

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: Газизянов Владислав Альбертович

Группа: НПИбд-01-23

МОСКВА

2023г.

Оглавление

1.Цель работы.....	3
2.Теоретическое введение	4
2.1 Введение в GNU Linux	4
2.2 Введение в командную строку GNU Linux.....	4
2.3Файловая структура GNU Linux: каталоги и файл	7
2.4Базовые команды bash	8
2.5Полезные комбинации клавиш	8
3.Выполнение лабораторной работы	10
3.1 Перемещение по файловой системе.....	10
3.2Создание пустых каталогов и файлов	11
3.3 Перемещение и удаление файлов или каталогов	11
4.Самостоятельная работа.....	12
5.Вопросы для самопроверки	14
6.Вывод.....	16

1.Цель работы

Получение практического опыта в управлении операционной системой через командную строку, включая организацию файловой системы, перемещение по файловой системе, а также создание и удаление файлов и папок.

2.Теоретическое введение

2.1 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).

2.2 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). Из командных оболочек GNU Linux наиболее популярны `bash`, `csh`, `ksh`, `zsh`. Команда `echo $SHELL` позволяет проверить, какая оболочка используется. В качестве предустановленной командной оболочки GNU Linux используется одна из наиболее распространённых разновидностей командной оболочки — `bash` (Bourne again shell). В GNU Linux доступ пользователя к командной оболочке обеспечивается через терминал (или консоль). Запуск терминала можно осуществить через главное меню Приложения Стандартные Терминал (или Консоль) или нажав `Ctrl + Alt + t`. Интерфейс командной оболочки очень прост. Обычно он состоит из приглашения командной строки (строки, оканчивающейся символом `$`), по которому пользователь вводит команды: `iivanova@dk4n31:~$` Это приглашение командной оболочки, которое несёт в себе информацию об имени пользователя `iivanova`, имени компьютера `dk4n31` и текущем каталоге, в котором находится пользователь, в данном случае это домашний каталог пользователя, обозначенный как

~. Команды могут быть использованы с ключами (или опциями) — указаниями, модифицирующими поведение команды. Ключи обычно начинаются с символа (-) или (--) и часто состоят из одной буквы. Кроме ключей после команды могут быть использованы аргументы (параметры) — названия объектов, для которых нужно выполнить команду (например, имена файлов и каталогов). Например, для подробного просмотра содержимого каталога documents может быть использована команда ls с ключом -l:

```
iivanova@dk4n31:~$ ls -l documents
```

В данном случае ls – это имя команды, l – ключ, documents – аргумент. Команды, ключи и аргументы должны быть отделены друг от друга пробелом. Ввод команды завершается нажатием клавиши Enter , после чего команда передаётся оболочке на исполнение. Результатом выполнения команды могут являться сообщения о ходе выполнения команды или об ошибках. Появление приглашения командной строки говорит о том, что выполнение команды завершено. Иногда в GNU Linux имена программ и команд слишком длинные, однако bash может завершать имена при их вводе в терминале. Нажав клавишу Tab , можно завершить имя команды, программы или каталога. Например, предположим, что нужно использовать программу mcedit. Для этого наберите в командной строке mc, затем нажмите один раз клавишу Tab . Если ничего не происходит, то это означает, что существует несколько возможных вариантов завершения команды. Нажав клавишу Tab ещё раз, можно получить список имён, начинающихся с mc:

```
iivanova@dk4n31:~$ mc
mc mcd mcedit mclasserase mcookie mcview
mcat mcdiff mcheck mcomp mcopy
iivanova@dk4n31:~$ mc
```

