

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ЛИПЕЦКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра автоматизированных систем управления

ЭМУЛЯТОР ФАЙЛОВОЙ СИСТЕМЫ

Методические указания
к выполнению индивидуального домашнего задания
по курсу «Операционные системы»

М.Г. Журавлева

Липецк
Липецкий государственный технический университет

2015

УДК 004.451 (07)

Ж911

Рецензент – В.В. Ведищев, канд. техн. наук, доц.

Журавлева, М.Г.

Ж911 Эмулятор файловой системы [Текст] : методические указания к выполнению индивидуального домашнего задания по курсу «Операционные системы» / М.Г. Журавлева. – Липецк: Изд-во Липецкого государственного технического университета, 2015. – 17 с.

Представлены теоретические сведения о назначении и устройстве файловых систем, порядок выполнения индивидуального домашнего задания и оформления отчета, варианты заданий, контрольные вопросы.

Предназначены для студентов направлений подготовки бакалавриата «Информатика и вычислительная техника», «Программная инженерия», «Математическое обеспечение и администрирование информационных систем», «Системный анализ и управление», «Механика и математическое моделирование», специальности «Информационные системы (по отраслям)», изучающих дисциплину «Операционные системы».

Могут быть использованы при изучении аналогичных дисциплин студентами других направлений.

Табл. 1. Ил. 8. Библиогр.: 5 назв.

© ФГБОУ ВПО «Липецкий
государственный технический
университет», 2015

Индивидуальное домашнее задание предполагает изучение способов построения файловых систем посредством создания эмулятора файловой системы в виде консольного приложения в операционных системах семейства Windows.

Рекомендуемый порядок выполнения индивидуального домашнего задания:

1. Проектирование файловой системы.
2. Создание алгоритмов низкоуровневого управления логическим диском (блоками файлов) и высокоуровневого управления файлами (каталогами).
2. Реализация эмулятора.
3. Тестирование программы, обработка результатов работы.
4. Оформление отчета в соответствии с СТО-13-2011 и предъявляемыми ниже требованиями.

Перед началом работы в лаборатории студент обязан пройти инструктаж по технике безопасности, ознакомиться с расположенными на стенде лаборатории инструкцией № 396 по охране труда для операторов и пользователей ЭВМ, выпиской из норм и правил ЛГТУ по охране труда, извлечениями из инструкции по пожарной безопасности зданий, сооружений и помещений ЛГТУ.

Исходные данные для индивидуального домашнего задания выбираются из таблицы приложения по номеру варианта, который соответствует номеру студента в журнале преподавателя.

1. Теоретические сведения

1.1. Понятие файлов и файловой системы

Общие сведения

Для предоставления пользователю удобных средств работы с данными, хранящимися на дисках, операционная система (ОС) подменяет физическую структуру хранящихся данных логической моделью – иерархической системой каталогов и файлов. Файл – именованный объект ОС во внешней памяти, имеющий определенное логическое представление, по отношению к которому применимы функции соответствующего интерфейса программирования приложений (API). Файловая система определяет способ организации, хранения и именования данных на диске, она содержит: все файлы на диске, структуры данных для управления файлами, системные программы, реализующие операции с файлами. Область действия файловой системы, как правило, – логический диск (том), являющийся частью (разделом) физического диска.

Для однозначной идентификации файлов файловая система назначает каждому файлу системный идентификатор файла (SFID).

Файловая система предоставляет своим клиентам сервисы хранения и каталога файлов (рис. 1). Сервис хранения скрывает физические характеристики дисков, места расположения файлов и обеспечивает сохранность файлов в случае программного или аппаратного сбоя. Сервис каталога позволяет сгруппировать файлы по каталогам и управлять доступом других пользователей к своим файлам. Каталоги – системные файлы, поддерживающие структуру файловой системы.

