**Оглавление**

[Введение 3](#_Toc130331211)

[Глава 1. Основная часть 5](#_Toc130331212)

[1.1 Техническое задание 5](#_Toc130331213)

[1.2 Спецификация на программный продукт 7](#_Toc130331214)

[1.4 Разработка функциональной схемы программного продукта, составление блок-схемы программных модулей программного продукта 10](#_Toc130331215)

[1.5 Разработка кода программных модулей программного продукта 15](#_Toc130331216)

[1.6 Разработка пользовательского интерфейса программного продукта в визуальной среде 18](#_Toc130331217)

[1.7 Выполнить интеграцию программных модулей в программный продукт 19](#_Toc130331218)

[1.8 Разработка процедуры тестирования программного продукта. Выполнение тестирование программного продукта 21](#_Toc130331219)

[1.9 Разработка справочной системы программного продукта 24](#_Toc130331220)

[1.10 Разработка руководства оператора (пользователя) 26](#_Toc130331221)

[Заключение 28](#_Toc130331222)

[Список использованных источников 30](#_Toc130331223)

# **Введение**

Сортировка массива является одним из фундаментальных алгоритмов, используемых в программировании и информатике. Она позволяет упорядочить элементы в массиве по заданному критерию, такому как возрастание или убывание значения элементов. Это может быть полезно во многих приложениях, включая поиск, фильтрацию, анализ данных и т.д.

Существует множество алгоритмов сортировки массива, каждый из которых имеет свои преимущества и недостатки в зависимости от конкретной задачи. Некоторые из самых популярных алгоритмов включают сортировку пузырьком, сортировку вставками, гномью сортировку и сортировку выбором.

В данном техническом задании мы рассмотрим разработку программы на языке C# с использованием Windows Form, которая позволит пользователю вводить данные в массив, а затем сортировать их с использованием различных методов сортировки. Разработка такой программы может быть полезна как для начинающих программистов, так и для опытных специалистов, которые хотят изучить различные методы сортировки и их применение в реальных приложениях

Цель: разработать программу для сортировки массива значений на языке C# с использованием Windows Form.

Задачи проекта по разработке программы для сортировки массива на C# с помощью Windows Form могут включать:

* создание удобного и интуитивно понятного интерфейса для ввода массива и выбора метода сортировки;
* разработка функции, позволяющей пользователю вводить данные массива, которая может включать проверку входных данных;
* реализация алгоритмов сортировки, таких как пузырьком, вставкой, гномовая сортировка и сортировка выбором, для сортировки данных массива;
* предоставление пользователю возможности выбрать желаемую функцию сортировки, которая будет применена к входному массиву;
* отображение данных отсортированного массива в пользовательском интерфейсе в легко читаемом формате;
* тестирование программы, чтобы убедиться, что она функционирует должным образом и что алгоритмы сортировки работают правильно;
* оптимизация программы для повышения производительности и эффективности, например, уменьшение количества ненужных вычислений и минимизация использования памяти.

# **Глава 1. Основная часть**

* 1. **Техническое задание**

Техническое задание — это документ, в котором фиксируются требования к проекту. Условно ТЗ можно назвать любое поручение исполнителю, главное, чтобы в нем были ясно прописаны характеристики итогового продукта [1].

Требования:

* приложение должно иметь возможность сортировки массива значений с использованием следующих методов сортировки: сортировка пузырьком, сортировка вставками, гномья сортировка и сортировка выбором;
* пользователь должен иметь возможность ввести массив значений в приложение;
* после сортировки массив должен выводиться в приложении;
* приложение должно быть разработано с использованием Windows Form;
* пользователь должен иметь возможность выбрать метод сортировки из списка;
* приложение должно иметь возможность обработки ошибок, связанных с некорректным вводом данных;
* приложение должно быть разработано на языке C# с использованием Visual Studio;
* приложение должно быть документировано в соответствии с требованиями к документации.

