

Кыргызский государственный технический университет им. И.
Раззакова

Филиал им. академика Х.А. Рахматулина в г. Токмок

Кафедра «Программное обеспечение компьютерных систем»

Отчет по учебной практике

Выполнил: ст. гр. ПИ-23 Развозжаев Н.В.

Руководитель учебной практики: старший
преподаватель Забабурина М.В.

Сроки прохождения с 2.06.2025г. по 30.06.2025г.

Дата сдачи: 30.06.2025г.

Оценка: 95 (отл) М.Замур

(отметка о сдаче, роспись руководителя)

Токмок 2025 г.

**Задание на учебную практику для
студентов направления:
710400 «Программная инженерия»**

Вариант № 1

1. Архитектура современного компьютера. Основные принципы организации компьютера.

2. Файл. Файловая система. Структура, функции, виды файловых систем.

3. Написать программу для расчета по двум формулам:

$$z_1 = \left(\frac{a+2}{\sqrt{2a}} - \frac{a}{\sqrt{2a}+2} + \frac{2}{a-\sqrt{2a}} \right) \cdot \frac{\sqrt{a}-\sqrt{2}}{a+2};$$

$$z_2 = \frac{1}{\sqrt{a} + \sqrt{2}}.$$

4. Пусть дано 100 вещественных чисел. Вычислите разность между максимальным и минимальным числами.

Содержание

Введение.....	1
Задание 1.....	2
1.1. Введение в архитектуру компьютера.....	3
1.2. Основные компоненты компьютера.....	4
1.3. Архитектура фон Неймана.....	4
1.4. Гарвардская архитектура и современные тенденции.....	5
1.5. Принципы организации современных компьютеров.....	5
Задание 2.....	6
2.1. Понятие файла.....	7
2.2. Файловая система.....	8
2.3. Структура файловой системы.....	9
2.4. Функции файловой системы.....	10
2.5. Виды файловых систем.....	11
2.6. Сравнительная таблица популярных ФС.....	11
Задание 3.....	12
3.1. Создание листинга.....	12
3.2. Результат программы.....	13
Задание 4.....	14
4.1. Создание блок-схемы.....	15
4.2. Листинг программы.....	16
4.3. Результат программы.....	17
Заключение.....	18
Список используемой литературы.....	20

Введение

Современные технологии стремительно развиваются, и с каждым годом возрастает значение глубокого понимания основ вычислительной техники и программирования. Компьютерные технологии пронизывают все сферы жизни — от образования до промышленности, от медицины до искусства. В этой связи фундаментальные знания об архитектуре компьютера, принципах организации вычислительных систем, а также практические навыки программирования становятся неотъемлемой частью подготовки специалиста в области информационных технологий. Целью учебной практики является закрепление теоретических знаний и приобретение практических навыков программирования. Практическая часть работы состоит из двух программных заданий. Первое из них — написание программы, производящей вычисления по двум заданным формулам. Реализация подобных программ помогает закрепить знания по основам программирования, работе с переменными, арифметическими операциями и алгоритмами. Особое внимание уделяется корректности вычислений, учёту математических особенностей выражений и обеспечению надёжности работы программы. Второе практическое задание направлено на решение задачи анализа массива данных. Практическая работа не только формирует технические навыки, но и развивает дисциплину, логическое мышление, умение анализировать поставленную задачу и находить оптимальные пути её решения. Каждый выполненный пример и каждая программа — это шаг вперёд в понимании сути программирования и построения надёжных программных решений. Таким образом, настоящая работа охватывает как теоретические аспекты компьютерной науки, так и практические задания, направленные на развитие прикладных навыков.

Комплексное изучение архитектуры компьютеров, организации файловой системы и реализация программ на языке программирования позволяют не только расширить кругозор, но и заложить прочную основу для дальнейшего изучения информационных технологий и разработки программного обеспечения.

1. Архитектура современного компьютера. Основные принципы организации компьютера

Архитектура компьютера — это совокупность концептуальных принципов, определяющих структуру и поведение вычислительной системы. Она описывает, как различные компоненты компьютера взаимодействуют между собой и как осуществляется обработка, хранение и передача информации. Развитие компьютерной архитектуры неразрывно связано с эволюцией вычислительной техники — от первых ламповых машин до современных многоядерных процессоров.

