

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ
Факультет физико-математических и естественных наук
Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №1
дисциплина: Архитектура компьютера

Студент: Репкина Елизавета
Группа: НКАбд-04-24

Москва
2024г.

Содержание

1	Цель работы
2	Задание
3	Теоретическое введение
4	Выполнение лабораторной работы.....
5	Выводы.....
6	Ответы на контрольные вопросы для самопроверки.....
7	Источники

1.Цель работы:

Приобретение практических навыков работы с операционной системой на уровне командной строки (организация файловой системы, навигация по файловой системе, создание и удаление файлов и директорий).

2. Задания

1. Перемещение по файловой системе.
2. Создание пустых каталогов и файлов.
3. Перемещение и удаление файлов или каталогов.
4. Команда cat: вывод содержимого файлов.
5. Выполнение заданий для самостоятельной работы.

3. Теоретическое введение

Введение в GNU Linux

Операционная система (ОС)— это комплекс взаимосвязанных программ, предназначенных для управления ресурсами компьютера и организации взаимодействия с пользователем. Сегодня наиболее известными операционными системами являются ОС семейства Microsoft Windows и UNIX-подобные системы. GNU Linux — семейство переносимых, многозадачных и многопользовательских операционных систем, на базе ядра Linux, включающих тот или иной набор утилит и программ проекта GNU, и, возможно, другие компоненты. Как и ядро Linux, системы на его основе, как правило, создаются и распространяются в соответствии с моделью разработки свободного и открытого программного обеспечения (Open-Source Software). Linux-системы распространяются в основном бесплатно в виде различных дистрибутивов. Дистрибутив GNU Linux — общее определение ОС, использующих ядро Linux и набор библиотек и утилит, выпускаемых в рамках проекта GNU, а также графическую оконную подсистему X Window System. Дистрибутив готов для конечной установки на пользовательское оборудование. Кроме ядра и, собственно, операционной системы дистрибутивы обычно содержат широкий набор приложений, таких как редакторы документов и таблиц, мультимедийные проигрыватели, системы для работы с базами данных и т.д. Существуют дистрибутивы, разрабатываемые как при коммерческой поддержке (Red Hat / Fedora, SLED / OpenSUSE, Ubuntu), так и исключительно усилиями добровольцев (Debian, Slackware, Gentoo, ArchLinux).

Введение в командную строку GNU Linux

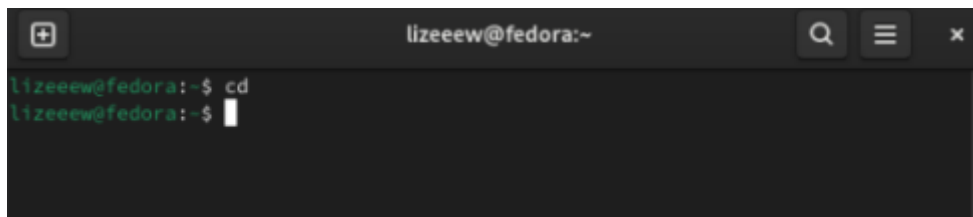
Работу ОС GNU Linux можно представить в виде функционирования множества взаимосвязанных процессов. При загрузке системы сначала запускается ядро, которое, в свою очередь, запускает оболочку ОС (от англ. shell «оболочка»). Взаимодействие пользователя с системой Linux (работа с данными и управление работающими в системе процессами) происходит в интерактивном режиме посредством командного языка. Оболочка операционной системы (или командная оболочка, интерпретатор команд) — интерпретирует (т.е. переводит на машинный язык) вводимые пользователем команды, запускает соответствующие программы (процессы), формирует и выводит ответные сообщения. Кроме того, на языке командной оболочки можно писать небольшие программы для выполнения ряда последовательных операций с файлами и содержащимися в них данными — сценарии (скрипты).