Технические требования:

* интерфейс должен быть понятным и удобным для пользователя;
* приложение должно быть стабильным и не должно выдавать ошибок при работе;
* приложение должно иметь хорошую производительность и быстро сортировать большие массивы данных;
* приложение должно быть написано с использованием объектно-ориентированного программирования;
* приложение должно быть протестировано на разных тестовых данных;
* код приложения должен соответствовать стандартам программирования на C#;
* приложение должно быть разработано с использованием шаблонов проектирования и паттернов архитектуры;
* приложение должно иметь возможность масштабирования для работы с большими объемами данных.

Функциональные требования:

* пользователь должен иметь возможность вводить данные вручную;
* программа должна иметь вводимые значения типа int, размерность массива данных доходить до 1000 чисел;
* время сортировки программы должна занимать не более пол секунды;
* программа должна стремиться к сортировке данных по возрастанию;
* пользователь должен иметь возможность выбирать метод сортировки из списка доступных методов;
* пользователь должен видеть отсортированный массив в приложении;
* приложение должно работать быстро и эффективно для сортировки больших массивов данных;
* приложение должно быть удобным и простым в использовании.

Системные требования для работы программы сортировки массива значений на языке C# с использованием Windows Form могут быть следующими:

* операционная система: Windows 7/8/10;
* процессор: Intel Core i3 или выше;
* оперативная память: не менее 4 ГБ;
* свободное место на жестком диске: не менее 100 МБ;
* графический адаптер: поддержка DirectX 9 и выше;
* разрешение экрана: не менее 1280x768 пикселей;
* Microsoft .NET Framework версии 4.7 или выше должен быть установлен на компьютере;
* рекомендуется наличие клавиатуры и мыши для ввода и управления данными в приложении.
  1. **Спецификация на программный продукт**

Спецификация требований, также известная как документация, представляет собой процесс записи всех системных и пользовательских требований в форме документа. Эти требования должны быть четкими, полными, всеобъемлющими и последовательными [2].

Требования:

* программа должна быть разработана на C# с использованием Windows Forms;
* пользователь должен иметь возможность вводить значения в массив;
* программа должна проверить правильность введенных значений, чтобы убедиться, что они являются действительными числами;
* пользователь должен иметь возможность выбрать один из четырех алгоритмов сортировки: пузырьковая сортировка, сортировка вставками, карликовая сортировка или сортировка выбором;
* программа должна отображать отсортированный массив после сортировки;
* программа должна включать таймер для измерения времени работы алгоритма сортировки;
* программа должна отображать время работы алгоритма сортировки;
* в программе должна быть кнопка "сброс" для очистки полей ввода и вывода;

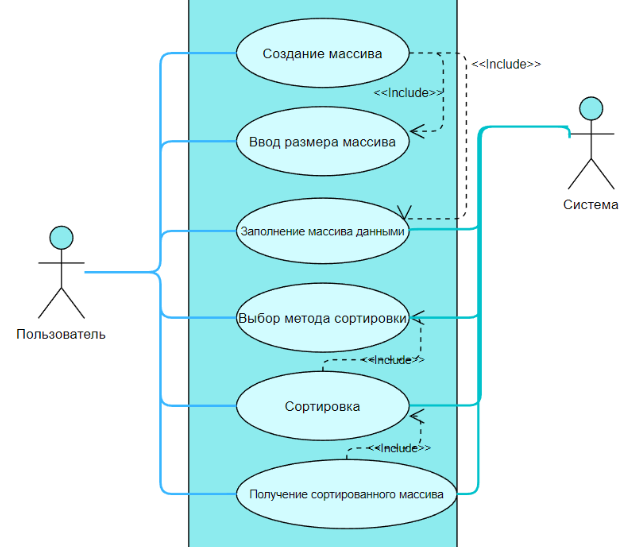
Интерфейс программы должен включать следующие элементы:

* текстовое поле для ввода значений в массив;
* раскрывающийся список или переключатели для выбора алгоритма сортировки;
* программа должна иметь вводимые значения типа int, размерность массива данных доходить до 1000 чисел;
* время сортировки программы должна занимать не более пол секунды;
* программа должна стремиться к сортировке данных по возрастанию;
* кнопка запуска алгоритма сортировки;
* текстовое поле для отображения отсортированного массива;
* текстовое поле для отображения времени работы алгоритма сортировки;
* кнопка «сброс» для очистки полей ввода и вывода.

