

Министерство образования Республики Беларусь

**Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»**

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ

Кафедра интеллектуальных информационных технологий

**Отчёт по лабораторной работе №5
по курсу «ОС» на тему:
«Изучение организации файловой системы»**

Выполнили студент группы 121702:	Кимстач Даниил Борисович Заломово Роман Андреевич
Проверил:	Цирук Владимир Александрович

Минск 08.12.2022

1. Цель и задача	3
2. Используемые методы	4
2.1. Общая архитектура.....	4
2.2. Особенности реализации	4
3. Тесты.....	5
3.1. Результаты исследования.....	6
4. Ответы на контрольные вопросы:	8
4.1. Файл - именованная область данных на носителе информации, используемая как базовый объект взаимодействия с данными в операционных системах.	8
4.2. Обычный, каталог, специальный	8
4.3. Файловая система – порядок, определяющий способ хранения и организации данных на носителях информации.....	8
4.4. Атрибут файла – метаданные, описывающие файл. Примером атрибута в файловой системе Linux может служить права доступа.....	8
4.5. Логическая структура файла – метод организации файла. Логическая запись – наименьший элемент данных, с которым можно оперировать при обмене с внешним устройством.	8
4.6. Физическая организация файла – описания правил расположения файла на физических накопителях. Физическая запись – наименьшая единица данных, которой внешнее устройство обменивается с оперативной памятью	8
4.7. Символьное имя файла – строка символов, определяющая файл в некотором пространстве имён файловой системы.	8
4.8. Проецируемый в память файл – файл, которому или части которому ставится в соответствие некоторый диапазон адресов оперативной памяти. При этом чтение такого файла есть чтение из поставленного ему в соответствие участка памяти.	8
5. Вывод	9

1. Цель и задача

Изучить подходы к организации файловой системы.

Вариант 1

Задача: реализовать одноуровневую файловую систему, с физической организацией файла
- непрерывное размещение

2. Использованные методы

2.1. Общая архитектура

Файловая система — порядок, определяющий способ организации, хранения и именования данных на носителях информации в компьютерах, а также в другом электронном оборудовании: цифровых фотоаппаратах, мобильных телефонах и т. п.

Одноуровневая файловая система с физической организацией файла – непрерывное размещение блоков представляет собой один из простейших видов файловых систем. Смысл хранения файлов в такой системе заключается в том, что файл представляет собой всего лишь последовательность блоков, размещённых непрерывно. Например, файлу размером 16кб с размером блока 4кб будет выделено 4 блока. Каждому файлу поставлен в соответствие индекс его первого блока и количество занимаемых им блоков. Такая файловая система просто реализуется, у нас есть возможность быстро читать файлы, но главный минус заключается в том, что диск будет сильно фрагментирован при после нескольких циклов создания/удаления файлов. Плюсом к этому перед созданием файла необходимо заранее знать его размер.

Подобная файловая система на данный момент считается устаревшей и используется только на CD-ROM дисках.

2.2. Особенности реализации

Наша реализация файловой системы предполагает наличие одного диска, размер которого требуется указать при инициализации файловой системы. Сам диск представляет собой набор (массив) блоков, размер которого определяется исходя из размера диска, который мы указали. Блок содержит текстовые данные и переменную, указывающую на его пустоту/непустоту. Файл представляет собой набор, состоящий из его имени, индекса его первого блока и количества блоков, им занимаемых.

3. Тесты

Здесь представлен тест, охватывающий различные ситуации, которые могут произойти при работе с файловой системой: как от правильных операций с базовыми командами, так и от неправильных – создание файлов с одинаковым именем, создание файла при недостаточном

