МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

|  |  |
| --- | --- |
|  | Утвержден на заседании кафедры  «Вычислительная техника»  "\_\_\_" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20 г.  Заведующий кафедрой                                              М.А. Митрохин |
|  |  |

**ОТЧЕТ ПО УЧЕБНОЙ (ОЗНАКОМИТЕЛЬНОЙ) ПРАКТИКЕ**

(2023/2024 учебный год)

                                          Молев Павел Олегович

Направление подготовки 09.03.01 «Информатика и вычислительная техника»

Наименование профиля подготовки «Прикладной искусственный интеллект»

Форма обучения – очная Срок обучения в соответствии с ФГОС – 4 года

Год обучения1семестр 2

Период прохождения практики с 25.06.24 по 08.07.24

Кафедра «Вычислительная техника»

Заведующий кафедрой д.т.н., профессор, Митрохин М.А.

*(должность, ученая степень, ученое звание, Ф.И.О.)*

Руководитель практики к.т.н., доцент, Карамышева Н.С.

*(должность, ученая степень, ученое звание)*

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

|  |  |
| --- | --- |
|  | Утвержден на заседании кафедры  «Вычислительная техника»  "\_\_\_" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20 г.  Заведующий кафедрой                                              М.А. Митрохин |
|  |  |

**ИНДИВИДУАЛЬНЫЙ ПЛАН ПРОХОЖДЕНИЯ УЧЕБНОЙ (ОЗНАКОМИТЕЛЬНОЙ) ПРАКТИКИ**

(2023/2024 учебный год)

                                          Молев Павел Олегович

Направление подготовки 09.03.01 «Информатика и вычислительная техника»

Наименование профиля подготовки «Прикладной искусственный интеллект»

Форма обучения – очная Срок обучения в соответствии с ФГОС – 4 года

Год обучения                  1                 семестр                 2

Период прохождения практики с 25.06.24 по 08.07.24

Кафедра «Вычислительная техника»

Заведующий кафедрой д.т.н., профессор, Митрохин М.А.

*(должность, ученая степень, ученое звание, Ф.И.О.)*

Руководитель практики к.т.н., доцент, Карамышева Н.С.

*(должность, ученая степень, ученое звание)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Планируемая форма работы во время практики | Количество часов | Календарные сроки проведения работы | Подпись  руководителя  практики от вуза |
| 1 | Выбор темы и разработка индивидуального плана проведения работ | 2 | 25.06.2024 -  25.06.2024 |  |
| 2 | Подбор и изучение материала по теме работы | 15 | 26.06.2024 –  28.06.24 |  |
| 3 | Разработка алгоритма | 43 | 01.07.24 –  03.07.24 |  |
| 4 | Описание алгоритма и программы | 18 | 03.07.24 –  04.07.24 |  |
| 5 | Тестирование | 5 | 04.07.24 –  05.07.24 |  |
| 6 | Получение и анализ результатов | 10 | 05.07.24 –  08.07.24 |  |
| 7 | Оформление отчёта | 15 | 05.07.24 –  08.07.2024 |  |
|  | **Общий объём часов** | 108 |  |  |

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

**ОТЧЁТ**

**О ПРОХОЖДЕНИИ УЧЕБНОЙ (ОЗНАКОМИТЕЛЬНОЙ) ПРАКТИКИ**

(2023/2024 учебный год)

                                         Молев Павел Олегович

Направление подготовки 09.03.01 «Информатика и вычислительная техника»

Наименование профиля подготовки «Прикладной искусственный интеллект»

Форма обучения – очная Срок обучения в соответствии с ФГОС – 4 года

Год обучения1семестр 2

Период прохождения практики с 25.06.24 по 08.07.24

Кафедра«Вычислительная техника»

Молев П.О. выполнял практическое задание «Сортировка методом Шелла». Произвел тестирование, получил и проанализировал результаты, выявил закономерность увеличения работы программы.

Бакалавр Молев П.О. \_\_\_\_\_\_\_\_\_ "\_\_\_" \_\_\_\_\_\_\_\_\_ 2024 г.

