Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Системное программное обеспечение вычислительных машин

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему

«Низкоуровневый редактор блочного устройства уровня секторов(ncurses)»

БГУИР КП 1–40 02 01 01 110 ПЗ

Студент: В. А. Гулевич

Руководитель: А. О. Игнатович

МИНСК 2024

Учреждение образования

«Белорусский государственный университет информатики

и радиоэлектроники»

Факультет компьютерных систем и сетей

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

Б.В.Никульшин

(подпись)

« » 2024 г.

ЗАДАНИЕ

по курсовому проектированию

Студенту *Гулевич Владиславу Александровичу*

1. Тема проекта  *Низкоуровневый редактор блочного устройства уровня секторов(ncurses)*

2. Срок сдачи студентом законченного проекта *25 мая 2024 г*.

3. Исходные данные к проекту *Язык программирования – C*

4. Содержание расчетно-пояснительной записки (перечень вопросов, которые подлежат разработке)

*Введение. 1. Обзор литературы. 2. Системное проектирование.   
3. Функциональное проектирование. 4. Разработка программных модулей.   
5. Программа и методика испытаний. 6. Руководство пользователя. Заключение. Список использованных источников*

5. Перечень графического материала (с точным обозначением обязательных чертежей и графиков) *1. Схема структурная. 2. Диаграмма классов*

6. Консультант по проекту *Игнатович А.О.*

7. Дата выдачи задания *20 февраля 2024 г.*

8. Календарный график работы над проектом на весь период проектирования (с обозначением сроков выполнения и трудоемкости отдельных этапов):

*разделы 1,2 к 15 марта 2024 г. – 20 %;*

*разделы 3,4 к 15 апреля 2024 г. – 30 %;*

*разделы 5,6,7 к 10 мая 2024 г. – 30 %;*

*оформление пояснительной записки и графического материала к 10 мая 2024 г. 20 %*

*Защита курсового проекта с 28 мая 2024 г. по 10 июня 2024 г.*

РУКОВОДИТЕЛЬ *Игнатович А.О.*

(подпись)

Задание принял к исполнению *В. А. Гулевич*

(дата и подпись студента)

**СОДЕРЖАНИЕ**

**ВВЕДЕНИЕ**

Целью данного курсового проекта является разработка низкоуровневого редактора блочного устройства уровня секторов с использованием библиотеки ncurses. В современных операционных системах(ОС) блочные устройства играют важную роль в работе с данными, так как они предоставляют доступ к физическим секторам на диске. Редактор блочного устройства предоставляет возможность прямого доступа и редактирования данных на физическом уровне блочных устройств, основываясь на работе с отдельными секторами данных.

Такой редактор позволяет осуществлять чтение и запись данных на уровне секторов, минуя файловую систему и обрабатывая данные непосредственно на физическом носителе, таком как жесткий диск или твердотельный накопитель. Это означает, что данные могут быть прочитаны или записаны без учета структуры файловой системы или файловых атрибутов.

Данный тип редактора обычно используется системными администраторами, разработчиками операционных систем и специалистами по восстановлению данных для различных целей, включая работу с поврежденными файловыми системами, исследование и анализ данных и разработку и отладку драйверов устройств.

К очевидным минусам можно отнести сложность работы с устройством и потенциальные риски, так как неправильное редактирование или действия с данными на физическом уровне могут привести к потере данных или повреждению устройств. Так же из-за того что редактор ориентирован на работу с секторами данных напрямую, он может быть ограничен в функциональности, поскольку не предоставляет полного набора возможностей, доступных в более высокоуровневых инструментах, таких как файловые менеджеры или инструменты анализа данных.

Для реализации редактора блочного устройства мы выбрали библиотеку ncurses. Ncurses предоставляет набор функций и макросов, которые позволяют программистам управлять выводом текста, обрабатывать клавиатурные и мышиные события, а также создавать интерактивные элементы интерфейса, такие как кнопки, текстовые поля, меню, окна и прогресс-индикаторы. Библиотека ncurses обеспечивает переносимость кода между различными операционными системами и терминалами, что позволяет создавать кросс-платформенные текстовые интерфейсы. Одной из ключевых особенностей ncurses является возможность работы с текстом в режиме символов, а не только в режиме строк, что позволяет создавать более гибкие и интерактивные интерфейсы. Библиотека также обладает возможностями цветового оформления и управления курсором, что позволяет создавать более привлекательные и информативные пользовательские интерфейсы.