программа должна включать следующие алгоритмы сортировки:

* пузырьковая сортировка;
* сортировка вставками;
* гномья сортировка;
* сортировка выбором.
  1. **Разработка функциональной диаграммы программного продукта, диаграмма потоков данных программных модулей продукта**

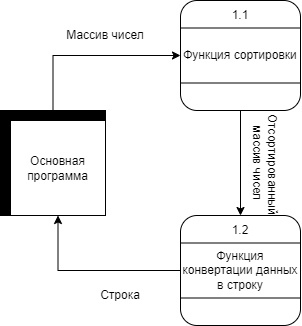
Функциональными называют диаграммы, в первую очередь отражающие взаимосвязи функций разрабатываемого программного обеспечения [3].



1. UML диаграмма

На данном рисунке представлена UML диаграмма программного комплекса. Субъекты диаграммы – пользователь и система. Пользователь задает размер массива и его наполнение, а также выбирает метод сортировки. Система считывает массив и сортирует его с помощью выбранного метода, после чего выводит пользователю.

Диаграммы потоков данных – представляют собой иерархию функциональных процессов, связанных потоками данных. Цель такого представления — продемонстрировать, как каждый процесс преобразует свои входные данные в выходные, а также выявить отношения между этими процессами [4].

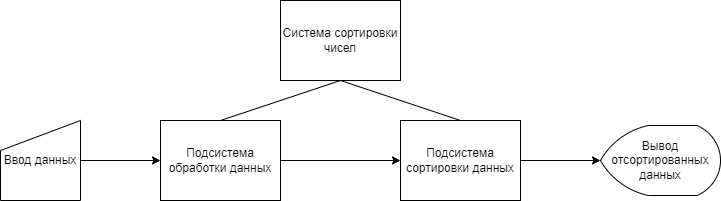


1. диаграмма потоков данных

На данном изображении представлена диаграмма потока данных программного комплекса.

* 1. **Разработка функциональной схемы программного продукта, составление блок-схемы программных модулей программного продукта**

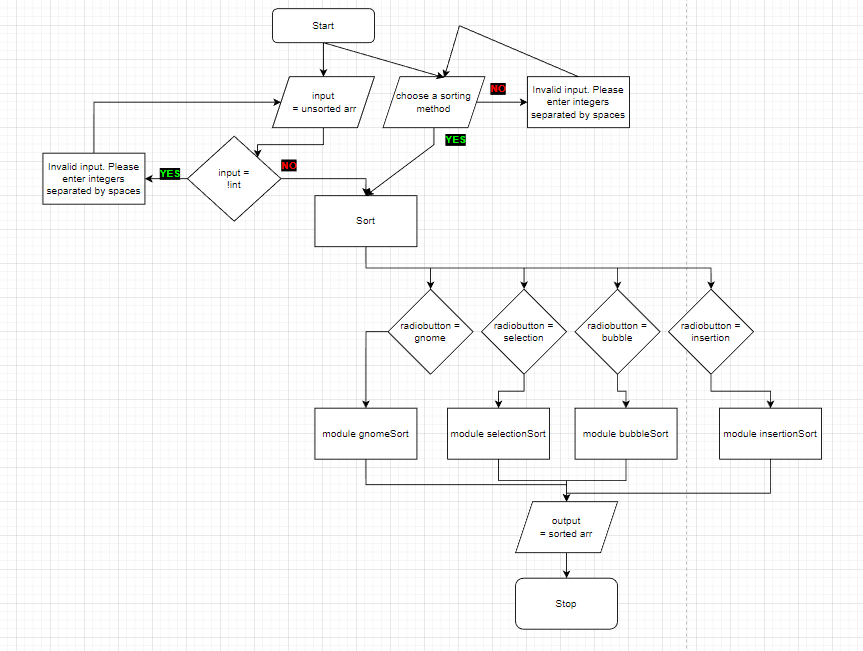
Функциональная схема — это схема взаимодействия компонентов программного обеспечения с описанием информационных потоков, состава данных в потоках и указанием используемых файлов и устройств.