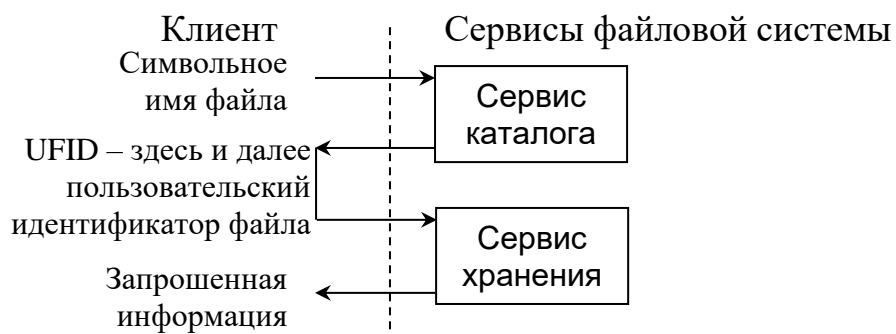


Рис. 1. Взаимодействие клиента с сервисами файловой системы

Каталог является структурированным объектом, состоящим из списка элементов, с каждым из которых связано символьное имя объекта и информация об этом объекте, в том числе SFID. Каждый каталог имеет свой SFID.

Интерфейс файловой системы

Файловая система предоставляет своим клиентам набор операций над данными во внешней памяти. Сервис каталога отвечает за разрешение полных (путевых) имен и контроль доступа. Ниже в виде «<операция> (<аргументы>), <возвращаемое значение> (<функция Windows API> [,<функция Windows API>])» представлены примерные списки операций сервисов каталога и хранения, доступные клиентам файловой системы, а также, в качестве примеров, имена некоторых соответствующих функций Windows API [1-3].

Операции сервиса каталога:

- создать каталог (полное имя каталога), выполнено (CreateDirectory);
- удалить каталог (полное имя каталога), выполнено (RemoveDirectory);
- список содержимого каталога (полное имя каталога), список содержимого каталога (FindFirstFile, FindNextFile, FindClose);
- установить права доступа к файлу или каталогу (полное имя файла или каталога, перечень пользователей и их прав), выполнено (SetNamedSecurityInfo);
- создать ссылку (полное имя каталога, полное имя файла или каталога), выполнено (CreateHardLink, CreateSymbolicLink – начиная с Windows Vista);
- создать файл (полное имя файла), выполнено (CreateFile);
- удалить файл (полное имя файла), выполнено (DeleteFile);
- открыть файл (полное имя файла, чтение/запись), UFID (OpenFile, CreateFile).

Операции сервиса хранения:

- чтение из файла (UFID, байтовый диапазон в памяти для размещения байтов), запрошенные байты (ReadFile);

- запись в файл (UFID, байтовый диапазон данных в памяти для записи), выполнено (WriteFile);
- закрыть файл (UFID), выполнено (CloseHandle);
- позиционировать указатель (UFID, позиция указателя в байтах), указатель файла (SetFilePointer).

1.2. Способы построения файловых систем

Для описания файлов требуется значительный объем информации об именах, расположении, правах доступа, атрибутах и т.п. Эти характеристики называют метаданными файлов. Далее приведено описание нескольких базовых файловых систем и их модификаций, отличающихся друг от друга структурой и/или способами хранения метаданных файлов.

FAT (Файловая система MS DOS и ранних версий Windows)

Структура тома с файловой системой FAT (название представляет собой аббревиатуру наименования основной информационной структуры – таблицы размещения файлов, File Allocation Table) представлена на рис. 2.

Загрузочная запись	Зарезервированные секторы	FAT	Копия FAT	Корневой каталог (только для, FAT8, FAT12 и FAT16)	Кластер1	Кластер2	Кластер3	...
--------------------	---------------------------	-----	-----------	--	----------	----------	----------	-----

Рис. 2. Структура тома FAT

Существует четыре версии FAT: FAT8, FAT12, FAT16 и FAT32, которые различаются размером элемента таблицы FAT – 8 бит (FAT8), 12 бит (FAT12), 16 бит (FAT16) и 32 бита (FAT32), а также расположением корневого каталога.

Адресное пространство тома (АПТ) делится на две области: системную область и область данных. Системная область состоит из загрузочной записи, зарезервированных секторов, таблицы размещения файлов (FAT), корневого каталога (для FAT32 это поле отсутствует, корневой каталог может находиться в любом месте области данных), расположенных в АПТ подряд. В области

данных находятся файлы (здесь и далее понятие «каталог» включено в понятие «файл», для FAT8, FAT12 и FAT16 подразумеваются некорневые каталоги). Область данных разбита на кластеры. Кластер представляет собой один или несколько смежных секторов в АПТ и является минимальной адресуемой единицей дисковой памяти, выделяемой файлу. Файл занимает целое число кластеров [4].