2.3 Файловая структура GNU Linux: каталоги и файлы

Файловая система определяет способ организации, хранения и именования данных на носителях информации в компьютерах и представляет собой иерархическую структуру в виде вложенных друг в друга каталогов (директорий), содержащих все файлы. В ОС Linux каталог, который является “вершиной” файловой системы, называется корневым каталогом, обозначается символом / и содержит все остальные каталоги и файлы. В большинстве Linux-систем поддерживается стандарт иерархии файловой системы (Filesystem Hierarchy Standard, FHS), унифицирующий местонахождение файлов и каталогов. Это означает, что в корневом каталоге находятся только подкаталоги со стандартными именами и типами данных, которые могут попасть в тот или иной каталог. Так, в любой Linux-системе всегда есть каталоги /etc, /home, /usr/bin. Обратиться к файлу, расположенному в каком-то каталоге, можно указав путь к нему. Существует несколько видов путей к файлу: • полный или абсолютный путь — начинается от корня (/), образуется перечислением всех каталогов, разделённых прямым слешем (/), и завершается именем файла (например, полный путь к файлу addition.txt из каталога user в каталоге home, находящемся в корневом каталоге, будет иметь вид: /home/user/documents/addition.txt; • относительный путь — так же как и полный путь, строится перечислением через (/) всех каталогов, но начинается от текущего каталога (каталога, в котором “находится” пользователь), т.е. пользователь, находясь в каталоге user, может обратиться к файлу addition.txt, указав относительный путь documents/addition.txt. Таким образом, в Linux если имя объекта начинается с /, то системой это интерпретируется как полный путь, в любом другом случае —

как относительный. В Linux любой пользователь имеет домашний каталог, который, как правило, имеет имя пользователя. В домашних каталогах хранятся документы и настройки пользователя. Для обозначения домашнего каталога используется знак тильды (~). При переходе из домашнего каталога знак тильды будет заменён на имя нового текущего каталога.

2.4 Базовые команды *bash*

В операционной системе GNU Linux взаимодействие пользователя с системой обычно осуществляется с помощью командной строки посредством построчного ввода команд.

Первые задачи, которые приходится решать в любой системе это — работа с данными (обычно хранящимися в файлах) и управление работающими в системе программами (процессами). Для получения достаточно подробной информации по каждой из команд используйте команду `man`, например,:

```
user@dk4n31:~$ man ls
```

2.5 Полезные комбинации клавиш

Для удобства и экономии времени при работе в терминале существует большое количество сокращённых клавиатурных команд. Клавиши ↑ и ↓ позволяют увидеть историю предыдущих команд в `bash`. Количество хранимых строк определено в переменной окружения `HISTSIZE`. Клавиши ← и → перемещают курсор влево и вправо в текущей строке, позволяя редактировать команды. Сочетания клавиш `Ctrl + a` и `Ctrl + e` перемещают курсор в начало и в конец текущей строки. Клавиши `Ctrl + k` удаляет всё от текущей позиции курсора до конца строки, а `Ctrl + w` или `Alt + Backspace` удаляют слово перед курсором. Сочетание клавиш `Ctrl + d` в пустой строке служит для завершения текущего

сеанса. Для завершения выполняющейся в данный момент команды можно использовать Ctrl + c . Также данное сочетание отменит редактирование командной строки и вернёт приглашение командной строки. Ctrl + l очищает экран.

3.Выполнение лабораторной работы

3.1 Перемещение по файловой системе

С помощью команды `pwd` узнаём полный путь к домашнему каталогу. (рис.1)

```
vagazizyanov@dk4n71 ~ $ cd /usr/local  
vagazizyanov@dk4n71 /usr/local $
```

(рис.1)

Переходим в каталог `local`- подкаталог `usr` корневого каталога указав абсолютный путь к нему. (рис.2)

```
vagazizyanov@dk4n71 ~ $ cd /usr/local  
vagazizyanov@dk4n71 /usr/local $
```

Рис.2

С помощью команды `cd` переходим в домашний каталог и выводим список файлов воспользовавшись командой `ls`. Затем убеждаемся, что файлы полученные от `ls` совпадают с файлами, отображающимися в графическом файловом менеджере. (рис.3 и рис. 4)