1. функциональная схема продукта

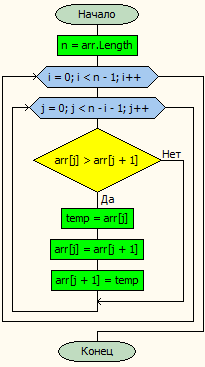
На данном рисунке обозначена функциональная схема продукта.

Блок-схема — это схематичное представление процесса, системы или компьютерного алгоритма. Блок-схемы часто применяются в разных сферах деятельности, чтобы документировать, изучать, планировать, совершенствовать и объяснять сложные процессы с помощью простых логичных диаграмм. Для построения блок-схем применяются прямоугольники, овалы, ромбы и некоторые другие фигуры (для обозначения конкретных операций), а также соединительные стрелки, которые указывают последовательность шагов или направление процесса [5].



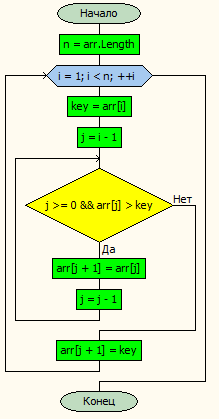
1. самая главная блок-схема

Изображена главная блок-схема продукта, на которой изображены различные блоки, такие как: несколько видов условий, выводов, вводов и обычных блоков с действиями.



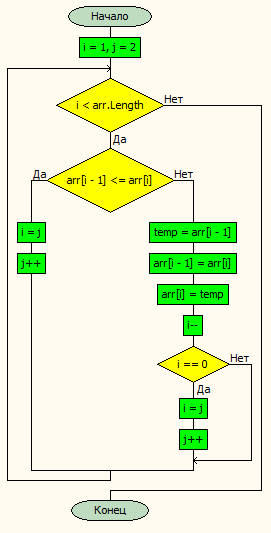
1. блок-схема модуля сортировки пузырьком

На данном рисунке изображена блок схема модуля сортировки пузырьком.



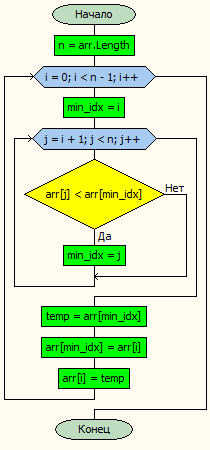
1. блок-схема модуля сортировки вставкой

На данном изображении изображена блок-схема модуля сортировки вставкой.



1. блок-схема модуля сортировки гномом

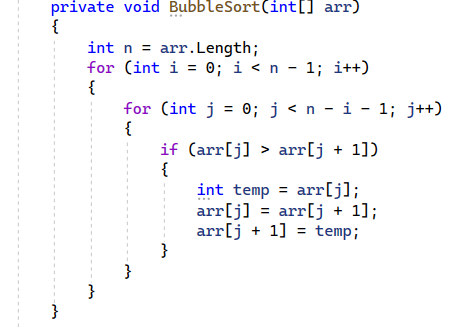
На данном рисунке показана блок-схема модуля сортировки гномом.



