Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Кафедра интеллектуальных информационных технологий

Отчет по лабораторной работе №4

по курсу «Операционные системы»

на тему: «Изучение организации файловой системы»

|  |  |
| --- | --- |
| Выполнил студент  группы 521702: | Македон Е.А. |
| Проверил: | Колб Д.Г. |

Минск 2017

**Цель:**

Изучить подходы к организации файловой системы.

**Описание задачи:**

Реализовать модель файловой системы и реализовать базовые команды просмотра файлов на диске – создание, удаление, копирование, перемещение, запись в файл, чтение файла. Обеспечить создание дампа файловой системы таким образом, чтобы можно было просмотреть структуру файловой системы и содержимое файлов.

**Описание методов, которые был использованы для решения задачи:**

Данная файловая система была разработана и написана на ОС LINUX при помощи языка программирования c++. Основными используемыми библиотеками являются #include <regex> - использования регулярных выражения на валидации входных значений и парсинга строки на определенные ее части и получения значения, библиотеки #include <sys/io.h> и #include <fcntl.h> - для записи\считывания байт в\из файла, а также использовались стандартные библиотеки языка c++.

**В рамках лабораторной работы были реализованы следующие функции:**

1) init – инициализация файловой системы.

2) load – загрузка файловой системы.

3) create – создание файла.

4) remove – удаление файла.

5) copy – копирование файла.

6) move – переименования файла.

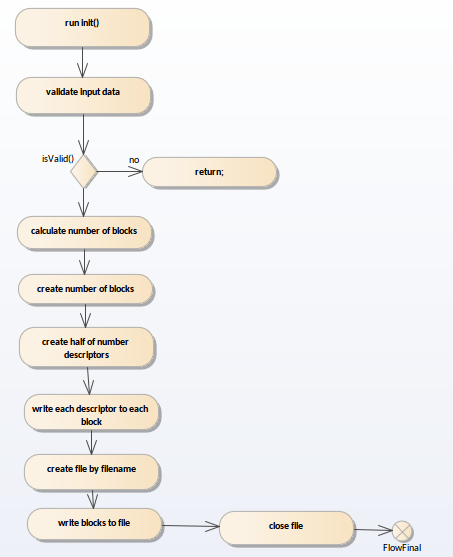
7) write – запись в файл.

8) read – чтение файла.

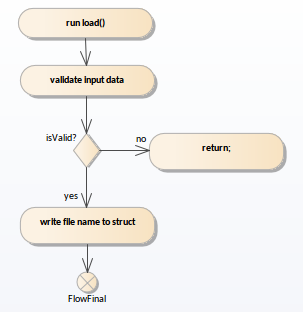
9) exit – закрытие файловой системы.

10) ls – просмотр файлов.

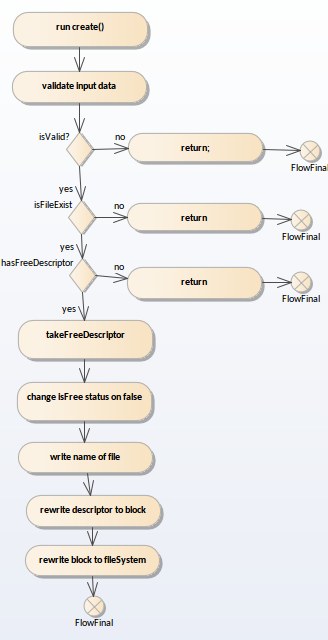
**init()**



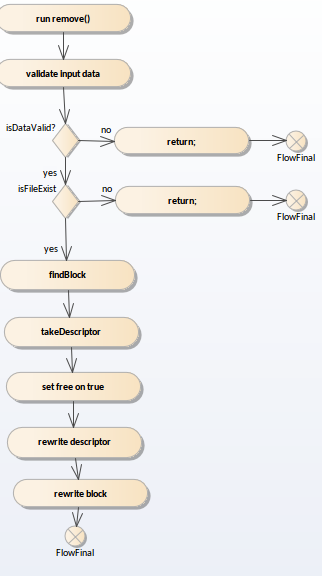
**load()**



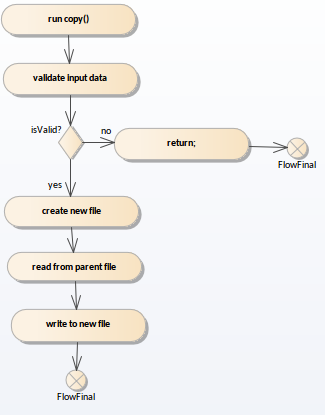
**create()**



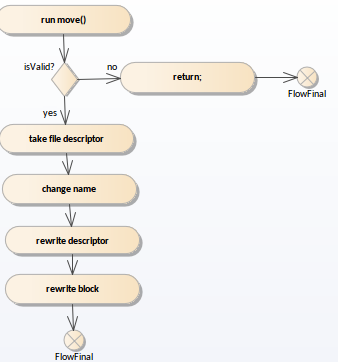
**remove()**



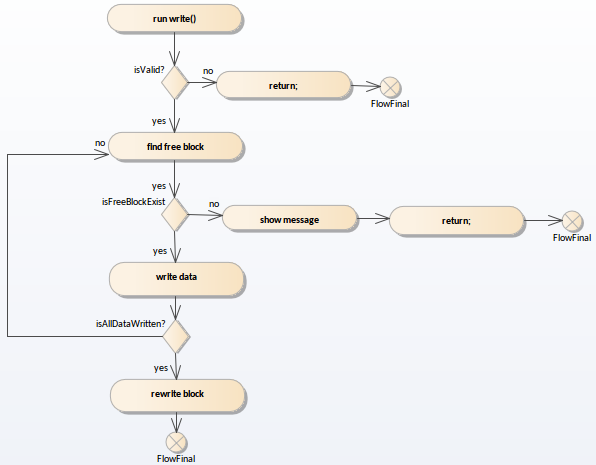
**copy()**



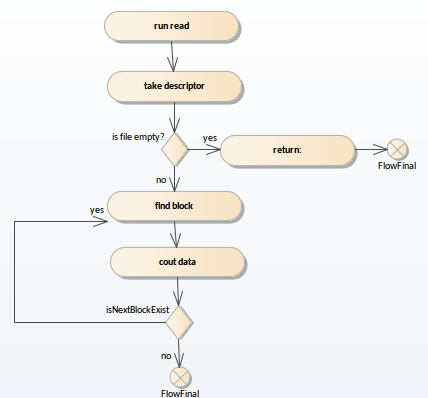
**move()**



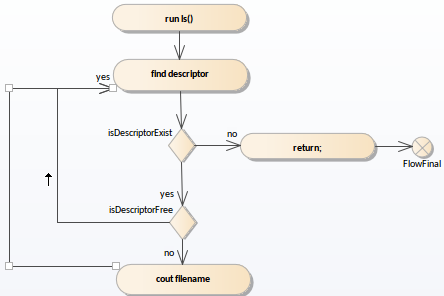
**write()**



**read()**



**ls()**



**Вывод:**В рамках лабораторной работы была реализована модель одноуровневой файловой системы, с физической организацией файла - связанный список блоков. Достоинства модели: простота реализации. Недостатки: высокое время доступа к произвольному файлу.