месте на диске(учитывая фрагментацию) и т.д. Был написан отдельный класс для проведения тестирования, т.к. почти каждый метод в каждом классе возвращает код успеха/неуспеха.

```
int main(){
    FileSystem file_system(16384);
    Tester::print_test(Tester::test_function(file_system.fcreate("test1", 4096, "Hello, world!"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fcreate("test2", 4096, "Ogryzok i oguzok"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fcreate("test2", 4096, "Ogryzok i oguzok"), FILE_ALREADY_EXISTS));
    Tester::print_test(Tester::test_function(file_system.fcreate("test3", 4096, "Sample text"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fcreate("test4", 4096, "Bla-bla"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.flookup(), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fread("test1"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fread("test2"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fcreate("test5", 4096, "Test file no 5"), NOT_ENOUGH_SPACE));
    Tester::print_test(Tester::test_function(file_system.fread("test5"), FILE_NOT_FOUND));
    Tester::print_test(Tester::test_function(file_system.fdelete("test2"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fdelete("test4"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fread("test2"), FILE_NOT_FOUND));
    Tester::print_test(Tester::test_function(file_system.fread("test4"), FILE_NOT_FOUND));
    Tester::print_test(Tester::test_function(file_system.fcreate("test6", 8192, "Bla-bla that is file test6"), NOT_ENOUGH_SPACE));
    Tester::print_test(Tester::test_function(file_system.fcreate("test7", 4096, "Bla-bla that is file test7"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.flookup(), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fread("test7"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fcopy("test7"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.flookup(), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fread("test7 - copy"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fcopy("test7"), NOT_ENOUGH_SPACE));
    Tester::print_test(Tester::test_function(file_system.fchange("test7", "New text for file test7"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.fread("test7"), CODE_SUCCESS));
    Tester::print_test(Tester::test_function(file_system.flookup(), CODE_SUCCESS));
    return 0;
}
```

```
Test 1.....Passed
Test 2.....Passed
Test 3.....Passed
Test 4.....Passed
Test 5.....Passed
test1 Size: 4096 b
test2 Size: 4096 b
test3 Size: 4096 b
test4 Size: 4096 b
Test 6.....Passed
Hello, world!
Test 7.....Passed
Ogryzok i oguzok
Test 8.....Passed
Test 9.....Passed
Test 10.....Passed
Test 11.....Passed
Test 12.....Passed
Test 13.....Passed
Test 14.....Passed
Test 15.....Passed
Test 16.....Passed
test1 Size: 4096 b
test3 Size: 4096 b
test7 Size: 4096 b
Test 17.....Passed
Bla-bla that is file test7
Test 18.....Passed
Test 19.....Passed
test1 Size: 4096 b
test3 Size: 4096 b
test7 Size: 4096 b
test7 - copy Size: 4096 b
Test 20.....Passed
Bla-bla that is file test7
Test 21.....Passed
Test 22.....Passed
Test 23.....Passed
New text for file test7
Test 24.....Passed
test1 Size: 4096 b
test3 Size: 4096 b
test7 Size: 4096 b
test7 - copy Size: 4096 b
Test 25.....Passed
```

3.1. Результаты исследования

Данные тесты показали все базовые команды, которые должна выполнять подобная файловая система: создание, удаление, копирование, запись в файл. Одноуровневая файловая система не предполагает наличие каталогов, поэтому операция перемещения тут не реализуется. Был выявлен главный минус подобной файловой системы: сегментация диска – казалось бы,

осталось два свободных блока, но они находятся не рядом друг с другом, поэтому мы можем поместить в них только два одноблочных файла, но не двублочный.

4. Ответы на контрольные вопросы:

- 4.1. Файл - именованная область данных на носителе информации, используемая как базовый объект взаимодействия с данными в операционных системах.
- 4.2. Обычный, каталог, специальный
- 4.3. Файловая система – порядок, определяющий способ хранения и организации данных на носителях информации.
- 4.4. Атрибут файла – метаданные, описывающие файл. Примером атрибута в файловой системе Linux может служить права доступа.
- 4.5. Логическая структура файла – метод организации файла. Логическая запись – наименьший элемент данных, с которым можно оперировать при обмене с внешним устройством.
- 4.6. Физическая организация файла – описания правил расположения файла на физических накопителях. Физическая запись – наименьшая единица данных, которой внешнее устройство обменивается с оперативной памятью
- 4.7. Символьное имя файла – строка символов, определяющая файл в некотором пространстве имён файловой системы.
- 4.8. Проецируемый в память файл – файл, которому или части которому ставится в соответствие некоторый диапазон адресов оперативной памяти. При этом чтение такого файла есть чтение из поставленного ему в соответствие участка памяти.

5. Вывод

В данной лабораторной работе мы были ознакомлены и получили понятие об организации файлов в файловых системах. Была реализована собственная файловая система, умеющая выполнять базовые команды, типичные для файловых систем.