Разработка низкоуровневого редактора блочного устройства требует понимания структуры и организации данных на физическом носителе, а также знания основных операций чтения, записи и модификации данных на уровне секторов. Будут реализованы основные функции редактирования, такие как чтение и запись секторов, перемещение по диску, поиск и замена данных.

В процессе выполнения курсового проекта мы надеемся углубить наши знания в области низкоуровневого программирования, а также приобрести практические навыки разработки текстового пользовательского интерфейса с использованием библиотеки ncurses. Мы также рассчитываем на получение полезного опыта работы с блочными устройствами и анализа данных на низком уровне.

**1 ОБЗОР ЛИТЕРАТУРЫ**

**1.1 Анализ существующих аналогов**

Тема курсового проекта была выбрана в первую очередь для углубления знаний по языку C а также получения знаний в взаимодействии с данными, поэтому моей целью не является разработать конкурентоспособный продукт. Тем не менее, чтобы создать корректно работающее приложение, нужно иметь представление о существующих аналогах, об их недостатках, преимуществах и реализованных внутри функциях.

* + 1. **HxD**

HxD – это редактор шестнадцатеричных данных для ОС Windows. Он предоставляет возможность просмотра и редактирования данных на низком уровне, включая работу с блочными устройствами уровня секторов. В программе реализованы алгоритмы чтения и записи, поиска, сравнения файлов, шифрования и дешифрования и алгоритм отображения данных. Для разработки был использован WinAPI для взаимодействия с ОС Windows, Microsoft Foundation Classes для разработки пользовательского интерфейса и иные библиотеки. Интерфейс HxD изображен на рисунке 1.1.

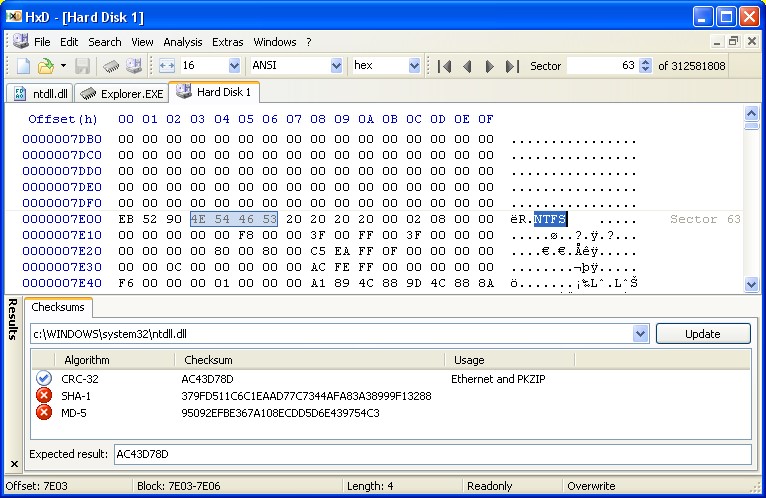


Рисунок 1.1 — Пользовательский интерфейс программы HxD

HxD разработан Майелем Герцем в 2003 году. Приложение написано на языке C++ для ОС Windows XP, 2003, Vista, 7, 8 или 10. Последняя версия была выпущена в 2021 году.

* + 1. **wxHexEditor**

wxHexEditor – это редактор шестнадцатеричных данных с графическим пользовательским интерфейсом. Имеет схожий функционал с HxD, однако имеет несколько значительных отличий. wxHexEditor использует 64–битный файловый дескриптор, что позволяет ему поддерживать файлы размером до 2^64 байт. Так же он хорошо оптимизирован для эффективной обработки больших объемов данных. Хорошей оптимизации помогли добиться технологии ленивой загрузки данных, буферизации данных и кэширование отображения. Более того, в написании приложения использовался фреймворк wxWidgets, который обеспечивает кросс–платформенную поддержку. Интерфейс wxHexEditor изображен на рисунке 1.2.