Метаданные файлов хранятся в записях соответствующих каталогов и FAT. Номер первого кластера файла находится в записи каталога (рис. 3), а остальные кластеры связываются в цепочки в FAT (рис. 4).

Количество во байт	8	3	1	10	2	2	2	4
Описание	Имя файла	Расширение	Атрибуты	Зарезервировано	Время	Дата	Номер первого кластера	Размер

Рис. 3. Формат каталоговой записи (в MS DOS)

0009	000A
000A	000B
000B	0010
000C	000D
000D	000E
000E	000F
000F	FFFF
0010	FFFF
0011	0000

Рис. 4. Фрагмент FAT, в котором хранятся цепочки кластеров двух файлов: первый файл записан в кластерах 0009, 000A, 000B, 0010, второй – в кластерах 000C, 000D, 000E, 000F; FFFF – признак конца файла

Если кластер не является конечным, то соответствующая ему ячейка в FAT содержит номер следующего кластера этого же файла. Если ячейка

соответствует последнему кластеру файла, то она содержит специальное значение.

s5fs (Базовая файловая система UNIX System V)

Том состоит из трех компонентов: суперблока, массива индексных дескрипторов и блоков хранения данных (рис. 5).

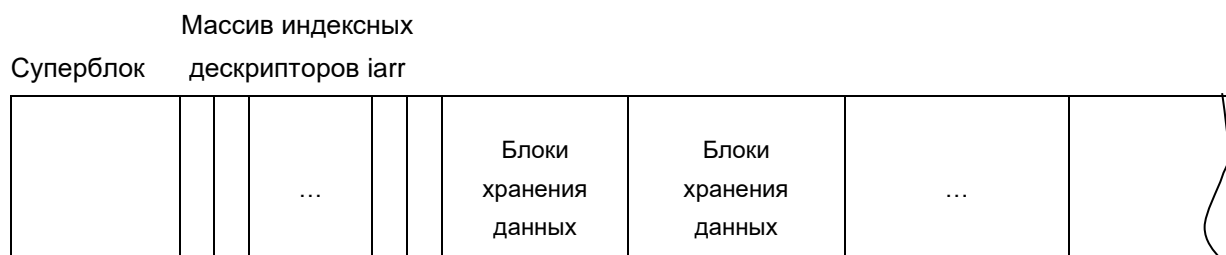


Рис. 5. Структура файловой системы s5fs

Суперблок содержит общую информацию о файловой системе, в том числе сведения об ее архитектуре, общем числе блоков и индексных дескрипторов, списки адресов свободных блоков и номеров свободных индексных дескрипторов. В индексных дескрипторах хранятся метаданные файлов. Информация о том, свободен или занят индексный дескриптор, содержится в одном из его полей. Список адресов свободных блоков может занимать несколько блоков хранения данных, но в суперблоке хранится только один блок этого списка. Первый элемент этого блока указывает на блок, хранящий продолжение списка свободных блоков, и т.д.

Массив индексных дескрипторов (iarr) содержит метаданные всех файлов файловой системы. Ядро обращается к индексному дескриптору по индексу в массиве iarr, размер которого является фиксированным и задается при создании файловой системы.

Данные файлов хранятся в блоках. Обработка файла осуществляется через индексный дескриптор, который содержит ссылки на его блоки хранения данных. Размер блока кратен 512-и байтам.

Индексный дескриптор (idesc) содержит статусную информацию о файле и указывает на расположение данных этого файла. В индексном дескрипторе не

хранится имя файла (оно хранится в каталоге) и содержимое файла (которое размещено в блоках хранения данных). При этом любой файл теоретически может иметь неограниченное число имен. Основные поля индексного дескриптора [5]:

- тип файла, дополнительные атрибуты выполнения и права доступа;
- число ссылок на файл;
- идентификаторы владельца-пользователя и владельца-группы;
- размер файла в байтах;
- время последнего доступа к файлу;
- время последней модификации индексного дескриптора;
- массив, в котором хранится тринадцать адресов дисковых блоков.

Дисковые блоки хранения данных файла в общем случае не располагаются последовательно. Первые десять элементов массива адресов дисковых блоков адресуют непосредственно блоки хранения данных файла. Одиннадцатый элемент массива адресует блок, содержащий адреса блоков хранения данных. Двенадцатый элемент – указывает на блок, хранящий адреса блоков, каждый из которых адресует блоки хранения данных файла. Тринадцатый элемент используется для тройной косвенной адресации, при этом для нахождения блока хранения данных файла используется три дополнительных блока.