4. Выполнение лабораторной работы

Перемещение по файловой системе

Открываю терминал. По умолчанию терминал открывается в домашнем каталоге пользователя, который обозначается символом `~`. `user@dk4n31:~$`

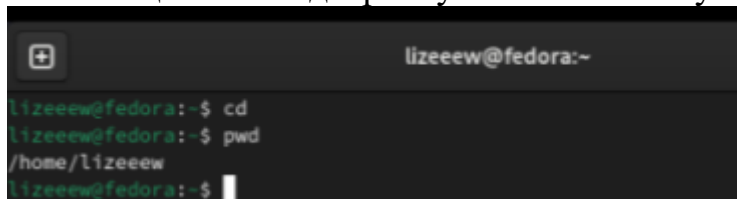
Убеждаюсь, что я нахожусь в домашнем каталоге. Если это не так, перехожу в него. Это можно сделать с помощью команды `cd` без аргументов.



```
lizeew@fedora:~$ cd
lizeew@fedora:~$
```

Рис 1- проверка нахождения в домашнем каталоге

С помощью команды `pwd` узнаю полный путь к своему домашнему каталогу.




```
lizeew@fedora:~$ cd
lizeew@fedora:~$ pwd
/home/lizeew
lizeew@fedora:~$
```

Рис 2- полный путь к домашнему каталогу

Команда `cd` позволяет сменить текущий каталог на другой, указав путь к нему в качестве параметра. Формат команды:

Команда `cd` работает как с абсолютными, так и с относительными путями.

Перехожу в подкаталог Документы домашнего каталога указав относительный путь



```
lizeew@fedora:~$ cd Документы
lizeew@fedora:~/Документы$
```

Рис 3 – переход к папке Документы

Перехожу в каталог `local` – подкаталог `usr` корневого каталога указав абсолютный путь к нему (`/usr/local`):



```
lizeew@fedora:~$ cd /usr/local
lizeew@fedora:~/Документы$ cd /usr/local
lizeew@fedora:/usr/local$
```

Рис 4 – переход в папку local по абсолютному пути

Обращаю внимание абсолютный путь всегда начинается от корневого каталога (т.е. с символа /).

Можно использовать комбинацию 'cd -' для возвращения в последний посещённый пользователем каталог. А 'cd ..' используется для перехода на один каталог выше по иерархии. Введите последовательно эти команды. В каком каталоге Вы находитесь?

```
lizeew@fedora:/usr/local$ cd -  
/home/lizeew/Документы  
lizeew@fedora:~/Документы$ cd ..  
lizeew@fedora:~$
```

Рис 5-последовательное использование команд cd-, cd ..

Для просмотра списка файлов текущего каталога может быть использована команда ls без аргументов.

Перехожу в домашний каталог

Вывожу список файлов домашнего каталога.

```
lizeew@fedora:~$ ls  
Видео  Документы  Загрузки  Изображения  Музыка  Общедоступные  'Рабочий стол'  Шаблоны  
lizeew@fedora:~$
```

Рис 6 – Вывод файлов домашнего каталога

Открываю домашний каталог с помощью файлового менеджера графического окружения ОС

.

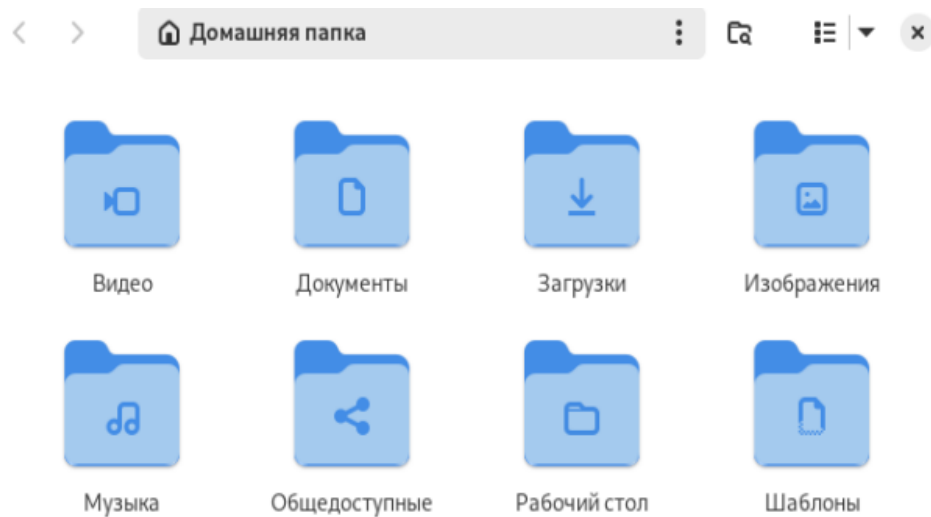


Рис 7 – Домашний каталог

Убеждаюсь в том, что список файлов полученных с помощью команды `ls` совпадает с файлами, отображающимися в графическом файловом менеджере.

