Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет   
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

на тему

УТИЛИТА ПОИСКОВ ОДИНАКОВЫХ ФАЙЛОВ   
(аналог fdupes, но с фильтрацией по именам и типам)

БГУИР КР 1-40 02 01 306 ПЗ

Студент: Григорик И. А.

Руководитель: Глоба А. А.

Минск 2022

Учреждение образования

«Белорусский государственный университет информатики

и радиоэлектроники»

Факультет компьютерных систем и сетей

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

Б.В. Никульшин

(подпись)

« » 2022 г.

ЗАДАНИЕ

по курсовому проектированию

Студенту *Григорик Ивану Александровичу*

1. Тема проекта  *Утилита поиска одинаковых файлов (аналог fdupes, но с фильтрацией по именам и типам)*

2. Срок сдачи студентом законченного проекта *25 мая 2022 г*.

3. Исходные данные к проекту *Язык программирования – C / С++*

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

*Введение. 1. Обзор литературы. 2. Структурное проектирование.   
3. Функциональное проектирование. 4. Разработка программных модулей. 5.Тестирование программы и ркуоводство пользователя. Список использованных источников*

5. Перечень графического материала (с точным обозначением обязательных чертежей и графиков)

6. Дата выдачи задания *20 февраля 2022 г.*

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

*разделы 1,2 к 15 марта 2021 г. – 20 %;*

*разделы 3,4 к 15 апреля 2021 г. – 30 %;*

*разделы 5,6,7 к 15 мая 2021 г. – 30 %;*

*оформление пояснительной записки и графического материала к 25 мая 2021 г. 20 %*

*Защита курсового проекта с ?? июня 2022 г. по ?? июня 2022 г.*

РУКОВОДИТЕЛЬ *А. А. Глоба*

(подпись)

Задание принял к исполнению *И. А. Григорик*

(дата и подпись студента)

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 5](#_Toc100447961)

[1. ОБЗОР ЛИТЕРАТУРЫ 6](#_Toc100447962)

[2. СИСТЕМНОЕ ПРОЕКТИРОВАНИЕ 8](#_Toc100447963)

[ЛИТЕРАТУРА 9](#_Toc100447964)

# ВВЕДЕНИЕ

В текущее время у каждого пользователя компьютера, вне зависимости от его опыта работы с компьютером, рано или поздно появляются одинаковые файлы. По различным причинам пользователь может не замечать их, или просто игнорировать, т.к. искать вручную все дублирующиеся файлы не так уж и просто. Именно поэтому используются специальные утилиты для автоматического поиска одинаковых файлов.

К примеру, fdupes – утилита, написанная Андрианом Лопесом, может искать одинаковые файлы из любого каталога в системе. Для этого используется получение хеша файла из его дескриптора. Следует пояснить, что такое хеш файла и файловый дескриптор:

* Хеш файла – это уникальный идентификатор файла, который высчитывается системой посредством определённых преобразований хранящейся в файле информации.
* Дескриптор файла – это целое неотрицательное число, с помощью которого процесс может обращаться к потоку ввода-вывода. Дескриптор может быть связан с файлом, сокетом или каталогом.

С помощью данных средств можно сверять файлы по содержанию, что и является целью поиска одинаковых файлов.

Данный курсовой проект представляет собой такую утилиту командной строки, подобную уже известной утилите fdupes, которая будет искать файлы с одинаковым содержимым, основываясь на хеше файла. Так же данная утилита будет сравнивать расширения файлов и их имена, помимо содержимого. Так же, с помощью некоторых управляющих флагов, утилита сможет производить действия над данными файлами, помимо простого поиска.

В рамках данной курсовой работы необходимо ознакомится с библиотекой openSSL/md5.h, которая используется для получения хеша файлов, или же с командой MD5, которые используются для получения MD5 хеша, с помощью которого файлы и будут проверяться на идентичность. В процессе разработки следует углубить знания по языку C / C++, а также осуществить взаимодействие пользовательского приложения с системой. В конце следует протестировать приложение и провести эксперименты на нескольких устройствах.

OpenSSL – полноценная криптографическая библиотека, с открытым исходным кодом, которая используется во многих проектах для хеширования MD5, MD2, SHA. Библиотека написана Эриком Янгом и Тимом Хадсоном, и получила популярность благодаря расширениям SSL/TLS, которые используются в веб-протоколе HTTPS.

# ОБЗОР ЛИТЕРАТУРЫ

Следует начать с определения утилиты.

**Утилита** – это некая вспомогательная компьютерная программа в составе программного обеспечения, которая используется для выполнения типовых задач, связанной с работой оборудования или ОС. В данном случае утилита используется для облегчения пользования компьютером.

Данная утилита основывается на получении хеша файла из его дескриптора, определения которых давались выше. Процедура получения хеша называется **хешированием**.

Хеширование может проводиться по разным алгоритмам. Основной смысл хеширования заключается в безопасности и надёжности, возможности сжимать любые куски информации в короткий стандарт сообщений, являющихся уникальными для каждой доли информации. Даже если один байт информации будет изменён – хеш так же поменяется.