Руководитель Карамышева Н.С. \_\_\_\_\_\_\_\_\_"\_\_\_" \_\_\_\_\_\_\_\_\_ 2024 г.

практики

**Содержание**

Введение ..................................................................................................2

1 Процесс тестирования........................................................................3

2 Тестирование на разных наборах данных........................................4

3 Заключение..........................................................................................6

4 Литература...........................................................................................7

5 Приложение А....................................................................................8

6 Прилоложение B...............................................................................13

**Введение**

Microsoft Visual Studio представляет собой мощную программную среду для разработки приложений под операционную систему Windows. Она включает в себя редактор кода с поддержкой IntelliSense для удобства написания кода и базового рефакторинга, отладчик для исправления ошибок, редактор форм для создания пользовательских интерфейсов, веб-редактор, дизайнер классов и инструменты для работы с базами данных.

Visual Studio также предоставляет возможность расширения функциональности через сторонние плагины, которые добавляют поддержку систем контроля версий, новые инструменты для разработки на специализированных языках программирования и другие аспекты разработки ПО.

Основные компоненты включают Visual Basic.NET, Visual C++, Visual C#, Visual F#, обеспечивая разработку на различных языках семейства Microsoft. Visual Studio предоставляет высокоуровневые функциональные возможности, включая встроенный веб-сервер, поддержку множества языков, интуитивный стиль кодирования, ускоренную разработку и эффективные инструменты отладки.

1. **Процесс тестирования**

В ходе моего исследования я проводил серию тестов, чтобы изучить время, необходимое для сортировки различных массивов данных. Каждый раз я запускал программу с новыми параметрами, начиная с небольшого массива из 100 чисел и постепенно увеличивая его размер до 100 000 чисел. Это позволило мне наблюдать, как меняется время выполнения сортировки по мере роста размера данных.

Записывая результаты тестов в таблицу, я фиксировал время, затраченное на сортировку каждого массива. Важно отметить, что с увеличением размера массива время сортировки также увеличивалось. Этот процесс позволил мне собрать достаточно данных для анализа.

После завершения экспериментов я визуализировал полученные результаты, построив график, который наглядно демонстрирует, как изменяется время выполнения сортировки в зависимости от размера массива. Такой подход не только помог мне понять поведение алгоритма сортировки на различных объемах данных, но и предоставил ценные данные для дальнейших исследований и оптимизации программного обеспечения.

1. **Тестирование на разных наборах данных.**

Тестовый набор данных представлен в таблице 1. Результаты тестирования приведены в Приложении А на рисунках А.1 - А.12.

Таблица 1 – Тестовый набор данных

|  |  |  |
| --- | --- | --- |
| № теста | Размер массива size | Время выполнения сортировки в секундах |
| 1 | 100 | 0.0003401 |
| 2 | 1 000 | 0.0008574 |
| 3 | 5 000 | 0.0020383 |
| 4 | 10 000 | 0.0042771 |
| 5 | 25 000 | 0.0090834 |
| 6 | 40 000 | 0.0144273 |
| 7 | 50 000 | 0.0272764 |
| 8 | 60 000 | 0.032332 |
| 9 | 70 000 | 0.0396772 |
| 10 | 80 000 | 0.0481651 |
| 11 | 90 000 | 0.0558291 |
| 12 | 100 000 | 0.0604902 |

Рисунок 1 - результаты тестирования

1. **Заключение**

На основании анализа данных, полученных в результате тестирования алгоритма быстрой сортировки, можно сделать вывод, что время, затраченное на работу программы относительно количества элементов увеличивается линейно, то есть с увеличением количества элементов пропорционально увеличивается время работы программы.

В ходе разработки данного приложения на C# с использованием Windows Forms я протестировал приложение на время выполнения сортировки.

Приложение демонстрирует основные возможности разработки на C# с использованием Windows Forms, включая создание и управление графическим интерфейсом, работу с файлами и массивами данных, а также использование таймеров для измерения производительности сортировки.