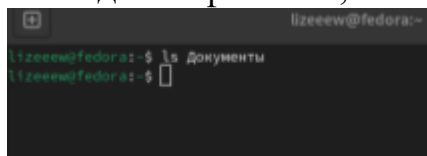
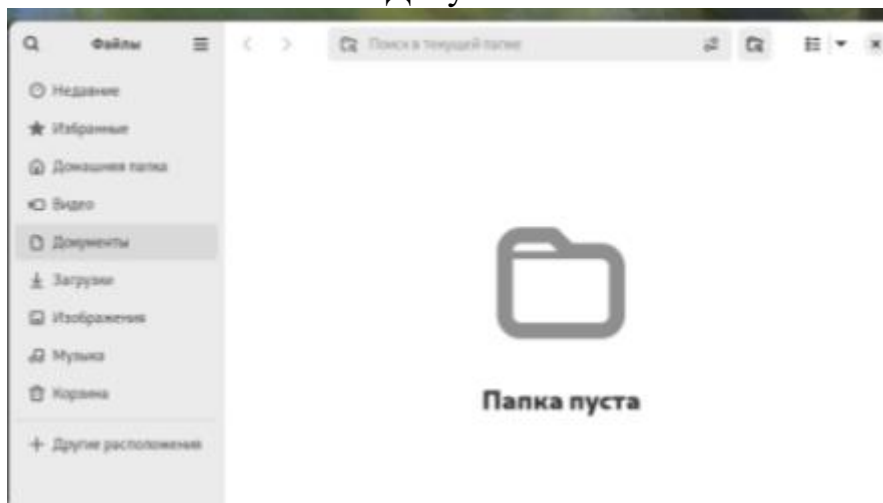


Рис 8 –Файлы каталога Документы



Вывожу список файлов подкаталога Документы моего домашнего каталога указав относительный путь

Вывожу список файлов каталога `/usr/local` указав абсолютный путь к нему:

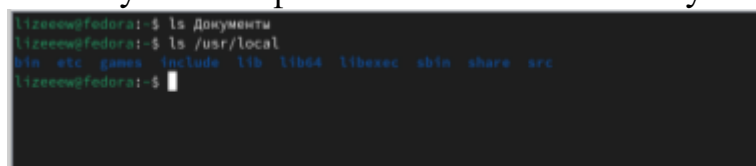


Рис 9- Файлы каталога local через абсолютный путь

Включаю в отчет примеры использования команды `ls` с разными ключами.

```
lizeew@fedora:~$ ls /usr/local -l
итого 0
drwxr-xr-x. 1 root root 0 миа 24 2024 bin
drwxr-xr-x. 1 root root 0 миа 24 2024 etc
drwxr-xr-x. 1 root root 0 миа 24 2024 games
drwxr-xr-x. 1 root root 0 миа 24 2024 include
drwxr-xr-x. 1 root root 0 миа 24 2024 lib
drwxr-xr-x. 1 root root 6 апр 15 01:57 lib64
drwxr-xr-x. 1 root root 0 миа 24 2024 libexec
drwxr-xr-x. 1 root root 0 миа 24 2024 sbin
drwxr-xr-x. 1 root root 38 апр 15 01:57 share
drwxr-xr-x. 1 root root 0 миа 24 2024 src
```

Создание пустых каталогов и файлов

Для создания каталогов используется команда `mkdir`.

Создаю в домашнем каталоге подкаталог с именем `parentdir`

С помощью команды `ls` проверяю, что каталог создан.

```
lizeew@fedora:~$ mkdir parentdir
mkdir: невозможно создать каталог «parentdir»: Файл существует
lizeew@fedora:~$ ls
parentdir  Видео  Документы  Загрузки  Изображения  Музыка  Общедоступные  'Рабочий стол'  Шаблоны
lizeew@fedora:~$
```

Рис 10-создание каталога `parentdir`

Создаю подкаталог в существующем каталоге:

```
lizeew@fedora:~$ mkdir dir1 dir2 dir3
lizeew@fedora:~$ ls
dir1  dir3  parentdir  Документы  Изображения  Общедоступные  Шаблоны
dir2  newdir  Видео      Загрузки   Музыка       'Рабочий стол'
lizeew@fedora:~$
```

Рис 11- создание подкаталога

Если требуется создать подкаталог в каталоге, отличном от текущего, то путь к нему требуется указать в явном виде:

```
user@dk4n31:~$ mkdir ~/newdir
```

Эта команда должна создать каталог `newdir` в домашнем каталоге (`~`).