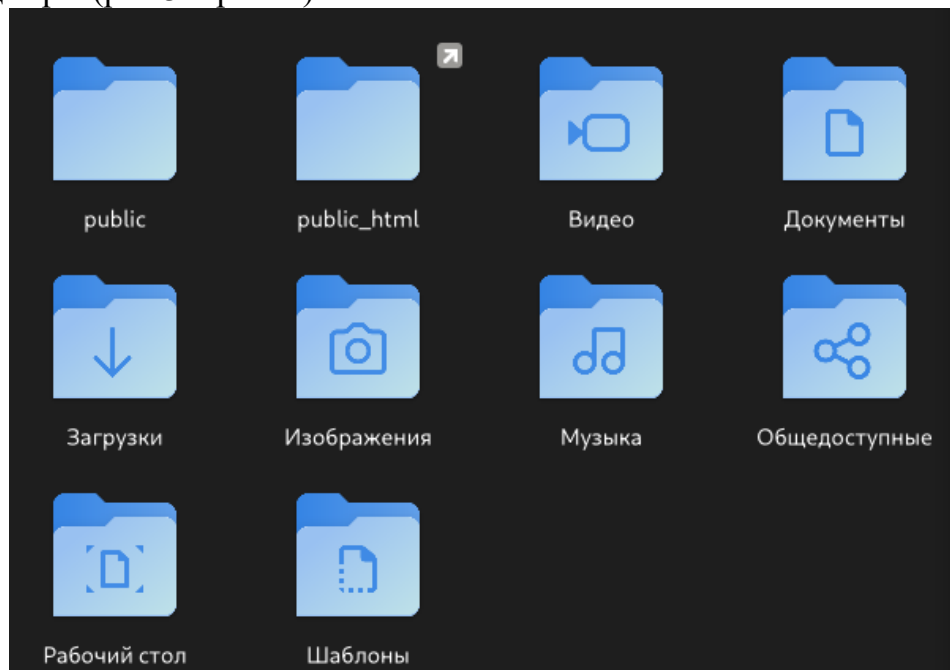


Рис 3

```
vagazizyanov@dk4n71 /usr/local $ cd ~  
vagazizyanov@dk4n71 ~ $ ls  
public      Видео      Загрузки   Музыка     'Рабочий стол'  
public_html Документы  Изображения  Общедоступные  Шаблоны  
vagazizyanov@dk4n71 ~ $
```

Рис 4

3.2 Создание пустых каталогов и файлов

Создаём, с помощью команды `mkdir`, в домашнем каталоге подкаталог с именем `parentdir` и затем ещё один подкаталог `dir` (рис 5)

```
vagazizyanov@dk4n71 ~ $ mkdir parentdir/dir
```

Рис 5

С помощью опции `-parents(-p)` создаём иерархическую цепочку подкаталогов, создавая все промежуточные каталоги. (рис.6)

```
vagazizyanov@dk4n71 ~ $ mkdir -p ~/newdir/dir1/dir2
```

Рис 6

Для создания файлов используем команду `touch` (рис 7)

```
vagazizyanov@dk4n71 ~ $ touch ~/newdir/dir1/dir2/test.txt
vagazizyanov@dk4n71 ~ $ ls ~/newdir/dir1/dir2
test.txt
```

Рис 7

3.3 Перемещение и удаление файлов или каталогов

Запросив подтверждение на удаление каждого файла в текущем каталоге, удаляем в подкаталоге `/newdir/dir1/dir2/` все файлы с именами, заканчивающимися на `.txt` (рис8)

```
vagazizyanov@dk4n71 ~ $ rm -R ~/newdir ~/parentdir/dir*
```

Рис 8

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

Создам файлы и каталоги в домашнем каталоге: (рис 9)

```
vagazizyanov@dk4n71 ~ $ mkdir -p parentdir1/dir1 parentdir2/dir2 parentdir3
vagazizyanov@dk4n71 ~ $ touch parentdir1/dir1/test1.txt parentdir2/dir2/test2.txt
```

Рис 9

Используя команды `cp` и `mv` файл `test1.txt` скопируем, а `test2.txt` перемещаем в каталог `parentdir3`: (рис 10)

```
vagazizyanov@dk4n71 ~ $ mv parentdir1/dir1/test1.txt parentdir3
vagazizyanov@dk4n71 ~ $ cp parentdir2/dir2/test2.txt parentdir3
```

Рис 10

С помощью `ls` проверяем правильность написания команд (рис 11)

```
vagazizyanov@dk4n71 ~ $ ls parentdir3
test1.txt test2.txt
vagazizyanov@dk4n71 ~ $ ls parentdir1/dir1
vagazizyanov@dk4n71 ~ $ ls parentdir2/dir2
test2.txt
```