1. блок-схема модуля сортировки выбором

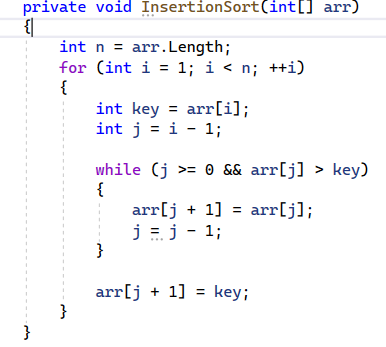
На данном изображении изображена блок-схема модуля сортировки выбором.

* 1. **Разработка кода программных модулей программного продукта**



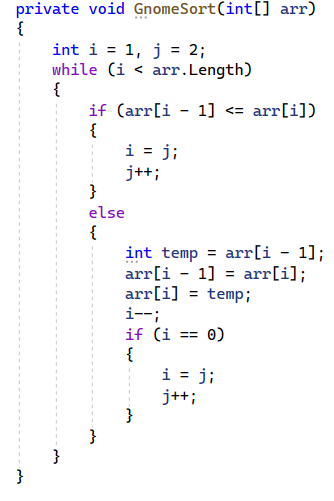
1. модуль сортировка Пузырьком

На данном рисунке предоставлен код сортировки пузырьком. Сортировка пузырьком — один из самых известных алгоритмов сортировки. Здесь нужно последовательно сравнивать значения соседних элементов и менять числа местами, если предыдущее оказывается больше последующего. Таким образом элементы с большими значениями оказываются в конце списка, а с меньшими остаются в начале.



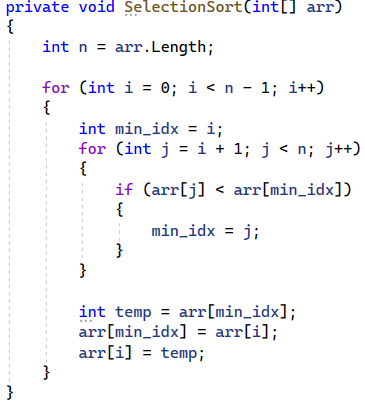
1. модуль сортировка Вставкой

Показан код сортировка вставкой. На каждом шаге алгоритма мы выбираем один из элементов входных данных и вставляем его на нужную позицию в уже отсортированном списке до тех пор, пока набор входных данных не будет исчерпан. Метод выбора очередного элемента из исходного массива произволен; может использоваться практически любой алгоритм выбора. Обычно, элементы вставляются по порядку их появления во входном массиве.



1. модуль сортировка Гномом

Алгоритм находит первое место, где два соседних элемента находятся в неправильном положении заказ и меняет их местами. Он использует преимущество того факта, что выполнение обмена может ввести новую неупорядоченную смежную пару рядом с ранее замененными элементами. Он не предполагает, что элементы, расположенные впереди текущей позиции, сортируются, поэтому ему нужно только проверить позицию непосредственно перед заменяемыми элементами.



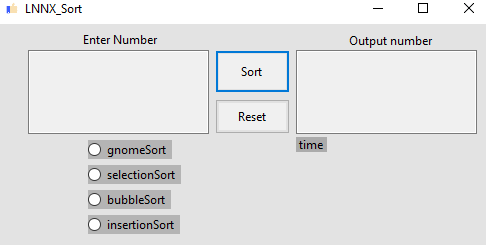
1. модуль сортировка Выбором

Данный код представляет собой сортировку выбором. Алгоритм ищет наименьший элемент в текущем списке и производит обмен его значения со значением первой неотсортированной позиции. То же самое происходит со вторым элементом с наименьшим значением. Цикл повторяется до тех пор, пока все элементы не займут нужную последовательность.