Проверяю это с помощью команды

```
lizeew@fedora:~/parentdir$ mkdir ~/newdir
lizeew@fedora:~/parentdir$ ls ~
newdir  parentdir  Видео  Документы  Загрузки  Изображения  Музыка  Общедоступные  'Рабочий стол'  Шаблоны
lizeew@fedora:~/parentdir$
```

Рис 12- создание каталога в домашнем каталоге

Опция – parents (краткая форма -p) позволяет создавать иерархическую цепочку подкаталогов, создавая все промежуточные каталоги. Создаю следующую последовательность вложенных каталогов newdir/dir1/dir2 в домашнем каталоге

```
lizeew@fedora:~$ mkdir -p ~/newdir/dir1/dir2
lizeew@fedora:~$ ls ~/newdir/dir1
dir2
lizeew@fedora:~$
```

Рис 13-создание цепочки каталогов

Для создания файлов может быть использована команда touch

Создаю файл test.txt в каталоге ~/newdir/dir1/dir2

Проверяю наличие файла

```
lizeew@fedora:~$ touch ~/newdir/dir1/dir2/test.txt
lizeew@fedora:~$ ls ~/newdir/dir1/dir2
test.txt
lizeew@fedora:~$
```

Рис 14 – создание файла через цепочку каталогов

Перемещение и удаление файлов или каталогов

Команда `rm` удаляет файлы и (или) каталоги и имеет следующий синтаксис:
`rm [опции] <файл|каталог> [файл|каталог...]`

Опции команды `rm`:

- `-r` или `-R`: рекурсивное удаление (это обязательная опция для удаления любого каталога, пустого или содержащего файлы и (или) подкаталоги);
- `-i`: запрос подтверждения перед удалением;
- `-v`: вывод подробной информации при выполнении команды;
- `-f`: принудительное удаление файлов или каталогов.

Для удаления пустых каталогов можно воспользоваться командой `rmdir`.

Запросив подтверждение на удаление каждого файла в текущем каталоге, удаляю в подкаталоге `/newdir/dir1/dir2/` все файлы с именами, заканчивающимися на `.txt`:

```
user@dk4n31:~$ rm -i ~/newdir/dir1/dir2/*.txt
```

```
bash-5.2$
```

```
bash-5.2$ rm -i ~/newdir/dir1/dir2/*.txt
```

Рис 15- удаление всех файлов с расширением `.txt`

Рекурсивно удаляю из текущего каталога без запроса подтверждения на удаление каталог `newdir`, а также файлы, чьи имена начинаются с `dir` в каталоге

parentdir:

```
user@dk4n31:~$ rm -R ~/newdir ~/parentdir/dir*
```

```
est.txt
```

```
izeew@fedora:~$ rm -R ~/newdir ~ /parentdir/dir*
```

Рис 16-удаление пустых каталогов

Команда `rm` удаляет файлы безвозвратно, и не существует способа для их восстановления.

Команда `mv` служит для перемещения файлов и каталогов и имеет следующий синтаксис:

```
mv [опции] <файл|каталог> [файл|каталог...] <назначение>
```

Некоторые опции:

- `-f`: принудительное выполнение операции (предупреждение не будет выводиться даже

при перезаписи существующего файла);

- `-i`: запрашивается подтверждение перед перезаписью существующего файла;

- `-v`: подробный режим, который сообщает обо всех изменениях и действиях при выполнении команды.

Команда `cp` копирует файлы и каталоги и имеет следующий синтаксис:

```
cp [опции] <файл|каталог> [файл|каталог ...] <назначение>
```

Некоторые опции команды `cp`:

- `-R`: рекурсивное копирование; является обязательной опцией для копирования каталогов;

- `-i`: запрос подтверждения перед перезаписью любых файлов;

- `-f`: заменяет любые существующие файлы без запроса подтверждения;

- `-v`: подробный режим, сообщает обо всех изменениях и действиях.