1.2 Основные компоненты компьютера

Современный компьютер состоит из следующих ключевых компонентов:

- Центральный процессор (ЦП, CPU) — выполняет обработку данных и управление другими устройствами.
- Оперативная память (ОЗУ, RAM) — хранит данные и инструкции, которые используются в данный момент.
- Постоянная память (накопители: HDD, SSD) — используется для длительного хранения информации.
- Устройства ввода и вывода (клавиатура, мышь, монитор и т.д.).
- Системная плата (материнская плата), обеспечивающая связь всех компонентов.

Все эти элементы взаимодействуют друг с другом посредством системной шины — набора линий передачи данных, адресов и управляющих сигналов.

1.3 Архитектура фон Неймана

Классическая архитектура фон Неймана, предложенная в 1945 году, легла в основу большинства современных компьютеров. Её основные принципы:

- Программа и данные хранятся в одной и той же памяти.
- Последовательное выполнение команд.
- Единый путь передачи данных между процессором и памятью (узкое место — шина данных).

Эта модель оказалась достаточно универсальной и долгое время применялась без существенных изменений.

1.4 Гарвардская архитектура и современные тенденции

В отличие от архитектуры фон Неймана, гарвардская архитектура предполагает отдельные блоки памяти для программ и данных. Это позволяет ускорить выполнение инструкций, поскольку исключается конфликт между чтением кода и данных.

Современные компьютеры часто используют модифицированные архитектуры, сочетающие элементы обеих моделей. Также в последние годы активно развиваются многоядерные процессоры, параллельные вычисления, ускорители графики (GPU), использование кэш-памяти нескольких уровней (L1, L2, L3), виртуализация и энергосберегающие технологии.

1.5 Принципы организации современных компьютеров

Организация компьютера определяет, как аппаратные и программные компоненты взаимодействуют друг с другом. Основные принципы:

- Иерархия памяти: регистры, кэш, оперативная и внешняя память.
- Последовательная и параллельная обработка данных.
- Использование шин и контроллеров для связи компонентов.
- Управление прерываниями и синхронизация устройств.

Также большое значение имеет программная архитектура: операционная система, драйверы, системы ввода-вывода, интерфейсы и т.д.

Архитектура компьютера — это фундамент, на котором строится всё программное обеспечение и вычислительные процессы. Понимание её основ позволяет эффективно использовать возможности техники, оптимизировать программы и разрабатывать более производительные решения. Современные тенденции требуют от специалистов умения разбираться как в аппаратной, так и в программной части компьютерных систем.

2. Файл. Файловая система. Структура, функции, виды файловых систем.

2.1. Понятие файла

С развитием информационных технологий и операционных систем появилась необходимость эффективного хранения, обработки и организации данных. Основой этой системы являются файлы и файловые системы, которые позволяют пользователю взаимодействовать с данными и организовывать их в логическую иерархию. В данной работе рассмотрим понятие файла, назначение и типы файловых систем, их структуру и функции, а также классификацию наиболее популярных типов файловых систем.

Определение

Файл — это именованная область хранения данных, размещенная в долговременной памяти компьютера. Он представляет собой логическую единицу информации, которую операционная система и пользователь могут читать, изменять, сохранять или удалять.

Характеристики файла

Каждый файл обладает определенным набором характеристик:

- Имя файла — уникальное в пределах каталога.
- Расширение — часть имени, указывающая тип содержимого (.txt, .docx, .jpg и т.д.).
- Размер — объем занимаемого файла на диске.
- Атрибуты — свойства файла, такие как "только для чтения", "скрытый", "системный".
- Дата создания/изменения/доступа — метаданные о времени работы с файлом.