Имена файлов хранятся в каталогах. Каталог файловой системы *s5fs* представляет собой таблицу, каждый элемент которой имеет фиксированный размер в 16 байтов: 2 байта хранят номер индексного дескриптора файла, 14 байтов – имя файла.

FFS (Файловая система BSD UNIX)

Файловая система *FFS* обладает полной функциональностью системы *s5fs*, но отличается от нее расположением на томе, организацией дисковых структур данных и алгоритмами размещения свободных блоков [5].

Суперблок содержит общее описание файловой системы и располагается в начале тома, но в нем не хранятся данные о массиве свободных блоков и

индексных дескрипторов. Суперблок многократно дублируется для повышения надежности.

Организация файловой системы предусматривает логическое деление дискового раздела на одну или несколько групп цилиндров (несколько последовательных дисковых цилиндров). Каждая группа цилиндров содержит резервную копию суперблока, массив индексных дескрипторов (iarr), данные о свободных блоках и итоговую информацию об использовании дисковых блоков в группе (рис. 6). Для каждой группы цилиндров при создании файловой системы выделяется место под определенное количество индексных дескрипторов (обычно на каждые 2 Кбайта блоков создается один индексный дескриптор) [5].

Суперблок	Массив свободных блоков и индексных дескрипторов	iarr	Блоки хранения данных	Суперблок	Массив свободных блоков и индексных дескрипторов	iarr	Блоки хранения данных	
-----------	--	------	-----------------------	-----------	--	------	-----------------------	--

Рис. 6. Структура файловой системы FFS

Существует возможность фрагментации блока: каждый блок может быть разбит на два, четыре или восемь фрагментов. Блок является единицей передачи данных в операциях ввода/вывода, а фрагмент определяет адресуемую единицу хранения данных на томе.

Информация о свободном пространстве хранится в виде битовой карты блоков. Битовая карта, связанная с определенной группой цилиндров, описывает свободное пространство во фрагментах для определения того, свободен данный блок или нет. Ядро анализирует биты фрагментов, составляющих блок.

Структура каталога FFS изменена для поддержки длинных имен файлов (до 255 символов). Вместо записей фиксированной длины запись каталога FFS представлена структурой, имеющей следующие поля [5]:

- номер индексного дескриптора;
- длина записи;
- длина имени файла;
- имя файла.

ext2fs (Файловая система ОС Linux)

Файловая система ext2fs разработана на основе файловой системы ufs, «наследницы» FFS [5]. Система состоит из группы блоков, логических подмножеств хранения данных на томе. Группы блоков соответствуют группам цилиндров FFS. Каждая группа блоков содержит резервную копию суперблока и индексных дескрипторов, а также битовую карту блоков и индексных дескрипторов, часть таблицы индексных дескрипторов и блоки данных.

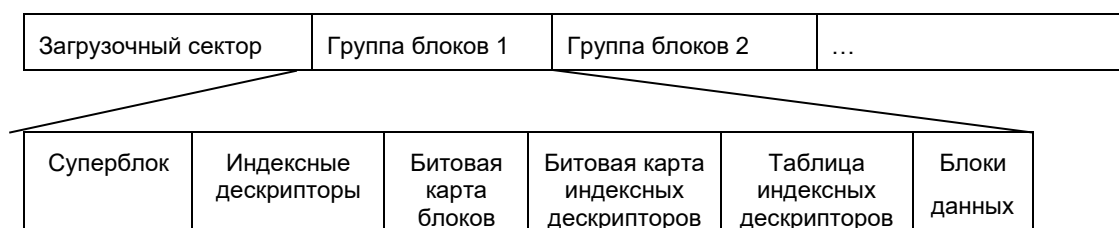


Рис. 7. Структура файловой системы ext2fs

Для хранения информации о расположении блоков файла в индексном дескрипторе выделено пятнадцать записей, каждая из которых занимает четыре байта и хранит адрес блока. Первые двенадцать записей указывают непосредственно на блоки файла. Тринадцатая запись содержит адрес блока, в котором хранятся адреса следующих блоков файла. Четырнадцатая запись используется для двойной, а пятнадцатая – для тройной косвенной адресации.