Для демонстрации работы команд `cp` и `mv` приведем следующие примеры.

Создаю файлы и каталоги в домашнем каталоге

```
izeew@fedora:~$ mkdir -p parentdir1/dir1 parentdir2/dir2 parentdir3/dir3
izeew@fedora:~$ touch parentdir1/dir1/test1.txt parentdir2/dir2/test2.txt
izeew@fedora:~$
```

Рис 18-создание каталогов и файлов в этих каталогах

Используя команды `cp` и `mv` файл `test1.txt` скопируйте, а `test2.txt` перемещая в каталог `parentdir3`:

```
user@dk4n31:~$ mv parentdir1/dir1/test1.txt parentdir3
```

```
user@dk4n31:~$ cp parentdir2/dir2/test2.txt parentdir3
```

С помощью команды `ls` проверяю корректность выполненных команд

```
bash-5.2$ mv parentdir1/dir1/test1.txt parentdir3
bash-5.2$ cp parentdir2/dir2/test2.txt parentdir3
bash-5.2$ ls parentdir3
dir3  test1.txt  test2.txt
bash-5.2$ ls parentdir2/dir2
test2.txt
bash-5.2$ ls parentdir1/dir1
bash-5.2$
```

Рис 19-перемещение и копирование файлов

Также команда `mv` может быть использована для переименования файлов и каталогов, а команда `cp` позволяет сделать копию файла с новым именем. Переименовываю файл `test1.txt` из каталога `parentdir3` в `newtest.txt`, запрашивая подтверждение перед перезаписью.

```
bash-5.2$ mv -i parentdir3/test1.txt parentdir3/newtest.txt
bash-5.2$ ls parentdir3
dir3  newtest.txt  test2.txt
bash-5.2$
```

Рис 20-переименование файла

Переименовываю каталог `dir1` в каталоге `parentdir1` в `newdir`:

```
bash-5.2$ mv parentdir1/dir1 parentdir1/newdir
bash-5.2$ ls parentdir1
newdir
```

Рис 21- переименование каталога

Команда cat: вывод содержимого файлов

Команда cat объединяет файлы и выводит их на стандартный вывод (обычно это экран)

```
bash-5.2$ cat /etc/hosts
# Loopback entries; do not change.
# For historical reasons, localhost precedes localhost.localdomain:
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
# See hosts(5) for proper format and other examples:
# 192.168.1.10 foo.example.org foo
# 192.168.1.13 bar.example.org bar
```

рис 22- выполнение команды cat

Задание для самостоятельной работы

1. Воспользовавшись командой pwd, узнайте полный путь к своей домашней директории.

```
bash-5.2$ pwd
/home/lizeew
```

- 2.
- 3.

2. Введите следующую последовательность команд

```
cd
mkdir tmp
cd tmp
pwd
cd /tmp
pwd
```

```
/home/lizeeeew/tmp
bash-5.2$ cd
bash-5.2$ mkdir tmp
mkdir: невозможно создать каталог «tmp»: Файл существует
bash-5.2$ cd tmp
bash-5.2$ pwd
/home/lizeeeew/tmp
bash-5.2$ cd /tmp
bash-5.2$ pwd
/tmp
bash-5.2$
```

Объясните, почему вывод команды `pwd` при переходе в каталог `tmp` дает разный результат.

При добавлении `/` в команду `cd` при переходе на `1` каталог используется как абсолютный переход к каталогу. После выполнения команды `cd /` путь терминала начинается с папки назначения.