Рис 11

Также команда `mv` может быть использована для переименования файлов и каталогов, а команда `cp` позволяет сделать копию файла с новым именем. Переименуем файл

test1.txt из каталога parentdir3 в newtest.txt, запрашивая подтверждение перед перезаписью: (рис 12)

```
vagazizyanov@dk4n71 ~ $ ls parentdir3
test1.txt test2.txt
vagazizyanov@dk4n71 ~ $ cp parentdir3/test2.txt parentdir3/subtest2.txt
vagazizyanov@dk4n71 ~ $ mv -i parentdir3/test1.txt parentdir3/newtest.txt
vagazizyanov@dk4n71 ~ $ ls parentdir3
newtest.txt subtest2.txt test2.txt
```

Рис 12

Переименуем каталог dir1 в каталоге parentdir1 в newdir: (рис 13)

Рис 13

4. Самостоятельная работа

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

2 Водим последовательность команд cd mkdir tmp cd tmp pwd cd /tmp pwd (рис 14)

```
vagazizyanov@dk4n71 ~/parentdir1 $ cd
vagazizyanov@dk4n71 ~ $ mkdir tmp
vagazizyanov@dk4n71 ~ $ cd tmp
vagazizyanov@dk4n71 ~/tmp $ pwd
/afs/.dk.sci.pfu.edu.ru/home/v/a/vagazizyanov/tmp
vagazizyanov@dk4n71 ~/tmp $ cd /tmp
vagazizyanov@dk4n71 /tmp $ pwd
/tmp
```

Рис 14

Pwd выдаёт разный результат, потому что мы переходим в два разных каталога tmp.

1. Когда вы выполняем команды:

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

Мы создаем новый каталог tmp в вашем домашнем каталоге и переходим в него. Команда pwd показывает путь к этому каталогу, который начинается с нашего домашнего каталога.

2. Когда мы выполняем команды:

```
cd /tmp
pwd
```

Мы переходим в каталог tmp, который является стандартным временным каталогом в системе Linux и находится в корневом каталоге (/). Команда pwd показывает

путь к этому каталогу, который начинается с корневого каталога.

3. Пользуясь командами `cd` и `ls`, смотрим содержимое корневого каталога, домашнего каталога (рис 15,16)

```
vagazizyanov@dk4n71 ~ $ ls /etc
a2ps          geoclue       mc             request-key.conf
acpi          ggi           mecabrc        request-key.d
adjtime       ggz.modules   mercurial       resolv.conf
afs.keytab    gimp          metalog.conf   revdep-rebuild
alsa          gnome-chess   mime.types     rhashrc
apache2       gnome-vfs-2.0 minicom         rmt
apparmor.d    gnome-vfs-mime-magic mke2fs.conf    rofi-pass.conf
appstream.conf GNUstep       mlocate-cron.conf rpc
ati           gpm           modprobe.conf  1100          rsyncd.conf
```

Рис 15

```
vagazizyanov@dk4n71 ~ $ ls /usr/local
bin  games  info  lib  lib32  lib64  man  sbin  share  src  texlive
```

Рис 16

Пользуясь изученными консольными командами, в своём домашнем каталоге создаём каталог `temp` и каталог `labs` с подкаталогами `lab1`, `lab2` и `lab3` одной командой (рис 17)

```
vagazizyanov@ONY-X:~$ mkdir -p ~/temp ~/labs/lab1 lab2 lab3
vagazizyanov@ONY-X:~$ ls
lab2  lab3  labs  temp
vagazizyanov@ONY-X:~$
```

Рис 17

В каталоге `temp` создаём файлы `text1.txt`, `text2.txt`, `text3.txt`. (рис 18)

```
vagazizyanov@ONY-X:~$ touch ~/temp/text3.txt
vagazizyanov@ONY-X:~$ touch ~/temp/text2.txt
vagazizyanov@ONY-X:~$ touch ~/temp/text1.txt
vagazizyanov@ONY-X:~$ ls temp
text1.txt  text2.txt  text3.txt
```