Сделал вывод о том, что программа корректно работает и интерфейс программы понятен и программой удобно пользоваться. Таймер корректно показывает время выполнения сортировки.

Выявил закономерность изменения времени сортировки при разном количестве элементов в массиве.

1. **Литература**

"Programming Windows Forms in C#"( Чарльз Петцольд)

"C# 8.0 and .NET Core 3.0 – Modern Cross-Platform Development"( Марк Дж. Прайс)

"Introduction to Algorithms"( Томас Х. Кормен, Чарльз Э. Лейзерсон, Рональд Л. Ривест, Клиффорд Штейн)

"The C# Programming Yellow Book"( Роб Майлз)

"Pro C# 8 with .NET Core: Foundational Principles and Practices in Programming"(Эндрю Троелсен, Филип Япиксе)

"Data Structures and Algorithms in C#"( Роберт Лафоре)

"Windows Forms Programming in C#"( Крис Селлс)

"C# in Depth"( Джон Скит)

"Algorithms in C#, Parts 1-4: Fundamentals, Data Structures, Sorting, Searching"(Роберт Седжвик)

"CLR via C#"(Джеффри Рихтер)

1. **Приложение А**

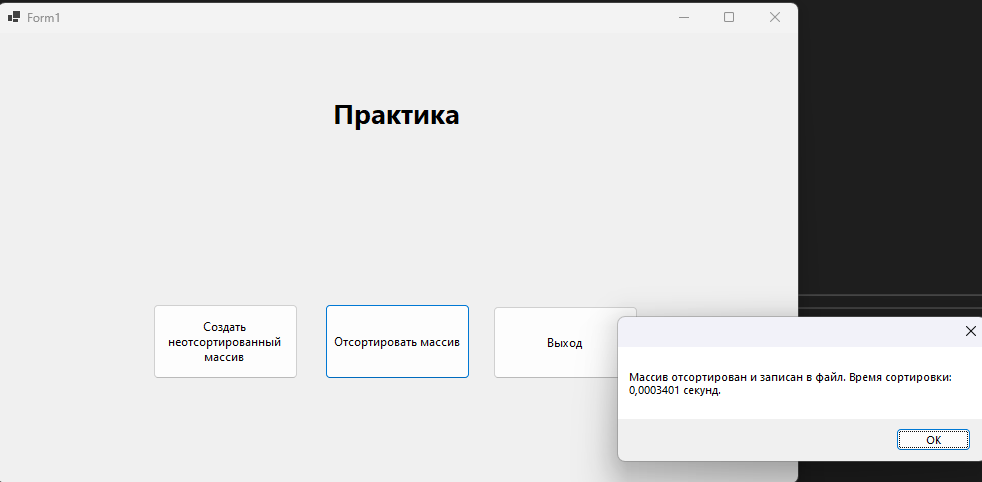


Рисунок А.1

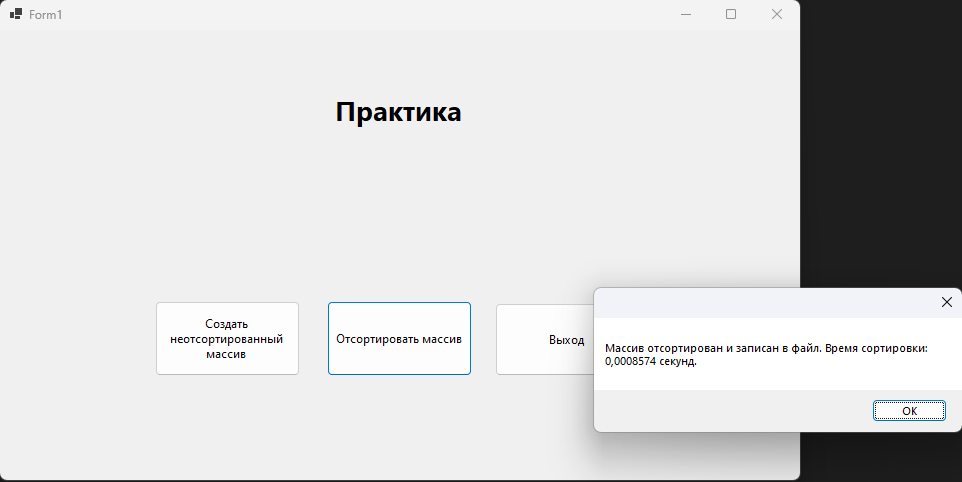