3. Пользуясь командами `cd` и `ls`, посмотрите содержимое корневого каталога, домашнего каталога, каталогов `/etc` и `/usr/local`.

```

bash-5.2$ cd
bash-5.2$ ls /etc
abrt                                init.d                             profile.d
adjtime                            inittab                           protocols
aliases                           inputrc                           pulse
alsa                               ipppusb                           qemu
alternatives                      iscsi                             qemu-ga
anaconda                          issue                              rc0.d
anthy-unicode.conf                issue.d                           rc1.d
asound.conf                       issue.net                          rc2.d
audit                             java                               rc3.d
authselect                        jvm                               rc4.d
avahi                             jvm-common                        rc5.d
bash_completion.d                kdump                             rc6.d
bashrc                             kdump.conf                       rc.d
bindresvport.blacklist            kernel                            reader.conf.d
binfmt.d                          keys                              redhat-release
bluetooth                        keyutils                         request-key.conf
brlapi.key                       krb5.conf                        request-key.d
brltty                            krb5.conf.d                     resolv.conf
brltty.conf                      ld.so.cache                      rpc
ceph                             ld.so.conf                       rpm
chkconfig.d                     ld.so.conf.d                     rsyncd.conf
chromium                         libaudit.conf                   rtab.d
chrony.conf                      libblockdev                      rygel.conf
cifs-utils                      libibverbs.d                    samba
colord                           libnl                             same.d
containers                       libreport                        sasl2
credstore                       libssh                           security
credstore.encrypted             libuser.conf                    selinux
crypto-policies                 libvirt                          services
crypttab                        locale.conf                     sestatus.conf
csh.cshrc                       localtime                       sgml
csh.login                       login.defs                      shadow
cups                             logrotate.conf                 shadow-
cupshelpers                     logrotate.d                     shells
dbus-1                           lvm                              skel
dconf                            machine-id                      sos
debuginfod                     magic                           speech-dispatcher
default                          mailcap                         ssh
depmod.d                       makedumpfile.conf.sample       ssl
dhcpcd                          man_db.conf                     sssd
DIR_COLORS                     mcelog                          statetab.d
DIR_COLORS.lightbgcolor         mdevctl.d                      subgid
dley-na-server-service.conf     mime.types                     subgid-

```

```

bash-5.2$ ls /usr/local
bin  etc  games  include  lib  lib64  libexec  sbin  share  src
bash-5.2$

```

4. Пользуясь изученными консольными командами, в своём домашнем каталоге создайте каталог temp и каталог labs с подкаталогами lab1, lab2 и lab3 одной командой.

```

bash-5.2$ mkdir temp -p ~/labs/lab1/lab2/lab3
bash-5.2$

```

В каталоге temp создайте файлы text1.txt, text2.txt, text3.txt. Пользуясь командой ls, убедитесь, что все действия выполнены успешно (каталоги и файлы созданы).

```

ash-5.2$ touch temp/test.txt
ash-5.2$ ls temp
test.txt
ash-5.2$ touch temp/test2.txt
ash-5.2$ ls temp
test2.txt  test.txt
ash-5.2$ touch temp/test3.txt
ash-5.2$ la temp
ash: la: команда не найдена
ash-5.2$ ls temp
test2.txt  test3.txt  test.txt
ash-5.2$

```

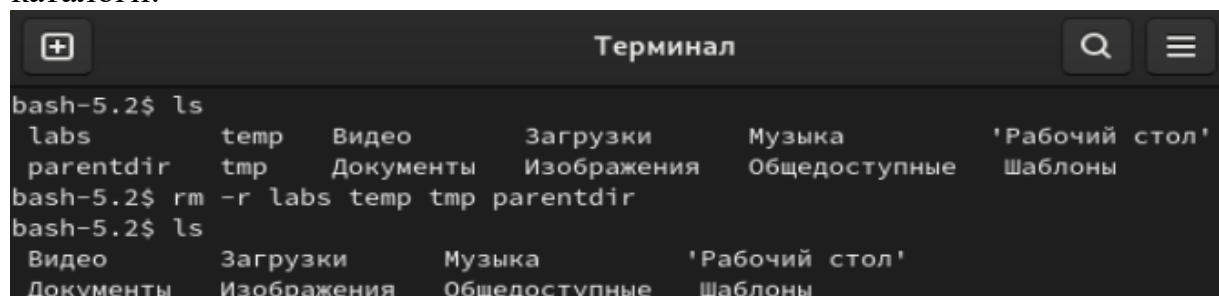
5. С помощью любого текстового редактора (например, редактора mcedit) запишите в файл text1.txt свое имя, в файл text2.txt фамилию, в файл text3.txt учебную группу. Выведите на экран содержимое файлов, используя команду cat.


```
bash-5.2$ cat temp/text1.txt temp/text2.txt temp/text3.txt
Репкина
Елизавета
НКАБд-04-24
```