Рис 18

5. С помощью текстового редактора запишем в файл `text1.txt` свое имя, в файл `text2.txt` фамилию, в файл `text3.txt` учебную группу и выведем на экран содержимое файлов, используя команду `cat`. (рис 19)

```
vagazizyanov@ONY-X:~$ echo Владислав > ~/temp/text1.txt
vagazizyanov@ONY-X:~$ echo Газизянов > ~/temp/text2.txt
vagazizyanov@ONY-X:~$ echo НПИбд-01-23 > ~/temp/text3.txt
vagazizyanov@ONY-X:~$ cat ~/temp/text1.txt
Владислав
vagazizyanov@ONY-X:~$ cat ~/temp/text2.txt
Газизянов
vagazizyanov@ONY-X:~$ cat ~/temp/text3.txt
НПИбд-01-23
vagazizyanov@ONY-X:~$
```

Рис 19

6. `cp ~/temp/*.txt ~/labs/`

Эта команда копирует все файлы, имена которых

заканчиваются на .txt, из каталога ~/temp в каталог labs. После этого переименуем файлы каталога labs и переместим их: text1.txt переименуем в firstname.txt и переместим 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, убедимся, что все действия выполнены верно.(рис.20)

```
vagazizyanov@ONY-X:~/labs$ mv ~/labs/text2.txt ~/labs/lab2/lastname.txt
vagazizyanov@ONY-X:~/labs$ mv ~/labs/text3.txt ~/labs/lab3/id-group.txt
vagazizyanov@ONY-X:~/labs$ ls ~/labs/lab1
firstname.txt lab2
vagazizyanov@ONY-X:~/labs$ cat ~/labs/lab1/firstname.txt
Владислав
vagazizyanov@ONY-X:~/labs$ ls ~/labs/lab2
lastname.txt
vagazizyanov@ONY-X:~/labs$ cat ~/labs/lab2/lastname.txt
Газиззянов
vagazizyanov@ONY-X:~/labs$ ls ~/labs/lab3
id-group.txt
vagazizyanov@ONY-X:~/labs$ cat ~/labs/lab3/id-group.txt
НПИбд-01-23
vagazizyanov@ONY-X:~/labs$
```

Рис 20

7. Удаляем всё выше сделанное(рис 21)

```
vagazizyanov@ONY-X:~/labs$ rm -r ~/temp
vagazizyanov@ONY-X:~/labs$ rm -r ~/labs
```

Рис 21

5.Вопросы для самопроверки

1. Командная строка - это интерфейс, позволяющий взаимодействовать с операционной системой с помощью текстовых команд. Примеры команд: ls (список файлов и каталогов), cd (смена директории), rm (удаление файлов или каталогов).

2. Для получения информации о команде вы можете использовать команду man. Например, man ls покажет вам справочную информацию о команде ls.

3. Относительный путь указывает на файл или каталог относительно текущего рабочего каталога, в то время как абсолютный путь начинается от корневого каталога. Например, если вы находитесь в каталоге /home/user, то относительный путь к файлу в этом каталоге может быть file.txt, а абсолютный путь будет /home/user/file.txt.

4. Для определения абсолютного пути к текущей директории вы можете использовать команду pwd (print working directory).

5. Для удаления файла можно использовать команду

rm, а для удаления каталога - rm -r. Таким образом, одной и той же командой можно удалить и файлы, и каталоги.

6. Можно запустить несколько команд в одной строке, разделив их символом ;. Например: cd /tmp; ls.

7. Если использовать опцию -l в команде ls, то на экран будет выведена дополнительная информация о файлах и каталогах: тип файла, права доступа, количество ссылок, владелец, группа, размер, время последней модификации и имя.

8. Для отображения информации о скрытых файлах можно использовать опцию -a в команде ls. Например: ls -a.

9. Для автоматического дополнения вводимых команд служит клавиша Tab.

6.Вывод

Приобретение практических навыков работы с операционной системой на уровне командной строки важно для эффективного взаимодействия с компьютером. Это включает организацию файловой системы, навигацию по файловой системе, создание и удаление файлов и директорий. Овладев этими навыками, пользователь сможет более гибко и точно управлять своей операционной системой, что повысит его продуктивность и эффективность работы. Это также может открыть новые возможности для автоматизации задач и работы с большими объемами данных. В общем, эти навыки являются ценным дополнением к любому набору технических умений.