Типы файлов

Существует несколько видов файлов по содержимому и назначению:

- Текстовые файлы (.txt, .md, .html)
- Бинарные файлы (исполняемые .exe, .dll, .bin)
- Мультимедийные файлы (изображения, аудио, видео)
- Системные файлы (служебные файлы ОС)
- Файлы данных (таблицы, базы данных)

2.2. Файловая система

Определение

Файловая система (ФС) — это способ организации, хранения и управления данными на запоминающих устройствах, таких как жесткие диски, SSD, флешки и т.д. Она определяет, как именно файлы именуются, сохраняются, индексируются и извлекаются.

Назначение файловой системы

Основные задачи файловой системы:

- Эффективное хранение данных
- Быстрый доступ и поиск информации
- Защита целостности и безопасности данных
- Разделение дискового пространства между файлами
- Учет свободного и занятого пространства

2.3. Структура файловой системы

Логическая организация

Большинство файловых систем используют иерархическую (древовидную) структуру, включающую:

- Корневой каталог (root directory)
- Подкаталоги (subdirectories)
- Файлы как конечные узлы

Компоненты структуры

1. Кластеры/секторы — минимальные единицы хранения на диске.
2. Таблицы размещения файлов (FAT, MFT) — содержат информацию о положении и связности данных.
3. Каталоги — списки файлов, находящихся в одной директории.
4. Метаданные — информация о файле: размер, дата, права доступа и т.д.

2.4. Функции файловой системы

Управление файлами

Файловая система позволяет:

- Создавать, копировать, перемещать, переименовывать и удалять файлы
- Чтение и запись данных
- Защиту от потерь и ошибок

Управление пространством

Она также обеспечивает:

- Разбиение диска на кластеры
- Учет свободного и занятого пространства
- Дефрагментацию для ускорения доступа

Контроль доступа

Файловая система поддерживает права доступа, включая:

- Право на чтение
- Право на запись
- Право на выполнение
- Контроль с использованием владельцев и групп (например, в Unix-подобных ОС)

2.5. Виды файловых систем

FAT (File Allocation Table)

Одна из старейших файловых систем, разработанная Microsoft.

- FAT12, FAT16, FAT32
- Преимущества: простота, совместимость
- Недостатки: низкая безопасность, ограничение на размер файлов (4 ГБ в FAT32)

NTFS (New Technology File System)

Современная файловая система Windows.

- Поддержка шифрования, прав доступа, журналирования
- Преимущества: высокая безопасность, поддержка больших файлов
- Недостатки: ограниченная поддержка в других ОС (например, Linux требует дополнительных драйверов)

exFAT

- Разработана для флеш-накопителей

- Преимущества: поддержка больших файлов (до 16 эксабайт)
- Хорошая совместимость между Windows и macOS

ext (Extended File System)

Файловые системы семейства Linux:

- ext2, ext3, ext4
- Поддержка журналирования, больших файлов, прав доступа
- ext4 — самая популярная, стабильная и быстрая в своем семействе

HFS+ и APFS

Файловые системы для macOS:

- HFS+ — старая система Apple
- APFS — современная система с поддержкой шифрования, снапшотов, клонирования файлов

Другие

- ReFS — Resilient File System от Microsoft (для серверов)
- Btrfs — продвинутая файловая система для Linux
- ZFS — файловая система и менеджер томов с высокой надежностью

2.6. Сравнительная таблица популярных ФС

Файловая система	Поддержка больших файлов	Журналирование	Совместимость	Использование
FAT32	Нет (>4 ГБ)	Нет	Windows, macOS	USB-флешки
NTFS	Да	Да	Windows	HDD, SSD
exFAT	Да	Нет	Windows, macOS	Flash-накопители
ext4	Да	Да	Linux	Linux-диски
APFS	Да	Да	macOS	Apple-устройств

Файловая система — ключевой компонент операционной системы, обеспечивающий логическую организацию, хранение и управление данными. Выбор файловой системы зависит от задач пользователя, операционной системы и типа носителя. Понимание принципов работы ФС важно не только для системных администраторов и разработчиков, но и для всех пользователей, работающих с цифровыми устройствами.