* 1. **Разработка пользовательского интерфейса программного продукта в визуальной среде**

Для создания пользовательского интерфейса, будет использоваться Windows Form 2019 года. Будут использоваться такие компоненты как:

* button – кнопка, используется для двух действий, для сортировки и сброса значений;
* radioButton – используется для выбора метода сортировки;
* textbox – предназначен для ввода и вывода значений;
* label – применяется для обозначения кнопок и вывода таймера;
* form – главная форма, на которой добавляются все компоненты, обозначенные выше.

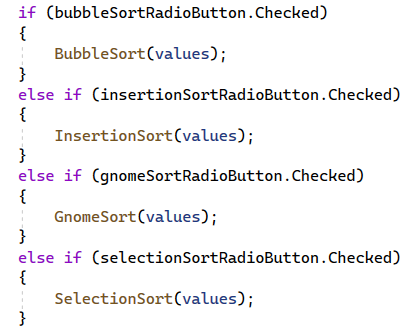


1. интерфейс приложения

Создан простой «friendly» интерфейс, с интуитивным управлением. Нужно ввести в левом textBox цифры для сортировки, выбрать способ сортировки и нажать кнопку Sort. В правом textbox получим отсортированные числа и время сортировки. Кнопка Reset предназначена для сброса всех значений.

* 1. **Выполнить интеграцию программных модулей в программный продукт**

Для интеграции программных модулей требуется в основном модуле воспользоваться вызовом тех модулей, которые нам нужны.



1. интеграция всех программных модулей: BubbleSort, InsertionSort, GnomeSort, SelectionSort.
   1. **Разработка процедуры тестирования программного продукта. Выполнение тестирование программного продукта**

Для тестирования программного продукта "Сортировка массива" можно разработать следующую систему тестирования:

* тестирование ввода данных;
* проверка корректности ввода чисел в массив;
* проверка наличия ошибок при вводе данных (например, некорректных символов или пустых значений);
* тестирование выбора метода сортировки;
* проверка корректности выбора метода сортировки;
* проверка наличия ошибок при выборе метода сортировки (например, некорректных символов или выбора несуществующего метода);
* тестирование сортировки массива;
* проверка корректности работы каждого метода сортировки на разных массивах данных;
* проверка наличия ошибок при сортировке (например, ошибок в логике сортировки или возможных переполнениях);
* тестирование вывода отсортированного массива данных;
* проверка корректности вывода отсортированного массива данных;
* проверка наличия ошибок при выводе (например, некорректных символов или пустых значений);
* тестирование вычисления времени выполнения;
* проверка корректности вычисления времени выполнения для каждого метода сортировки на разных массивах данных;
* проверка наличия ошибок при вычислении времени выполнения (например, некорректных символов или пустых значений);
* тестирование работы программы в целом;
* проверка наличия ошибок при работе программы в различных сценариях использования (например, ошибок в логике программы или возможных переполнениях);
* тестирование производительности;
* проверка производительности программы при работе с массивами разных размеров и методах сортировки;
* оценка времени выполнения и использования ресурсов при работе с массивами разных размеров и методах сортировки.

Протокол тестирования программного продукта "Сортировка массива"

Дата тестирования: [19.03.2023]

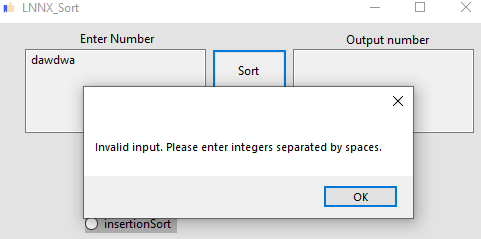
Тестировщик: [LNNX]

Тестирование будет проводиться на компьютере оснащенным процессором AMD FX 8350, видеокартой Palit GeForce GTX 1050 Ti, ОЗУ 8 GB.

1. Тестирование ввода данных.