Каталоги в ext2fs образуют связные списки с входами переменной длины. Каждый вход содержит номер индексного дескриптора, длину входа, имя файла и его длину.

HPFS (Файловая система OS/2)

Идеи, положенные в основу разработки файловой системы HPFS, использованы в файловых системах современных ОС семейства Windows.

В начале тома HPFS расположено несколько управляющих блоков (рис. 7) [4]. Все остальное дисковое пространство разбито на множество областей из смежных секторов, или полос (bands). В каждой такой области располагаются и

сами данные файлов, и вспомогательная служебная информация о свободных или занятых секторах в этой области.

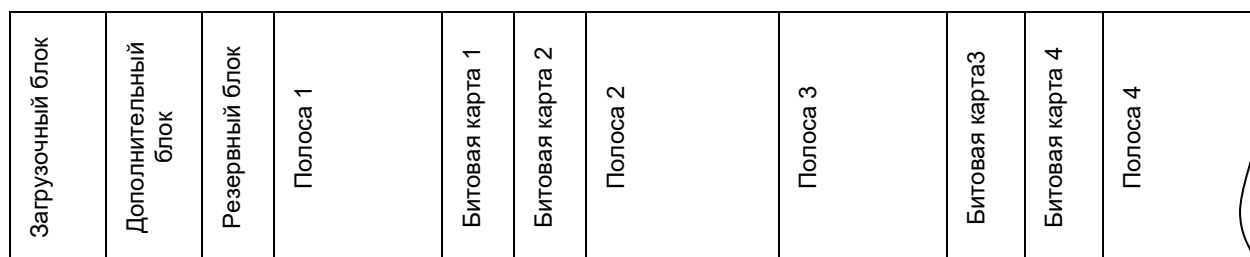


Рис. 8. Структура тома HPFS

Каждая полоса занимает на томе пространство в 8 Мбайт и имеет собственную битовую карту распределения секторов. Каждому сектору полосы данных соответствует один бит в ее битовой карте. Если бит имеет значение 1, то соответствующий сектор занят, если 0 — свободен. Последовательность полос и карт выглядит следующим образом: битовая карта, битовая карта, полоса данных, полоса данных, битовая карта, битовая карта и т. д.

Дисковое пространство в HPFS выделяется блоками, размер блока равен одному сектору. На томе с HPFS имеются еще три информационные структуры: загрузочный блок (boot block), дополнительный блок (super block) и резервный блок (spare block). Загрузочный блок содержит имя тома, его серийный номер, блок параметров BIOS и программу начальной загрузки. В дополнительном блоке содержится указатель на список битовых карт и другая информация. В списке битовых карт перечислены все блоки на диске, в которых расположены битовые карты, используемые для обнаружения свободных секторов. Резервный блок содержит указатель на карту или области аварийного замещения, указатель на список свободных запасных блоков каталогов и ряд системных флагов и дескрипторов.

Файлы в HPFS базируются на фундаментальном объекте, файловом узле (аналоге индексного дескриптора). Каждый файловый узел занимает один сектор и всегда располагается поблизости от своего файла. Файловый узел — структура, в которой содержится информация о расположении файла и его расширенных атрибутах. HPFS рассматривает файл как совокупность одного или более

секторов. Если файл непрерывен, то его размещение на диске описывается двумя 32-разрядными числами. Первое число представляет собой указатель на первый блок файла, а второе — длину экстента (фрагмента файла, располагающегося в смежных секторах диска), то есть число следующих друг за другом блоков файла. Если файл фрагментирован, то размещение его экстентов описывается в файловом узле дополнительными парами 32-разрядных чисел. В файловом узле можно разместить информацию, максимум, о восьми экстентах файла. Если файл имеет больше экстентов, то в его файловый узел записывается указатель на блок размещения, который может содержать до сорока указателей на экстенты, или, по аналогии с блоком дерева каталогов, на другие блоки размещения.

Полоса каталогов находится в центре диска и используется для хранения каталогов. Структура каталога представляет собой сбалансированное дерево с записями, расположенными в алфавитном порядке. Каждая запись, входящая в состав двоичного дерева, содержит атрибуты файла, указатель на соответствующий файловый узел и другие характеристики [4].