Таким образом не надо хранить всё содержимое файла или каждый раз его открывать, чтобы сверить его информацию с другим файлом. Достаточно будет просто сгенерировать его хеш и запомнить его.

Недостатком хеширования является неизбежность коллизии.

**Коллизия** – это равенство значений хеш-функций на двух различных кусках информации. В данном случае это означает, что если функция сгенерирует одинаковый хеш для двух разных файлов, то в системе они будут считаться за одинаковые. Для решения вопроса коллизий создаются современные хеш-функции, в которых шанс появления коллизий стремится к минимуму.

Так же стоит пояснить, что длина строки зависит от конкретной хеш-функции, но одна функция не может сгенерировать две строки разной длины.

На данный момент популярны следующие хеш-функции:

* **SHA256** – одна из наиболее устойчивых к коллизиям функция. Недостаток: по сравнению с другими, имеет довольно большое время выполнения и большая длина строки (256 байт).
* **RIPEMD160** – так же устойчивая к коллизиям функция, которая, к тому же, имеет длину строки почти в два раза меньше (160 байт), чем SHA256. Время выполнения примерно такое же, как у SHA256.
* **MD5** – самая быстрая криптографическая хеш-функция из широко используемых, к тому же имеет наименьший размер хеша (128 байт). Недостаток: небезопасна. Легко подвергается коллизиям, поэтому не стала использоваться в таких проектах, как криптовалютные кошельки.

Так же хеширование используется для сокрытия данных, так как хеширование одностороннее (т.е. нельзя преобразовать хеш в первоначальные данные). Получить первоначальные данные можно только сгенерировав такую же строку, или создать коллизию, которая приведёт конвертации другой строки

В данном случае сокрытие не требуется, и могут допускаться коллизии, поэтому может использоваться функция хеширования MD5.

Теперь же о файловых дескрипторах.

Файловый дескриптор в данном случае используется для отображения файла на память. Данный метод является эффективным, ибо помогает разгрузить систему, и вообще не использовать физическую память, чем помогает снизить нагрузку на диск для нескольких программ, обращающимся к одному и тому же файлу.

# СИСТЕМНОЕ ПРОЕКТИРОВАНИЕ

# ЛИТЕРАТУРА

1. Лафоре, Р. Объектно-ориентированное программирование в C++ / Р. Лафоре. – СПб. : Питер, 2004.
2. Буч, Г. Язык UML. Руководство пользователя / Г. Буч, Д. Рамбо, А. Джекобсон; пер. с англ. – СПб. : ДМК, 2004.
3. Лав Р. Системное программирование на Linux/ 2-е издание 2014.
4. Керниган Б. Язык программирования С/ 4-е издание М.:Питер, 2004. – 923 с.
5. Рочкинд М. Программирование для UNIX, 2-е изд. СПб, БХВ-Петербург, 2005.
6. Калле Р. Грокаем технологию биткоин / Р. Калле – СПб. : Питер, 2020.

Контрльный лист (то что помещается 40 строк на лист) 1111111111111111111111111111111111111111111111111111111111111111111122222222222222222222222222222222222222222222222222222222223333333333333333333333333333333333333333333333333333333333333333333444444444444444444444444444444444444444444444444444444444444444445555555555555555555555555555555555555555555555555555555555555555555566666666666666666666666666666666666666666666666666666666666666666667777777777777777777777777777777777777777777777777777777777777777777777778888888888888888888888888888888888888888888888888888888888899999999999999999999999999999999999999999999999999999999999999999999000000000000000000000000000000000000000000000000000000000000000000011111111111111111111111111111111111111111111111111111111111111112222222222222222222222222222222222222222222222222222222222222222222223333333333333333333333333333333333333333333333333333333333333334444444444444444444444444444444444444444444444444444444444444444444455555555555555555555555555555555555555555555555555555555555555556666666666666666666666666666666666666666666666666666666666666666666677777777777777777777777777777777777777777777777777777777777777777788888888888888888888888888888888888888888888888888888888888888888889999999999999999999999999999999999999999999999999999999999999999990000000000000000000000000000000000000000000000000000000000000000001111111111111111111111111111111111111111111111111111111111111111222222222222222222222222222222222222222222222222222222222222222222233333333333333333333333333333333333333333333333333333333333333333333444444444444444444444444444444444444444444444444444444444444444444555555555555555555555555555555555555555555555555555555555555555566666666666666666666666666666666666666666666666666666666666666666677777777777777777777777777777777777777777777777777777777777777778888888888888888888888888888888888888888888888888888888888888888888899999999999999999999999999999999999999999999999999999999999999999990000000000000000000000000000000000000000000000000000000000001111111111111111111111111111111111111111111111111111111111111111111112222222222222222222222222222222222222222222222222222222222222222222333333333333333333333333333333333333333333333333333333333333333333334444444444444444444444444444444444444444444444444444444444444444444555555555555555555555555555555555555555555555555555555555555555666666666666666666666666666666666666666666666666666666666666667777777777777777777777777777777777777777777777777777777777777777777777888888888888888888888888888888888888888888888888888888888888888888899999999999999999999999999999999999999999999999999999999999999990000000000000000000000000000000000000000000000000000000000000000000