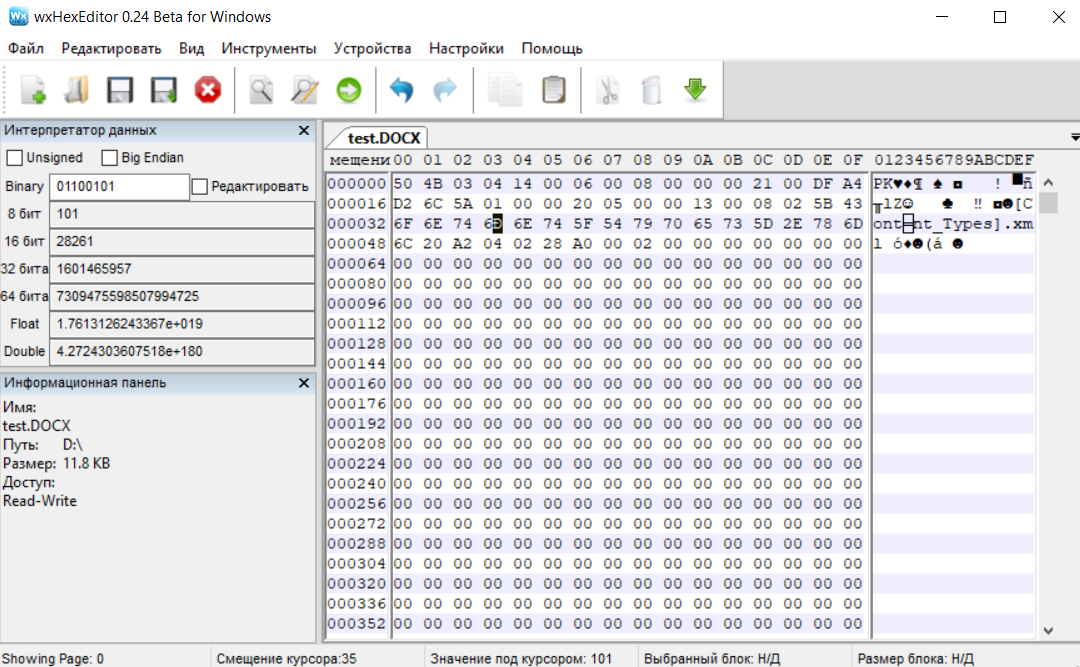


Рисунок 1.2 — Пользовательский интерфейс программы wxHexEditor

wxHexEditor был разработан Андреасом Шретером в 2006 году. Приложение написано на C++ и поддерживает ОС Windows, Linux и MacOS. Последняя версия была выпущена в 2012 году, и сейчас приложение не поддерживается.

* + 1. **DiskGenius**

DiskGenius отличается от двух предыдущих аналогов. Он является мощным программным обеспечением для управления дисками и восстановления данных. Он предоставляет широкий спектр функций для работы с жесткими дисками, разделами и файловыми системами. DiskGenius включает в себя шестнадцатеричный редактор, имеет низкоуровневый доступ к блочным устройствам и имеет алгоритмы сканирования и восстановления данных. Интерфейс DiskGenius изображен на рисунке 1.3.

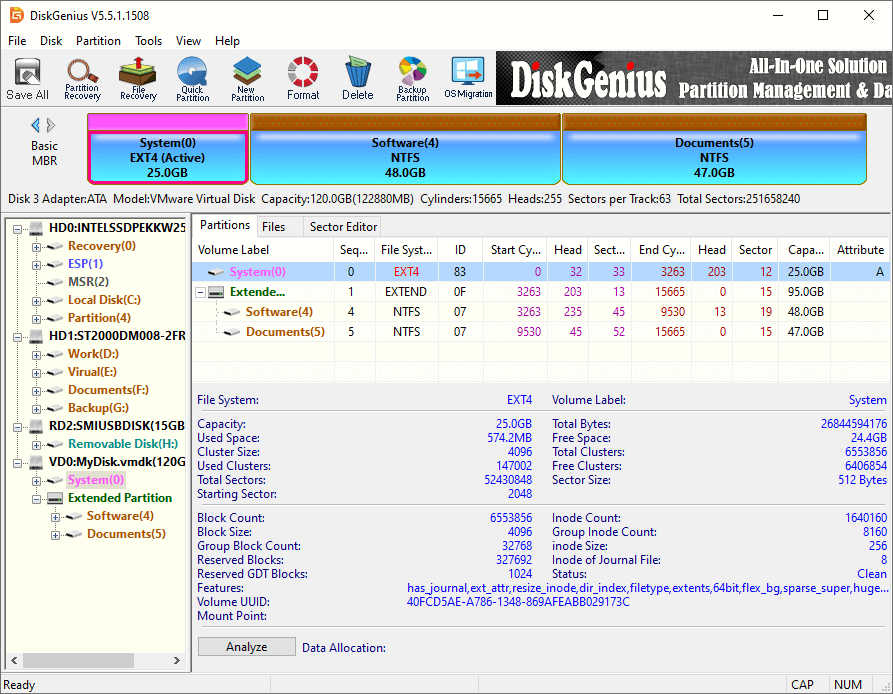


Рисунок 1.3 — Пользовательский интерфейс программы DiskGenius

DiskGenius был разработан Eassos Ltd. Он был выпущен в 2005 году и с тех пор продолжает обновляться. Приложение написано на C++ и поддерживает ОС Windows.

