

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук
Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 01

дисциплина: Архитектура компьютера

Студент: Танатти Е.Г.

Группа: НФПбд-01-21

№ студ билета: 1032187272

МОСКВА

2024 г.

Оглавление

1. Цель работы	3
2. Задание	3
3. Теоретическое введение.....	3
4. Выполнение лабораторной работы.....	8
5. Выводы	11
Список литературы.....	12

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. Задание

1. Воспользовавшись командой `pwd`, узнайте полный путь к своей домашней директории.
2. Введите следующую последовательность команд `cd mkdir tmp cd tmp pwd cd /tmp pwd`. Объясните, почему вывод команды `pwd` при переходе в каталог `tmp` дает разный результат.
3. Пользуясь командами `cd` и `ls`, посмотрите содержимое корневого каталога, домашнего каталога, каталогов `/etc` и `/usr/local`.
4. Пользуясь изученными консольными командами, в своём домашнем каталоге создайте каталог `temp` и каталог `labs` с подкаталогами `lab1`, `lab2` и `lab3` одной командой. В каталоге `temp` создайте файлы `text1.txt`, `text2.txt`, `text3.txt`. Пользуясь командой `ls`, убедитесь, что все действия выполнены успешно (каталоги и файлы созданы).
5. С помощью любого текстового редактора (например, редактора `mc`) 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`.
6. 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.
7. Удалите все созданные в ходе выполнения лабораторной работы файлы и каталоги.

3. Теоретическое введение

3.1. 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 про-

грамм 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).

3.2. 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 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`.

3.3. 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` и т.п.

В табл. 3.1 приведено краткое описание нескольких каталогов.

Таблица 3.1. Описание некоторых каталогов файловой системы GNU Linux

Каталог	Описание
/	Корневая директория, содержащая всю файловую
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям (например: cat, ls, cp)
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя
/media	Точки монтирования для сменных носителей, таких как CD-ROM, DVD-ROM, flash
/root	Домашняя директория пользователя root
/tmp	Временные файлы
/usr	Вторичная иерархия для данных пользователя; содержит большинство пользовательских приложений и утилит, используемых в многопользовательском режиме; может быть смонтирована по сети только для чтения и быть общей для нескольких машин

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.

Базовые команды bash

В операционной системе GNU Linux взаимодействие пользователя с системой обычно осуществляется с помощью командной строки посредством построчного ввода команд. Общий формат команд можно представить следующим образом:

<имя_команды><разделитель><аргументы>

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
```

В таблице 3.2 приведены основные команды взаимодействия пользователя с файловой системой в GNU Linux посредством командной строки.

Таблица 3.2. Основные команды взаимодействия пользователя с файловой системой

Команда	Описание	
<code>pwd</code>	Print Working Directory	определение текущего каталога
<code>cd</code>	Change Directory	смена каталога
<code>ls</code>	LiSt	вывод списка файлов
<code>mkdir</code>	MaKe DIRectory	создание пустых каталогов
<code>touch</code>		создание пустых файлов
<code>rm</code>	ReMove	удаление файлов или каталогов
<code>mv</code>	MoVe	перемещение файлов и каталогов
<code>cp</code>	CoPy	копирование файлов и каталогов
<code>cat</code>		вывод содержимого файлов

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 сеан-

са. 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.

4. Выполнение лабораторной работы

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

```
etanatti@fedora:~$ pwd
/home/etanatti
etanatti@fedora:~$
```

2. Введите следующую последовательность команд

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

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

```
etanatti@fedora:~$ pwd
/home/etanatti
etanatti@fedora:~$ cd
etanatti@fedora:~$ mkdir tmp
etanatti@fedora:~$ cd tmp
etanatti@fedora:~/tmp$ pwd
/home/etanatti/tmp
etanatti@fedora:~/tmp$ cd /tmp
etanatti@fedora:/tmp$ pwd
/tmp
etanatti@fedora:/tmp$
```

Команда `pwd` означает рабочий каталог печати. При вызове команда печатает полный путь к текущему рабочему каталогу. Так как текущий рабочий каталог изменился, то и результат различен.

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