Рисунок А.2

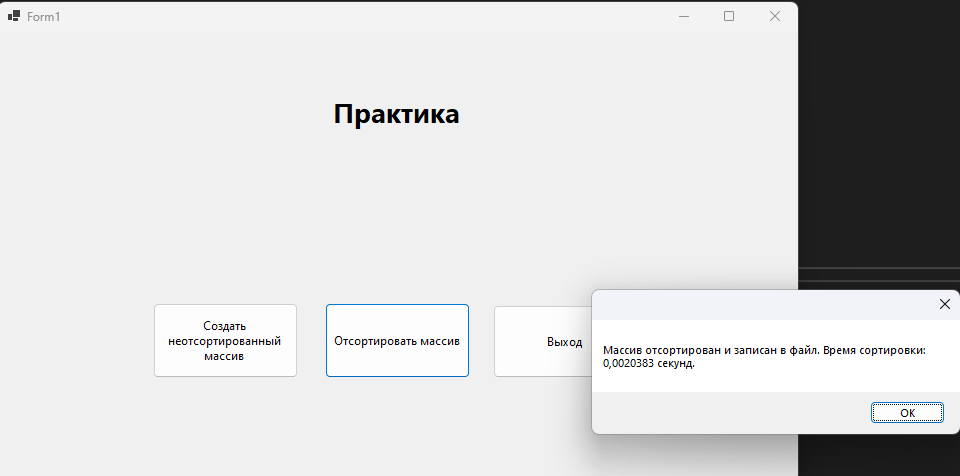


Рисунок А.3

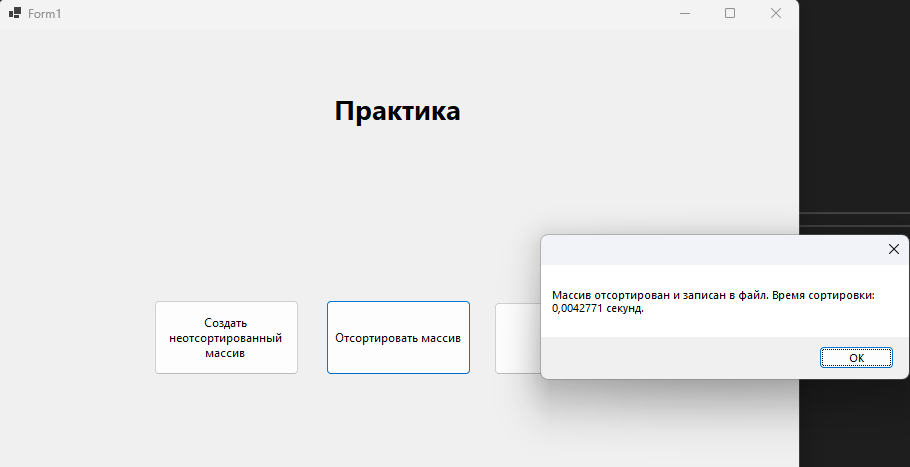


Рисунок А.4

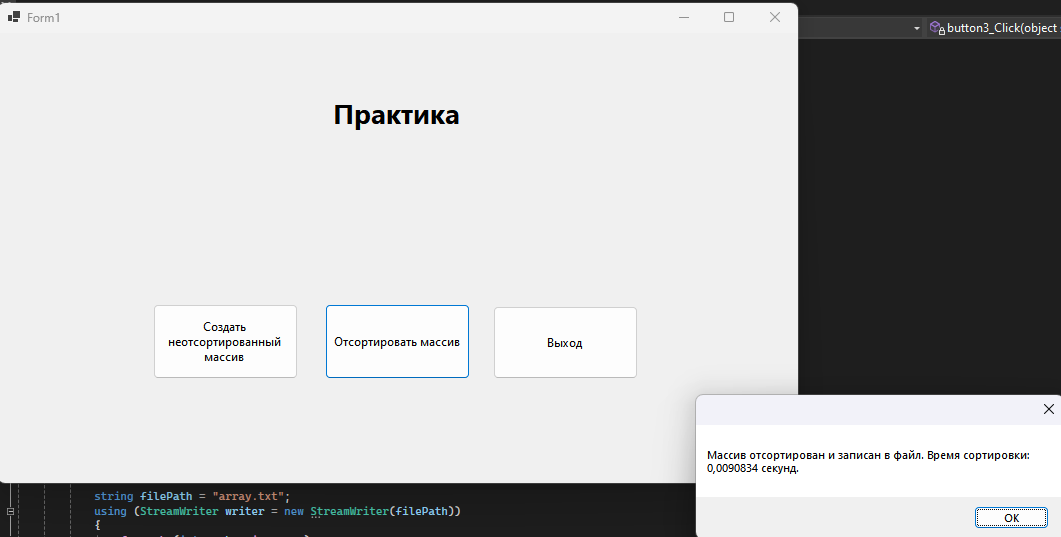


Рисунок А.5

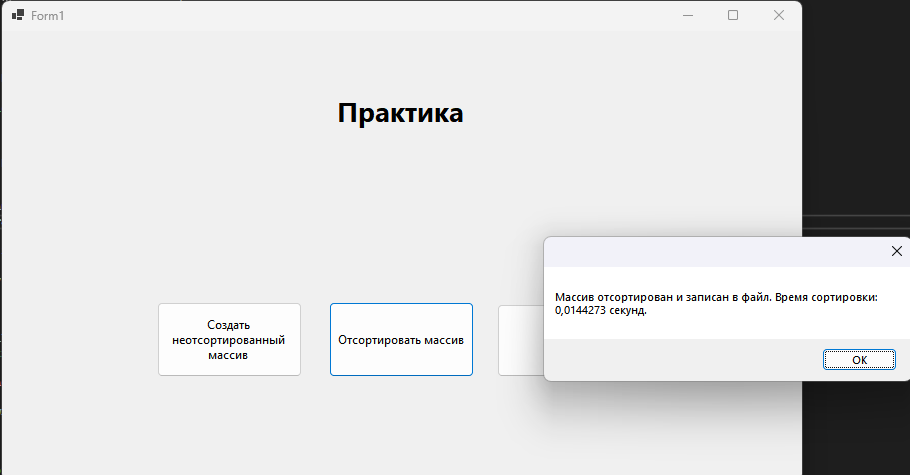


Рисунок А.6

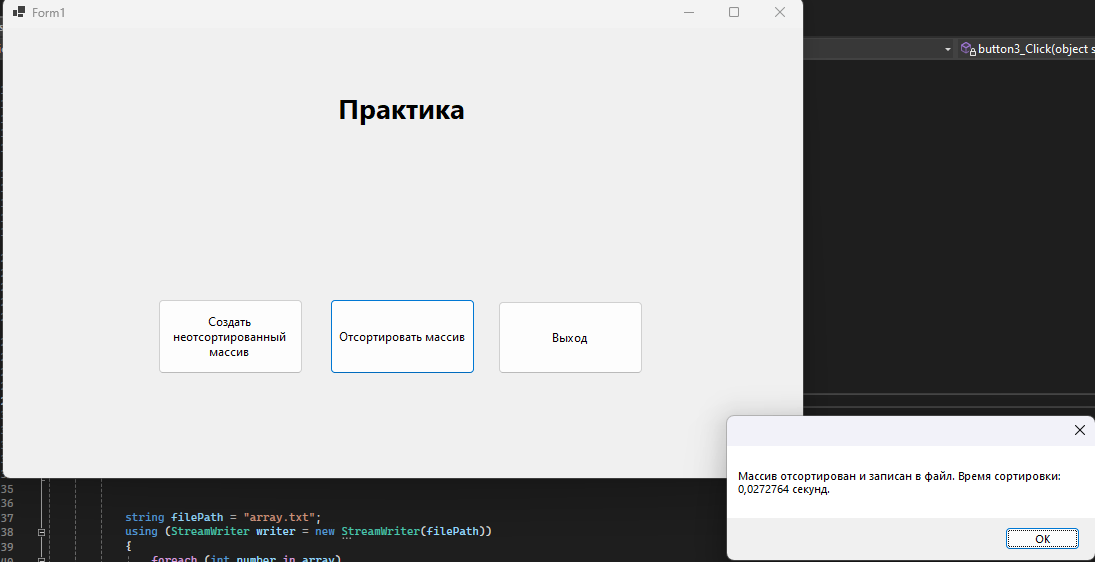


Рисунок А.7

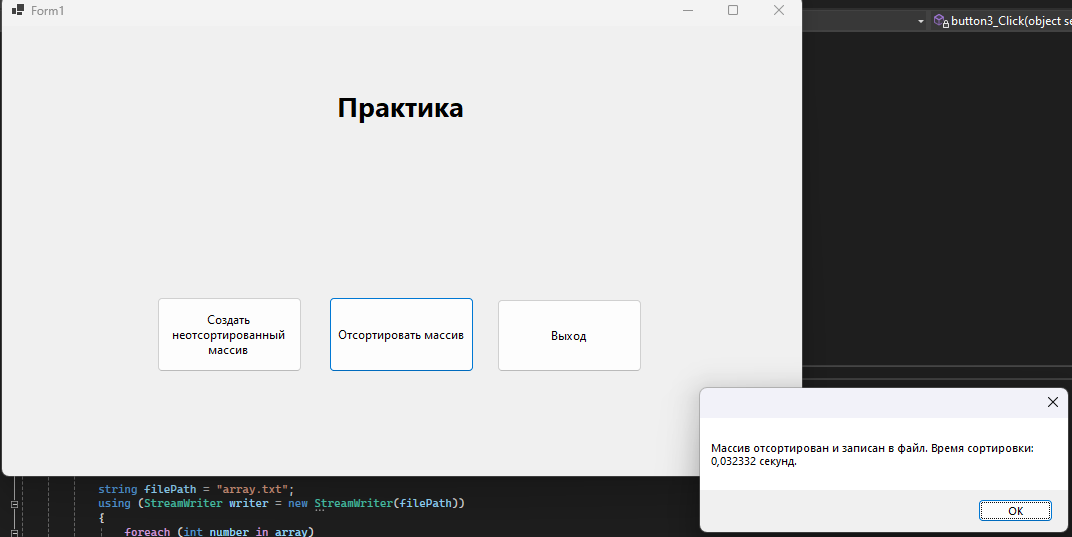


Рисунок А.8

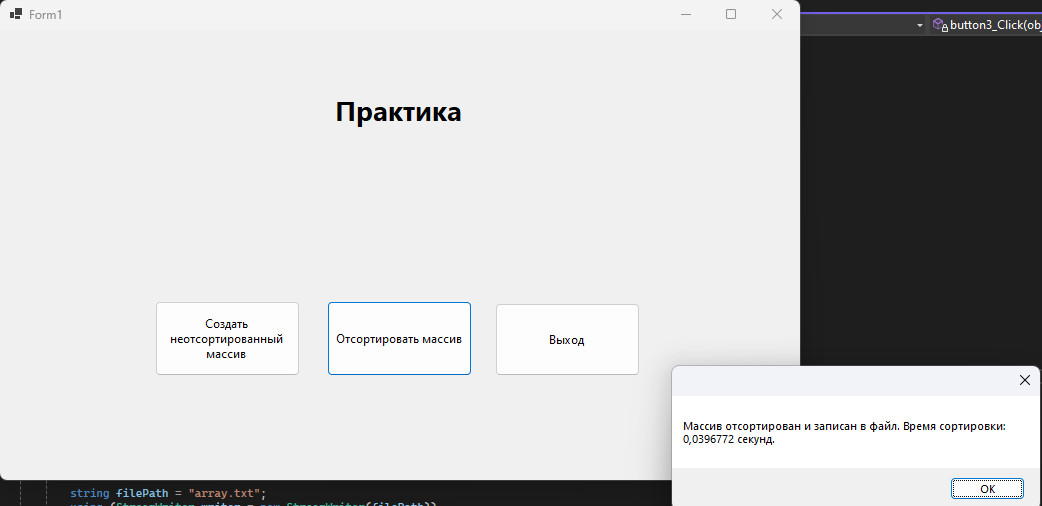


Рисунок А.9

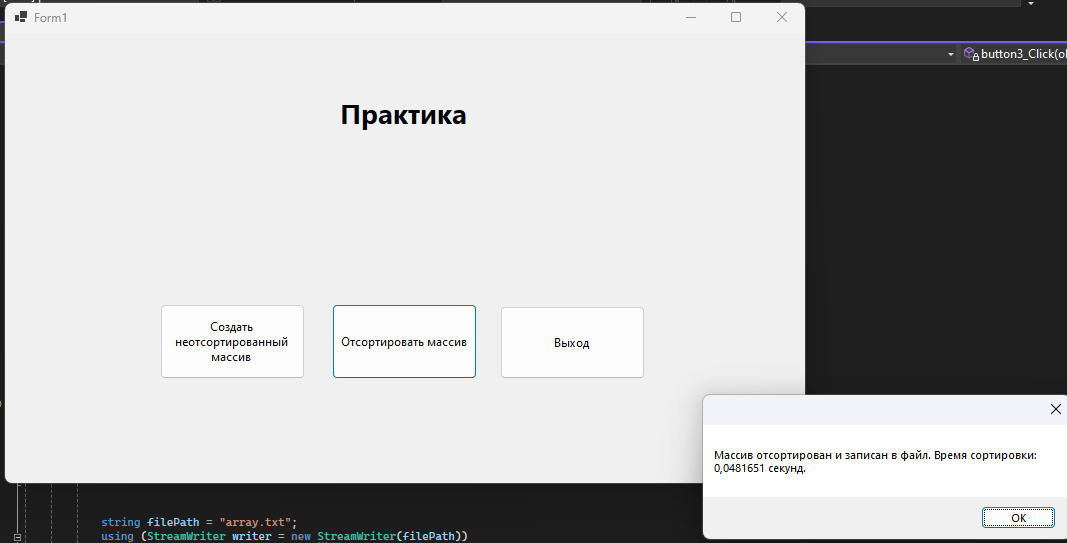


Рисунок А.10

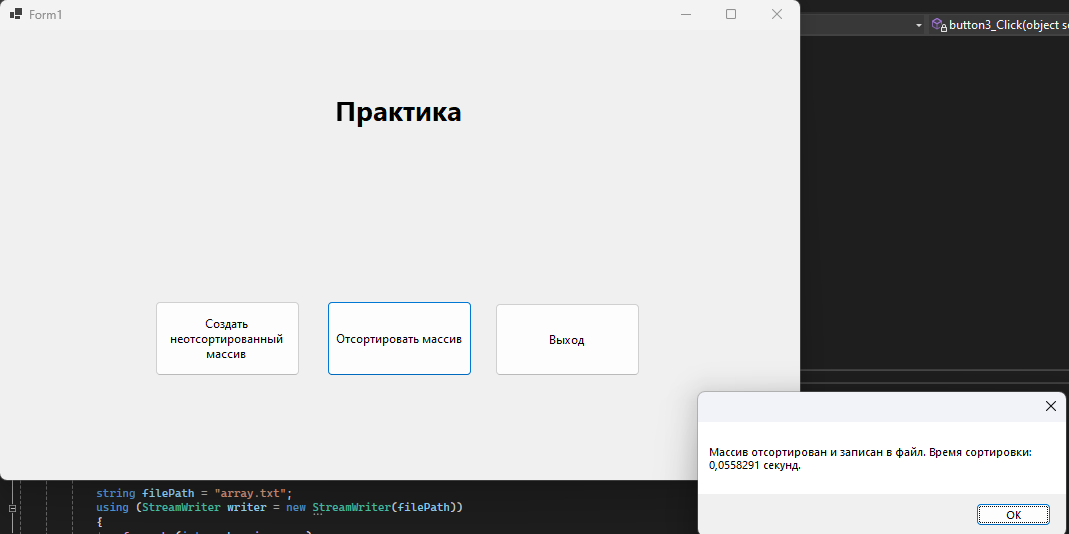


Рисунок А.11

