

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2.

СИСТЕМА КОНТРОЛЯ ВЕРСИЙ GIT

дисциплина: Архитектура компьютера

Студент: lianzeu

Группа: НКАбд-04-24

МОСКВА

2024 г.

Содержание

1. Цель работы	4
2. Теоретическое введение	错误!未定义书签。
2.1. Введение в GNU Linux	错误!未定义书签。
2.2. Введение в командную строку GNU Linux.....	错误!未定义书签。
2.3. Файловая структура GNU Linux: каталоги и файлы	错误!未定义书签。
2.4. Базовые команды bash	错误!未定义书签。
2.5. Полезные комбинации клавиш	错误!未定义书签。
3. Порядок выполнения работы	5
3.1. Техническое обеспечение.....	5
3.2. Перемещение по файловой системе.....	5
3.3 Создание пустых каталогов и файлов	5
3.4 Перемещение и удаление файлов или каталогов	6
3.5 Команда cat: вывод содержимого файлов	8
4. Задание для самостоятельной работы.....	12
5. Содержание отчёта.....	13
6. Вопросы для самопроверки	14

7. Заключение	16
Список литературы	17

2. Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git.

2.3. Техническое обеспечение

Лабораторная работа подразумевает выполнение настройки и работы с системой контроля версий Git (<https://git-scm.com/>). Выполнение работы возможно как в дисплейном классе факультета физико-математических и естественных наук РУДН, так и дома. Описание выполнения работы приведено для дисплейного класса со следующими характеристиками техники:

- Intel Core i3-550 3.2 GHz, 4 GB оперативной памяти, 8 GB свободного места на жёстком диске;
- ОС Linux Gentoo (<http://www.gentoo.ru/>).

2.4. Порядок выполнения работы

2.4.1. Настройка github

Существует несколько доступных серверов репозитория с возможностью бесплатного размещения данных. Например, <http://bitbucket.org/>, <https://github.com/> и <https://gitflic.ru>. Для выполнения лабораторных работ предлагается использовать Github.

Создайте учётную запись на сайте <https://github.com/> и заполните основные данные.

2.4.2. Базовая настройка git

Сначала сделаем предварительную конфигурацию git. Откройте терминал и введите следующие команды, указав имя и email владельца

репозитория:

```
git config --global user.name "<Name Surname>"
```

```
git config --global user.email "<work@mail>"
```

Настроим utf-8 в выводе сообщений git:

```
git config --global core.quotepath false
```

Зададим имя начальной ветки (будем называть её master):

```
git config --global init.defaultBranch master
```

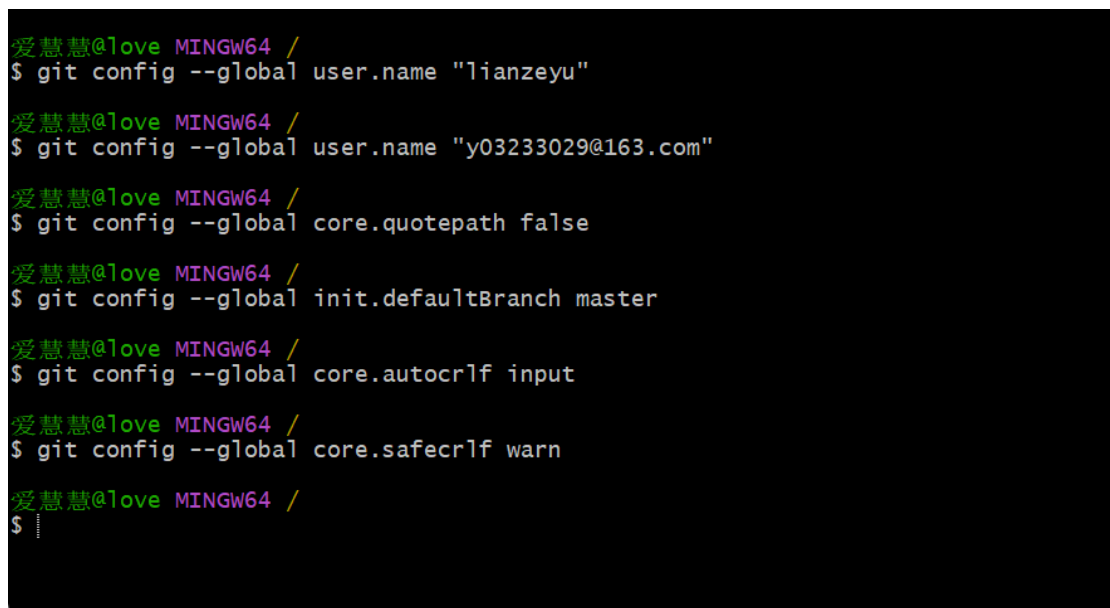
Параметр autocrlf:

```
git config --global core.autocrlf input
```

