МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Ижевский государственный технический университет имени М. Т. Калашникова»

Факультет «Информатика и вычислительная техника»

Кафедра «Программное обеспечение»

Алгоритмы и структуры данных

Лабораторная работа №1

«Алгоритмы сортировки»

Вариант № 4

Выполнил:

Студент группы Б03-191-2з

Закиров Айдар Ринатович

Проверил:

Старший преподаватель

[Еланцев Михаил Олегович](https://istu.ru/staff/1092)

Ижевск 2020

**Задача:**

Реализовать алгоритмы сортировки вставками и слиянием на любом языке программирования целочисленного массива.

В качестве ответа приложите файл с исходным текстом программы, контрольный пример и его текстовое описание.

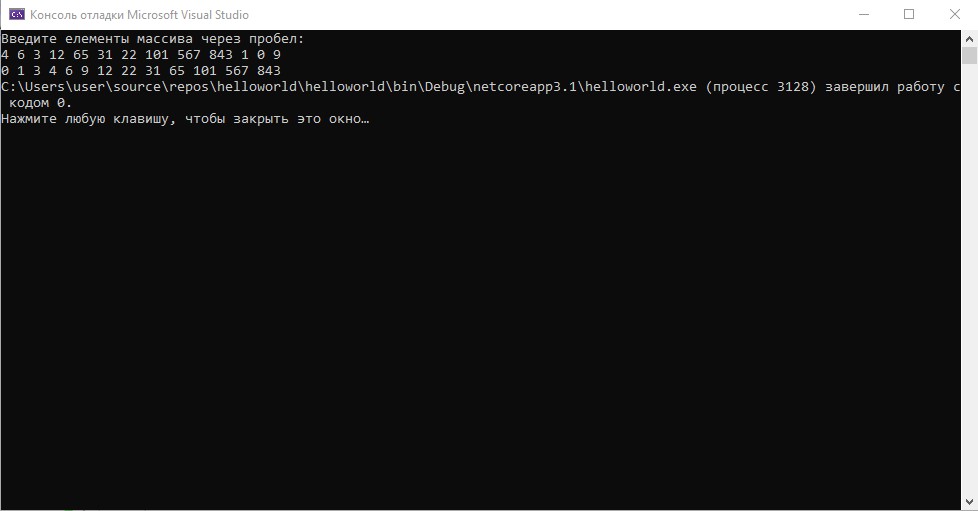
1. **Сортировка вставками.**

**Решение:**

Для сортировки массива методом вставок была составлена программа на языке программирования C#, использующая два метода:

1. InsertionSort - метод принимающий параметром массив целочисленный элементов введённых пользователем и возвращает отсортированный по возрастанию элементов массив. Данный массив использует алгоритм сортировки вставками.
2. Main - метод запускающий программу. Так же в данном методе реализованы ввод и привидение к нужному нам типу данных, введённых пользователем элементов массива, которые в последующем передаются параметром в метод InsertionSort;

**Вывод программы:**



**Программа:**

using System;

using System.Security.Cryptography.X509Certificates;

namespace InsertionSort

{

class Program

{

public int [] InsertionSort(int [] arr)

{

int[] result = new int[arr.Length];

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

{

int j = i;

while (j > 0 && result[j - 1] > arr[i])

{

result[j] = result[j - 1];

j--;

}

result[j] = arr[i];

}

return result;

}

static void Main()

{

Console.WriteLine("Введите елементы массива через пробел: ");

string stringElements = Console.ReadLine();

string[] stringArr = stringElements.Split(" ");

int[] arr = new int[stringArr.Length];

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

{

arr[i] = Convert.ToInt32(stringArr[i]);

}

Program a = new Program();

int[] sortedArr = a.InsertionSort(arr);

for(int i = 0; i<sortedArr.Length; i++)

{

Console.Write($"{sortedArr[i]} ");

}

}

}

}

1. **Сортировка слиянием.**

**Решение:**

Для сортировки массива методом слияния была составлена программа на языке программирования C#, использующая пять методов:

1. TransformationArrays - метод принимающий параметром массив строк введённых пользователем и возвращающий приведённый к нужному нам типу данных массив данных введённых пользователем.