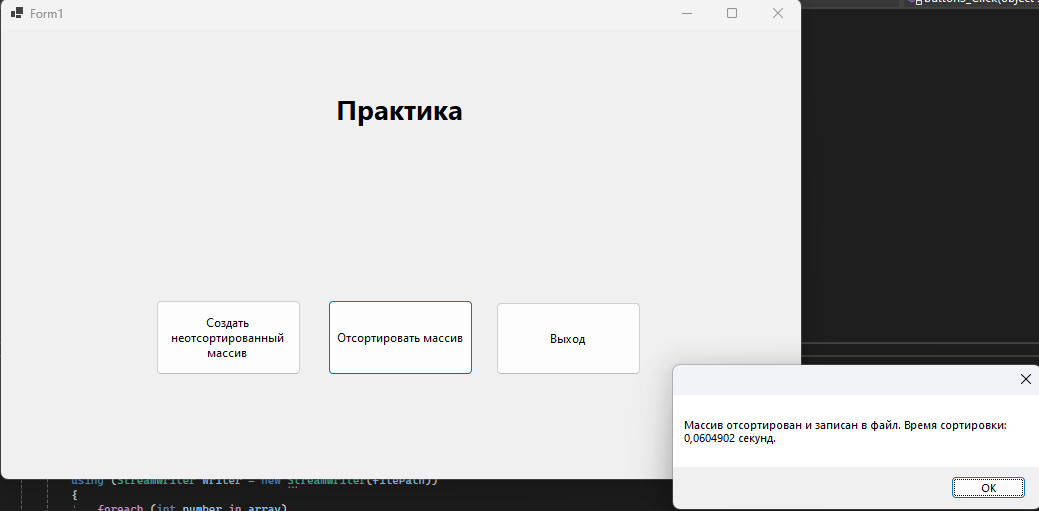


Рисунок А.12

1. **Прилоложение B.**

Листинг программы

using System;

using System.Diagnostics;

using System.IO;

using System.Windows.Forms;

using System.Diagnostics;

namespace practice

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

Application.Exit();

}

private void button3\_Click(object sender, EventArgs e)

{

Random random = new Random();

int[] array = new int[100];

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

{

array[i] = random.Next(-10001, 10001);

}

string filePath = "array.txt";

using (StreamWriter writer = new StreamWriter(filePath))

{

foreach (int number in array)

{

writer.WriteLine(number);

}

}

MessageBox.Show("Файл с неотсортированным массивом создан.");

}

private void button1\_Click\_1(object sender, EventArgs e)

{

string filePath = "array.txt";

if (!File.Exists(filePath))

{

MessageBox.Show("Файл с массивом не найден.");

return;

}

string[] lines = File.ReadAllLines(filePath);

int[] array = new int[lines.Length];

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

{

array[i] = int.Parse(lines[i]);

}

Stopwatch stopwatch = new Stopwatch();

stopwatch.Start();

array = ShellSort(array);

stopwatch.Stop();

string sortedFilePath = "sorted\_array.txt";

using (StreamWriter writer = new StreamWriter(sortedFilePath))

{

foreach (int number in array)

{

writer.WriteLine(number);

}

}

MessageBox.Show($"Массив отсортирован и записан в файл. Время сортировки: {stopwatch.Elapsed.TotalSeconds} секунд.");

}

private void label1\_Click(object sender, EventArgs e)

{

}

}

}