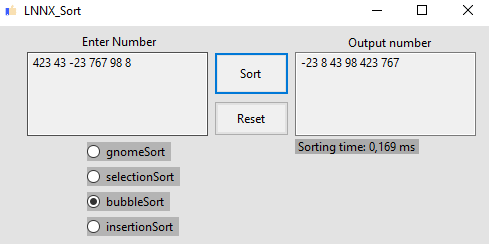
* проверка корректности ввода чисел в массив: НЕ ПРОЙДЕНО;

При вводе не натурального числа или был произведен ввод букв, выдает ошибку:



1. ошибка на ввод не натурального числа или ввод букв

При вводе натурального числа и выборе способа сортировки, программа не будет выдавать ошибок, а сделает сортировку массива:



1. правильный ввод данных в программу

* проверка корректности ввода чисел в массив: ПРОЙДЕНО.
* проверка наличия ошибок при вводе данных: ПРОЙДЕНО.

1. Тестирование выбора метода сортировки.

* проверка корректности выбора метода сортировки: ПРОЙДЕНО;
* проверка наличия ошибок при выборе метода сортировки: ПРОЙДЕНО.

1. Тестирование сортировки массива.

* проверка корректности работы каждого метода сортировки на разных массивах данных: ПРОЙДЕНО;
* проверка наличия ошибок при сортировке: ПРОЙДЕНО.

1. Тестирование вывода отсортированного массива данных.

* проверка корректности вывода отсортированного массива данных: ПРОЙДЕНО;
* проверка наличия ошибок при выводе: ПРОЙДЕНО.

1. Тестирование вычисления времени выполнения.

* проверка корректности вычисления времени выполнения для каждого метода сортировки на разных массивах данных: ПРОЙДЕНО;
* проверка наличия ошибок при вычислении времени выполнения: ПРОЙДЕНО.

1. Тестирование работы программы в целом.

* проверка корректности работы программы в различных сценариях использования: ПРОЙДЕНО;
* проверка наличия ошибок при работе программы в различных сценариях использования: ПРОЙДЕНО.

1. Тестирование производительности.

* проверка производительности программы при работе с массивами разных размеров и методах сортировки: ПРОЙДЕНО;
* оценка времени выполнения и использования ресурсов при работе с массивами разных размеров и методах сортировки: ПРОЙДЕНО.

Выводы: программный продукт "Сортировка массива" прошел все тесты и работает корректно в различных сценариях использования. Производительность программы также соответствует требованиям к производительности. Все виды сортировок работают быстро. Рекомендуется продолжать использование программы. Замечание и ошибки отсутствуют.

* 1. **Разработка справочной системы программного продукта**

Для программного продукта Сортировка массива можно разработать следующую справочную систему:

* введение;
* описание программного продукта;
* инструкция по установке;
* требования к системе;
* основные функции;
* ввод массива данных;
* выбор метода сортировки;
* сортировка массива данных;
* вывод отсортированного массива данных;
* вычисление времени выполнения;
* использование программы;
* шаги по запуску и использованию программы;
* инструкция по вводу данных и выбору метода сортировки;
* инструкция по просмотру отсортированных данных и времени выполнения
* методы сортировки;
* описание методов сортировки, доступных в программе (bubble sort, insertion sort, dwarf sort, selection sort);
* примеры использования каждого метода сортировки;
* ошибки и их исправление;
* список наиболее распространенных ошибок, которые могут возникнуть при использовании программы;
* инструкции по их исправлению;
* справочная информация;
* общая информация о сортировке массивов;
* ссылки на дополнительные ресурсы, связанные с сортировкой массивов и программированием на c#;
* контакты разработчиков;
* контактная информация разработчиков программного продукта;
* инструкции по отправке отзывов и сообщений об ошибках.

Справочная система помогает пользователям понять, как использовать программу и решать возможные проблемы, которые могут возникнуть в процессе ее использования. Это также может помочь новым пользователям ознакомиться с основами сортировки массивов и программирования на C#.