NTFS (Файловая система Windows NT)

NTFS — стандартная файловая система для семейства ОС Microsoft Windows NT, разработанная на основе файловой системы HPFS. Метаданные файлов хранятся в главной файловой таблице — Master File Table (MFT). Спецификации файловой системы NTFS закрыты.

2. Задание кафедры

Написать программу-эмулятор файловой системы, использующую в качестве тома участок памяти на внешнем носителе (файл). Исходные данные выбираются в соответствии с вариантом из таблицы приложения.

Минимальный набор операций интерфейса пользователя:

- создать каталог;
- удалить каталог;
- вывести список содержимого каталога;
- создать файл в заданном каталоге;

- удалить файл.

3. Цель работы

Изучение особенностей построения файловых систем и приобретение практических навыков их реализации.

4. Содержание отчета

1. Титульный лист.
2. Задание кафедры.
3. Цель работы.
4. Спецификации разрабатываемой файловой системы.
5. Документы «Описание программы», «Описание применения», «Руководство программиста» в соответствии со стандартами ЕСПД.
6. Листинг программы.

5. Контрольные вопросы

1. Понятие файлов и файловой системы.
2. Способы построения файловых систем.
3. Метаданные файлов.
4. Функции управления файлами.
5. Создание, открытие и закрытие файла.
6. Чтение файла.
7. Запись в файл.
8. Перемещение указателя файла.
9. Получение и установка атрибутов файла.
10. Сжатые и зашифрованные файлы.

6. Библиографический список

1. Microsoft Developer Network [Эл. ресурс]. – Режим доступа: [www. URL: msdn.microsoft.com](http://www.msdn.microsoft.com). – 22.10.2015.
2. Харт, Дж. М. Системное программирование в среде Windows [Текст] / Джонсон М. Харт. – Москва: Вильямс, 2005. – 592 с.
3. Русинович, М. Внутреннее устройство Microsoft Windows [Текст] / М. Русинович, Д. Соломон. – Санкт-Петербург: Питер, 2013. – 800 с.
4. Гордеев, А.В. Операционные системы [Текст]: учебн. для вузов / А.В. Гордеев. – Санкт-Петербург: Питер, 2009. – 416 с.
5. Робачевский, А.М. Операционная система Unix [Текст] / А.М. Робачевский, С.А.Немнюгин, О.Л.Стесик. – Санкт-Петербург: БХВ-Петербург, 2010. – 656 с.

Варианты заданий

№ варианта	Прототип файловой системы (FAT - 1, s5fs - 2, FFS - 3, ext2fs - 4, HPFS - 5, другой прототип - 6)	Размер блока кратно, байт	Способ хранения дерева каталогов (1 – список дочерних узлов, 2 – дерево)	№ варианта	Прототип файловой системы (FAT - 1, s5fs - 2, FFS - 3, ext2fs - 4, HPFS - 5, другой прототип - 6)	Размер блока кратно, байт	Способ хранения дерева каталогов (1 – список дочерних узлов, 2 – дерево)
1	1	512	1	21	3	1024	1
2	2	512	1	22	4	1024	1
3	3	512	1	23	5	1024	1
4	4	512	1	24	6	1024	1
5	5	512	1	25	1	2048	1
6	6	512	1	26	2	2048	1
7	1	1024	2	27	3	2048	1
8	2	1024	2	28	4	2048	1
9	3	1024	2	29	5	2048	1
10	4	1024	2	30	6	2048	1
11	5	1024	2	31	1	4096	2
12	6	1024	2	32	2	4096	2
13	1	512	2	33	3	4096	2
14	2	512	2	34	4	4096	2
15	3	512	2	35	5	4096	2
16	4	512	2	36	6	4096	2
17	5	512	2	37	1	8192	1
18	6	512	2	38	2	8192	1
19	1	1024	1	39	3	8192	1
20	2	1024	1	40	4	8192	1

ЭМУЛЯТОР ФАЙЛОВОЙ СИСТЕМЫ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

**к выполнению индивидуального домашнего задания
по курсу «Операционные системы»**

Журавлева Марина Гарриевна

Редактор Г. В. Казьмина

Подписано в печать

Формат 60x84 1/16. Бумага офсетная.

Ризография. Печ.л. 1,0. Тираж 90 экз. Заказ *N* .

Издательство Липецкого государственного технического университета.

Полиграфическое подразделение Издательства ЛГТУ.

398600, Липецк, ул. Московская, 30.