3. Написать программу для расчета по двум формулам

3.1. Создание листинга

```
#include <iostream>

#include <cmath>    // Для sqrt

#include <iomanip>    // Для setprecision

double calculateZ1(double a) {

    double part1 = (a + 2.0) / std::sqrt(2.0 * a);

    double part2 = a / std::sqrt(2.0 * a + 2.0);

    double part3 = 2.0 / (a - std::sqrt(2.0 * a));

    double multiplier = (std::sqrt(a) - std::sqrt(2.0)) / (a + 2.0);

    return (part1 - part2 + part3) * multiplier;

}

double calculateZ2(double a) {

    return 1.0 / (std::sqrt(a) - std::sqrt(2.0));

}

int main() {

    double a;

    std::cout << "Введите значение a: ";

    std::cin >> a;

    if (a <= 0.0 || a == std::sqrt(2.0)) {

        std::cerr << "Ошибка: недопустимое значение 'a'. Деление на 0 или
```

корень из отрицательного.\n";

```
    return 1;
```

```
}
```

```
double z1 = calculateZ1(a);
```

```
double z2 = calculateZ2(a);
```

```
std::cout << std::fixed << std::setprecision(6);
```

```
std::cout << "z1 = " << z1 << std::endl;
```

```
std::cout << "z2 = " << z2 << std::endl;
```

```
return 0;
```

```
}
```

3.2. Результат программы

Ввод:

Введите значение a: 4

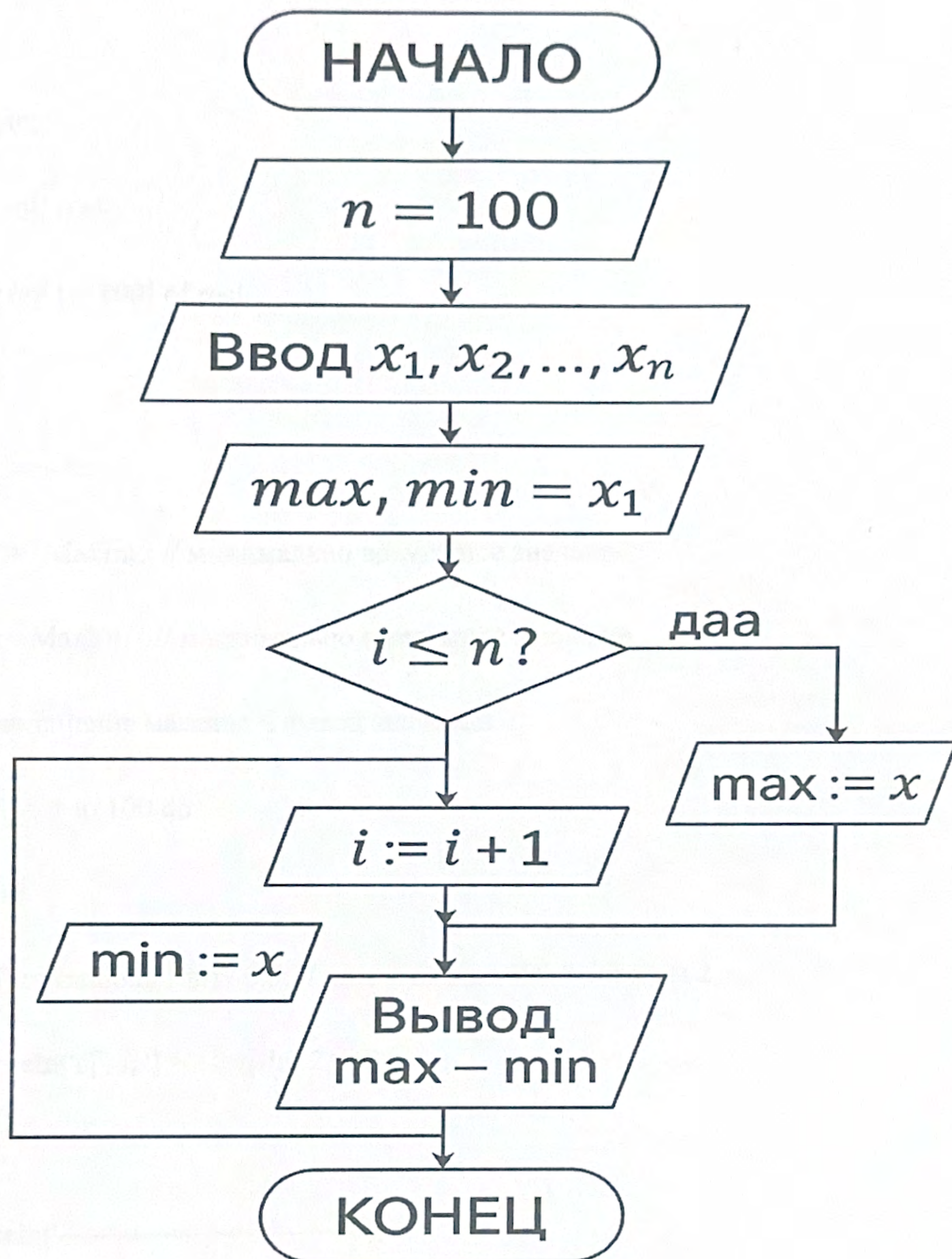
Вывод:

z1 = 0.282071

z2 = 1.707107

4. Пусть дано 100 вещественных чисел. Вычислите разность между максимальным и минимальным числами.

4.1. Создание блок-схемы



4.2. Листинг программы

```
program FindMaxMinDifference;

uses crt;

var

    i: byte;

    ma, mi: real;

    a: array [1..100] of real;

begin

    randomize;

    ma := -MaxInt; // минимально возможное значение

    mi := MaxInt;  // максимально возможное значение

    // Заполнение массива и вывод значений

    for i := 1 to 100 do

        begin

            a[i] := random(100) / 0.5; // число от 0 до 199.8 с шагом 2

            writeln('a[', i, '] => ', a[i]:0:2);

        end;

    writeln('-----');
```

```

// Поиск максимума и минимума

for i := 1 to 100 do

begin

    if a[i] > ma then

        ma := a[i];

    if a[i] < mi then

        mi := a[i];

end;

writeln('Максимум: ', ma:0:2);

writeln('Минимум: ', mi:0:2);

writeln('Разность (ma - mi): ', (ma - mi):0:2);

end.

```

4.3. Результат программы

a[1] => 62.00

a[2] => 84.00

...

a[100] => 18.00

Максимум: 198.00

Минимум: 0.00

Разность (ma - mi): 198.00

Заключение

В ходе выполнения работы я познакомился с важными темами, связанными с работой современных компьютеров, файловыми системами и обработкой данных. Сначала я узнал, как устроены современные компьютеры. Их работа основана на архитектуре фон Неймана, где чётко разделены части: память, процессор и устройства ввода-вывода. Компьютер выполняет команды поочерёдно, и данные, и программы хранятся в одной и той же памяти. Эти принципы используются во всех современных ПК, ноутбуках и даже смартфонах. Затем я изучил, что такое файл и файловая система. Файл — это просто хранилище данных с именем, а файловая система — это способ, с помощью которого операционная система организует и управляет файлами на диске. Существует несколько типов файловых систем, например NTFS, FAT32, ext4. Они отличаются по возможностям, скорости и совместимости. Именно файловая система отвечает за то, чтобы мы могли удобно сохранять, открывать и удалять файлы. Так же этой работе мы решили две практические задачи на программирование и немного обсудили результаты. Вычисление по двум формулам Мы написали программу, которая получает на вход число a и рассчитывает по нему два выражения. Работа с массивом из 100 вещественных чисел Во второй части нам дали 100 чисел и попросили найти разницу между максимальным и минимальным. Мы просто просматриваем все элементы, запоминая текущие минимум и максимум, а затем вычитаем их. Это стандартная операция при анализе данных: позволяет оценить разброс — насколько велика разница между самыми большими и самыми маленькими значениями. В целом, работа помогла мне лучше понять, как устроен компьютер, как он работает с файлами, и как с помощью программ можно выполнять полезные вычисления. Всё это — важные базовые знания,

которые пригодятся мне в дальнейшем обучении и в профессии.