* 1. **Разработка руководства оператора (пользователя)**

Руководство пользователя программы "Сортировка массива"

1. Для запуска программы "Сортировка массива" необходимо дважды щелкнуть на файле .exe или выбрать программу из меню "Пуск".
2. Введите числа в поле ввода массива, разделяя их пробелами. Нажмите кнопку "Добавить", чтобы добавить массив в список массивов. Вы можете добавлять несколько массивов.
3. Выберите один из четырех методов сортировки: "Bubble sort", "Insertion sort", "Dwarf sort" или "Selection sort". Можно выбрать только один метод сортировки.
4. Нажмите кнопку "Сортировать", чтобы отсортировать выбранный массив методом сортировки, выбранным на предыдущем шаге. Отсортированный массив будет отображен в поле вывода.
5. Время выполнения метода сортировки для выбранного массива будет отображено в поле "Время выполнения".
6. Для удаления массивов из списка выберите нужный массив и нажмите кнопку "Backspace".
7. Для очистки списка массивов нажмите кнопку "Reset".
8. Чтобы выйти из программы, нажмите крестик.

Важно:

* вводимые числа должны быть целыми числами, разделенными запятой или пробелом;
* программа не поддерживает сортировку массивов, содержащих более 1000 элементов;
* при сортировке массивов с большим количеством элементов методы сортировки могут выполняться дольше;
* не используйте программу для сортировки конфиденциальной информации, так как программа не обеспечивает безопасность данных.

# **Заключение**

В рамках проекта была разработана программа "Сортировка массива", которая позволяет пользователю сортировать массивы чисел с использованием четырех методов сортировки: "Bubble sort", "Insertion sort", "Dwarf sort" и "Selection sort". Кроме того, программа позволяет измерять время выполнения каждого метода сортировки и удалять или очищать список массивов.

Для создания программы были использованы язык программирования C# и платформа Windows Forms. В процессе разработки были применены различные методы и инструменты, такие как функциональная диаграмма, диаграмма потоков данных и блок-схема.

На протяжении всей работы было выполнено:

* создание удобного и интуитивно понятного интерфейса для ввода массива и выбора метода сортировки;
* разработка функции, позволяющей пользователю вводить данные массива, которая может включать проверку входных данных;
* реализация алгоритмов сортировки, таких как пузырьком, вставкой, гномовая сортировка и сортировка выбором, для сортировки данных массива;
* предоставление пользователю возможности выбрать желаемую функцию сортировки, которая будет применена к входному массиву;
* отображение данных отсортированного массива в пользовательском интерфейсе в легко читаемом формате;
* тестирование программы, чтобы убедиться, что она функционирует должным образом и что алгоритмы сортировки работают правильно;
* оптимизация программы для повышения производительности и эффективности, например, уменьшение количества ненужных вычислений и минимизация использования памяти.

Тестирование программы показало, что она работает корректно и эффективно. Однако, следует отметить, что программа не обеспечивает безопасность данных и не рекомендуется использовать для сортировки конфиденциальной информации.

В целом, проект был успешно завершен, и программа "Сортировка массива" может быть полезна для пользователей, которые нуждаются в быстрой и удобной сортировке массивов чисел.

# **Список использованных источников**

1. Контур: сайт. – URL: https://kontur.ru/articles/5945 (дата обращения: 19.03.2023)
2. Visure: сайт. – URL: https://visuresolutions.com/ru/blog/requirements-specification/ (дата обращения: 19.03.2023)
3. Функциональная схема программного продукта // videouroki: сайт. – URL: https://videouroki.net/razrabotki/funktsional-naia-skhiema-proghrammnogho-produkta.html (дата обращения: 19.03.2023)
4. Диаграммы потоков данных // samara.mgpu: сайт. – URL: https://samara.mgpu.ru/~dzhadzha/dis/15/180.html (дата обращения: 19.03.2023)
5. Что такое блок-схема? // lucidchart: сайт. – URL: https://www.lucidchart.com/pages/ru/блок-схем (дата обращения: 19.03.2023)