная сортировка

static void ShakerSort(int[] arr)

{

int e\_left = 0;

int e\_right = arr.Length;

int count = 0;

for (int i = 0; i < arr.Length/2; i++)

{

int temp;

if (i % 2 == 0)

{

for (int j = e\_left; j < e\_right - 1; j++)

{

if (arr[j] > arr[j + 1])

{

temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

count++;

}

}

e\_right--;

}

else

{

for (int j = e\_right - 1; j > e\_left; j--)

{

if (arr[j] < arr[j - 1])

{

temp = arr[j];

arr[j] = arr[j - 1];

arr[j - 1] = temp;

count++;

}

}

e\_left++;

}

}

Console.WriteLine($"Количество перестановок: {count}");

}

//вывод маасива на проверку

static void show\_arr(string name, int[] arr)

{

Console.Write($"Массив \"{name}\" : ");

foreach (int i in arr)

{

Console.Write($"{i} ");

}

Console.WriteLine();

}

static int[] get\_arr(string method, int len)

{

int[] arr = new int[len];

switch (method)

{

case "random":

Random rnd = new Random();

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

{

arr[i] = rnd.Next(50);

}

break;

case "manual":

Console.WriteLine($"Введите элементы массива ({len} элемента/ов):");

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

{

Console.Write($"{i + 1}: ");

arr[i] = Convert.ToInt16(Console.ReadLine());

}

break;

case "example":

default:

arr = new int[] { 1, 2, 3, 4, 5, 9, 8, 7, 6 };

break;

}

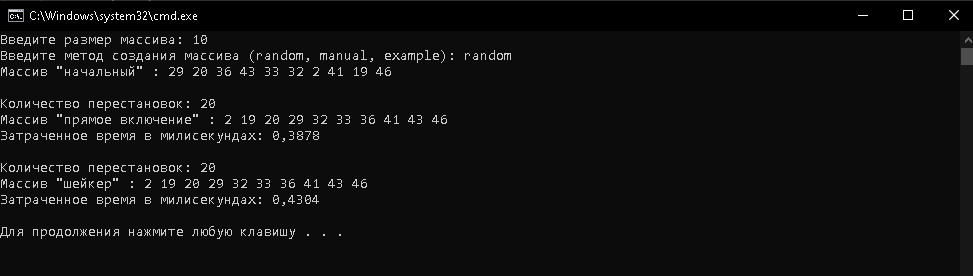
return arr;

}

}

}

Вывод программы:



Вывод: Исходя из проведенного сравнения, метод сортировки прямым включением оказался быстрее шейкерной сортировки.