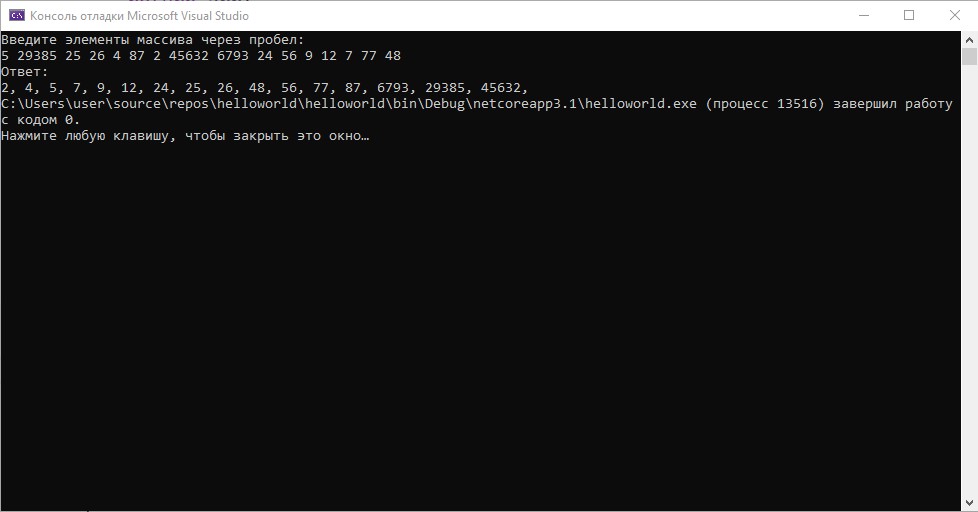
2 ) Sort - метод принимающий параметром массив целочисленных элементов полученных после применения метода TransformationArrays. Данный метод производит деление массива на подмассивы для простоты их дальнейшей сортировки с помощью метода MergeSort который вызывается в методе Sort. По окончании метод возвращает отсортированный по возрастанию элементов целочисленный массив.

3) MergeSort - метод принимающий параметрами два подмассива образованных в методе Sort и возвращает уже отсортированный с помощью алгоритма слияния массив.

4) PrintArray - метод принимающий параметром целочисленный массив, который получен нами из метода Sort, и производит вывод отсортированного массива в консоль.

5) Main - метод запускающий программу, а так же принимающий в виде строки элементы введённые пользователем и преобразует их в массив типа данных string для передачи параметром в метод TransformationArrays.

**Вывод программы:**



**Программа:**

using System;

using System.Security.Cryptography.X509Certificates;

namespace MergeSort {

class Program {

//Преобразование массива string в массив int

public static int[] TransformationArrays(string [] arr) {

int[] intigerArr = new int[arr.Length];

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

intigerArr[i] = Convert.ToInt32(arr[i]);

}

return intigerArr;

}

public static int[] Sort(int[] arr) {

if (arr.Length > 1) {

int[] left = new int[arr.Length / 2];

int[] right = new int[arr.Length - left.Length];

for (int i = 0; i < left.Length; i++) {

left[i] = arr[i];

}

for (int i = 0; i < right.Length; i++) {

right[i] = arr[left.Length + i];

}

if (left.Length > 1)

left = Sort(left);

if (right.Length > 1)

right = Sort(right);

arr = MergeSort(left, right);

}

return arr;

}

public static int[] MergeSort(int[] left, int[] right) {

int[] arr = new int[left.Length + right.Length];

int i = 0;

int l = 0;

int r = 0;

for (; i < arr.Length; i++) {

if (r >= right.Length) {

arr[i] = left[l];

l++;

}

else if (l < left.Length && left[l] < right[r]) {

arr[i] = left[l];

l++;

}

else {

arr[i] = right[r];

r++;

}

}

return arr;

}

public static void PrintArray(int[] arr) {

Console.WriteLine("Ответ:");

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

Console.Write($"{arr[i]}, ");

}

}

static void Main() {

Console.WriteLine("Введите элементы массива через пробел: ");

string elementsArr = Console.ReadLine();

string[] strArr = elementsArr.Split(' ');

int[] array = TransformationArrays(strArr);

PrintArray(Sort(array));

}

}

}

**Вывод:**

В данной лабораторной работе были изучены и применены алгоритмы сортировки методом вставок и методом слияния.