```

etanatti@fedora:~$ cd
etanatti@fedora:~$ ls
test  Видео      Загрузки      Музыка      'Рабочий стол'
tmp   Документы    Изображения    Общедоступные  Шаблоны
etanatti@fedora:~$ ls ~/test
Test1  test2

```

4. Пользуясь изученными консольными командами, в своём домашнем каталоге создайте каталог temp и каталог labs с подкаталогами lab1, lab2 и lab3 одной командой. В каталоге temp создайте файлы text1.txt, text2.txt, text3.txt. Пользуясь командой ls, убедитесь, что все действия выполнены успешно (каталоги и файлы созданы)

```

etanatti@fedora:~$ mkdir temp labs
etanatti@fedora:~$ ls
labs  test  Видео      Загрузки      Музыка      'Рабочий стол'
temp  tmp   Документы    Изображения    Общедоступные  Шаблоны
etanatti@fedora:~$ cd labs
etanatti@fedora:~/labs$ mkdir lab1 lab2 lab3
etanatti@fedora:~/labs$ ls
lab1  lab2  lab3
etanatti@fedora:~/labs$ cd ~/temp
etanatti@fedora:~/temp$ touch text1.txt text2.txt text3.txt
etanatti@fedora:~/temp$ ls
text1.txt  text2.txt  text3.txt
etanatti@fedora:~/temp$

```

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.

```

etanatti@fedora:~/temp$ ls
text1.txt  text2.txt  text3.txt
etanatti@fedora:~/temp$ cat text1.txt
Елена
etanatti@fedora:~/temp$ cat text2.txt
Танатти
etanatti@fedora:~/temp$ cat text3.txt
НФПбд-01-21
etanatti@fedora:~/temp$

```

```

etanatti@fedora:~/temp$ cat *.txt
Танатти
Елена
НФПбд-01-21
etanatti@fedora:~/temp$

```

6. 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.

```
etanatti@fedora:~/temp$ cd ~/labs
etanatti@fedora:~/labs$ ls
lab1 lab2 lab3 tex2.txt text1.txt text3.txt
etanatti@fedora:~/labs$ mv ~/labs/text1.txt ~/labs/lab1/firstname.txt
etanatti@fedora:~/labs$ mv ~/labs/tex2.txt ~/labs/lab2/lastname.txt
etanatti@fedora:~/labs$ mv ~/labs/text3.txt ~/labs/lab3/id_group.txt
mv: не удалось выполнить stat для '/home/etanatti/labs/text3.txt': Нет
etanatti@fedora:~/labs$ mv ~/labs/text3.txt ~/labs/lab3/id_group.txt
mv: невозможно переместить '/home/etanatti/labs/text3.txt' в '/home/eta
xt': Нет такого файла или каталога
etanatti@fedora:~/labs$ mv ~/labs/text3.txt ~/labs/lab3/id_group.txt
etanatti@fedora:~/labs$ ls
lab1 lab2 lab3
```

```
etanatti@fedora:~/labs$ ls
lab1 lab2 lab3
etanatti@fedora:~/labs$ cd lab1
etanatti@fedora:~/labs/lab1$ ls
firstname.txt
etanatti@fedora:~/labs/lab1$ cat *.txt
Елена
etanatti@fedora:~/labs/lab1$ cd ~/labs
etanatti@fedora:~/labs$ cd lab2
etanatti@fedora:~/labs/lab2$ ls
lastname.txt
etanatti@fedora:~/labs/lab2$ cat *.txt
Танатти
etanatti@fedora:~/labs/lab2$ cd ~/labs
etanatti@fedora:~/labs$ cd lab3
etanatti@fedora:~/labs/lab3$ ls
id_group.txt
etanatti@fedora:~/labs/lab3$ cat *.txt
НФБд-01-21
etanatti@fedora:~/labs/lab3$
```

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

```
etanatti@fedora:~$ ls -R labs
labs:
lab1 lab2 lab3

labs/lab1:
firstname.txt

labs/lab2:
lastname.txt

labs/lab3:
id_group.txt
etanatti@fedora:~$ rm -r labs temp
etanatti@fedora:~$ ls -r
Шаблоны      Общедоступные  Изображения  Документы  test
'Рабочий стол' Музыка        Загрузки     Видео
```

5. Выводы

В результате выполнения работы были приобретены практические базовые навыки работы с ОС на уровне командной строки (организация файловой системы, навигация по файловой системе, создание и удаление файлов и директорий).

Также сделала выводы о высоком уровне сложности установки данной – сама работа с терминалом довольно проста и понятна

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).