1. Скопируйте все файлы, чьи имена заканчиваются на .txt, из каталога ~/temp в каталог labs. После этого переименуйте файлы каталога labs и переместите их: text1.txt переименуйте в firstname.txt и переместите в подкаталог lab1, text2.txt в lastname.txt в подкаталог lab2, text3.txt в id-group.txt в подкаталог lab3. Пользуясь командами ls и cat, убедитесь, что все действия выполнены верно.

```
bash-5.2$ mv text1.txt firstname.txt
bash-5.2$ mv firstname.txt lab1
bash-5.2$ mv text2.txt lastname.txt
bash-5.2$ mv lastname.txt lab2
bash-5.2$ mv text3.txt id-group.txt
bash-5.2$
bash-5.2$ mv id-group.txt lab3
bash-5.2$ cat lab1 lab2 lab3
Репкина
Елизавета
НКАБд-04-24
```

2. Удалите все созданные в ходе выполнения лабораторной работы файлы и каталоги.



```
bash-5.2$ ls
labs      temp  Видео  Загрузки  Музыка  'Рабочий стол'
parentdir tmp   Документы  Изображения  Общедоступные  Шаблоны
bash-5.2$ rm -r labs temp tmp parentdir
bash-5.2$ ls
Видео      Загрузки  Музыка  'Рабочий стол'
Документы  Изображения  Общедоступные  Шаблоны
```

5. Выводы

При выполнении данной лабораторной работы я приобрела практические навыки работы с операционной системой на уровне командной строки, изучила организацию файловой системы, научилась создавать и удалять файлы и директории.

6. Ответы на контрольные вопросы для самопроверки.

1. Командная строка – это текстовый интерфейс между человеком и компьютером, в котором инструкции компьютеру даются путём ввода с клавиатуры текстовых строк.
2. Для получения достаточно подробной информации по каждой из команд можно использовать команду `man`: “`man ls`”
3. Абсолютный путь — начинается от корневого каталога (`/`), образуется перечислением всех каталогов, разделённых прямым слешем (`/`), и завершается именем файла. Относительный путь тоже строится перечислением через (`/`) всех каталогов, но начинается от каталога, в котором “находится” пользователь.
4. Определить абсолютный путь к текущей директории можно с помощью утилиты `pwd`.
5. При помощи команд `rmdir` и `rm` можно удалить файл и каталог? Командой `rmdir` нельзя удалить файлы, а командой `rm` можно удалить файлы и директории (с помощью опции `-r`). Утилита `rmdir` удаляет только пустые каталоги.
6. Запустить несколько команд в одной строке можно, перечисляя их через точку с запятой. Например: `cd /my_folder; rm *.txt`. Также можно использовать логические И и ИЛИ как `&` и `||` соответственно.
7. `-l` – выводит дополнительные параметры файлов (права доступа, владельцы и группы, размеры файлов и время последнего доступа)
8. Информацию о скрытых файлах можно просмотреть утилитой `ls` с ключами `-la`. `-l` – выведет дополнительную информацию о файлах, `-a` – выведет скрытые файлы. Можно использовать только ключ `-a`, если дополнительная информация о файле не нужна.
9. Для автоматического дополнения вводимых команд может служить клавиша `Tab`.

7.Список литературы

1. GDB: The GNU Project Debugger. — URL: <https://www.gnu.org/software/gdb/>.
2. GNU Bash Manual. — 2016. — URL: <https://www.gnu.org/software/bash/manual/>.
3. Midnight Commander Development Center. — 2021. — URL: <https://midnight-commander.org/>.
4. NASM Assembly Language Tutorials. — 2021. — URL: <https://asmtutor.com/>.
5. Newham C. Learning the bash Shell: Unix Shell Programming. — O'Reilly Media, 2005. — 354 с. — (In a Nutshell). — ISBN 0596009658. — URL: <http://www.amazon.com/Learningbash-Shell-Programming-Nutshell/dp/0596009658>.
6. Robbins A. Bash Pocket Reference. — O'Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
7. The NASM documentation. — 2021. — URL: <https://www.nasm.us/docs.php>.
8. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
9. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
10. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс, 2017.
11. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
12. Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.
13. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВПетербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
14. Столяров А. Программирование на языке ассемблера NASM для ОС Unix. — 2-е изд. — М. : МАКС Пресс, 2011. — URL: http://www.stolyarov.info/books/asm_unix.
15. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
16. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015. — 1120 с. — (Классика Computer Science).