**1.2 Постановка задачи**

После рассмотрения аналогов можно сказать, что все они обладают большим количеством функций, которые невозможно реализовать в курсовом проекте за данный период времени. Поэтому были выбраны несколько ключевых возможностей, которые будут выполнены в рамках данного курсового проекта:

- Программа должна иметь удобный и понятный пользовательский интерфейс с необходимыми кнопками выбора действия.

- В программе должны быть основные функции низкоуровневого редактора блочных устройств уровня секторов, такие как чтение секторов и запись в них, редактирование данных, поиск, копирование и перемещение, изменение размеров секторов и создание и удаление разжелов.

В качестве языка был выбран язык программирования C. Основные достоинства этого языка для реализации этого проекта – это близость к аппаратуре, так как C является низкоуровневым языком программирования, что обеспечивает близкое соответствие кода программы аппаратуре компьютера. Другим достоинством языка является эффективность и производительность, что важно для работы с большими объемами данных и выполнения операций непосредственно на уровне секторов. Также С дает доступ к системным ресурсам и позволяет работать с памятью.

Для реализации графического пользовательского интерфейса будет использована библиотека ncurses. Она предоставляет набор функций для создания текстового пользовательского интерфейса в терминале. Ncurses позволяет управлять экраном, обрабатывать ввод с клавиатуры и события мыши, создавать окна и панели, что позволяет организовать различные части интерфейса и управлять их расположением.

Данный список средств позволяет реализовать все задачи, поставленные для курсового проекта.

**2 СИСТЕМНОЕ ПРОЕКТИРОВАНИЕ**

После определения требований к функционалу разрабатываемого приложения его следует разбить на функциональные блоки. Такой подход упростит понимание проекта, позволит устранить проблемы в архитектуре, обеспечит гибкость и масштабируемость программного продукта в будущем путем добавления новых блоков.

**2.1**

**3 ФУНКЦИОНАЛЬНОЕ ПРОЕКТИРОВАНИЕ**

В данном разделе описывается функционирование и структура разрабатываемого приложения.

**4 РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ**

**5 ТЕСТИРОВАНИЕ ПРОГРАММЫ**

**6 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ**

**ЗАКЛЮЧЕНИЕ**

В результате работы над данным курсовым проектом было разработано работоспособное приложение со своим набором функций и графическим интерфейсом. Данный курсовой проект был разработан в соответствии с поставленными задачами, весь функционал был реализован в полном объеме.

Для создания курсового проекта была подробна исследована технология работы с блочными устройствами. В ходе разработки были углублены знания языка программирования С. Также были успешно использованы такие преимущества языка C, как близость к аппаратуре, эффективность и доступ к системным ресурсам. Как и ожидалось, библиотека ncurses значительно упростила разработку текстового пользовательского интерфейса для редактора блочных устройств.

Работа была разделена на такие этапы, как анализ существующих аналогов, литературных источников, постановка требований к проектируемому программному продукту, системное и функциональное проектирование, конструирование программного продукта, разработка программных модулей и тестирование проекта. После последовательного выполнения вышеперечисленных этапов разработки было получено исправно работающее приложение.

В дальнейшем планируется усовершенствование текущего функционала приложения, путем улучшения графического интерфейса, добавления новых функций и модулей, а также добавления возможности работы под разными системами.

**СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Вычислительные машины, системы и сети: дипломное проектирование (методическое пособие) [Электронный ресурс]. – Минск БГУИР 2019. – Режим доступа: <https://www.bsuir.by/m/12_100229_1_136308.pdf>

2. Habr [Электронный ресурс]. – Руководство по ncurses. – Режим доступа: <https://habr.com/ru/articles/778040/> – Дата доступа: 29.03.2024

3. Жешке Рекс. Толковый словарь стандарта языка Си / Пер. с англ.– М.: Мир, 1992.–687с.