Параметр safecrlf:

```
git config --global core.safecrlf warn
```

Результат выполнения команд показан на рисунке 2.1



```
爱慧慧@love MINGW64 /  
$ git config --global user.name "lianzeyu"  
  
爱慧慧@love MINGW64 /  
$ git config --global user.name "y03233029@163.com"  
  
爱慧慧@love MINGW64 /  
$ git config --global core.quotepath false  
  
爱慧慧@love MINGW64 /  
$ git config --global init.defaultBranch master  
  
爱慧慧@love MINGW64 /  
$ git config --global core.autocrlf input  
  
爱慧慧@love MINGW64 /  
$ git config --global core.safecrlf warn  
  
爱慧慧@love MINGW64 /  
$ .....
```

Рисунок 2.1 – Выполнение списка команд

2.4.3. Создание SSH ключа

Для последующей идентификации пользователя на сервере репозитория необходимо сгенерировать пару ключей (приватный и

открытый):

ssh-keygen -C "Имя Фамилия <work@mail>" (рисунок 2.2.)

```
$ ssh-keygen -C "lianzeyu y03233029@163.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/███/.ssh/id_ed25519): lzy
Enter passphrase for "lzy" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in lzy
Your public key has been saved in lzy.pub
The key fingerprint is:
SHA256:Sq5s5Q00jvPp+3sJXDnx9xP7ZB9rGiUJJJuANUK46qQ lianzeyu y03233029@163.com
The key's randomart image is:
+--[ED25519 256]--+
|      .oo+ . .      |
|      . . + =      |
|      * .          |
|      + . o o      |
|      . . . S . . + +|
|      o .o=.      *.o|
|+   o =o+ . .   =+|
|E.o.ooo +      .oo|
|      =Booo      .o|
+-----[SHA256]-----+
爱慧慧@love MINGW64 /
$ |
```

Рисунок 2.2 – Создание SSH ключа

Ключи сохраняться в каталоге ~/.ssh/.

Далее необходимо загрузить сгенерённый открытый ключ. Для этого зайти на сайт <http://github.org/> под своей учётной записью и перейти в меню Setting . После этого выбрать в боковом меню SSH and GPG keys и нажать кнопку New SSH key . Скопировав из локальной консоли ключ в буфер обмена

```
cat ~/.ssh/id_rsa.pub | xclip -sel clip
```

вставляем ключ в появившееся на сайте поле и указываем для ключа имя (Title). Результат выполнения показан на рисунке 2.3.

Add new SSH Key

Title

Key type

Authentication Key ▾

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQGDt5N4oEwdOPNHneGdfDrQLfeOt15ezHmNn/pncOv2LB8jOYx+A0tYNIqMbb1S4Vx+r6VJkldxXO+el2vW8
ol6CTmH0eTAisiRuiPRo3V99L0ybMv+jSY6DpLZJyrVYAIWIK5KUapLuRlj0MEg7M6uxWvFMatl1rLwXPfQUTlQU4rnPIYCV9q9YQxLu7bD4rMI5XIHOZ74
3O8te2JULJQojGopAYWKA62fDuse/CAH6xVDg4sWR9bUYfBr5Ajmp4uwHagP7qcjWDxhSxb33sD0CDTz5FH1XZ3spXu8AzSP4B4A7gWi7CcOWZ15Pv
d7n//XNbqMoGQw6TLvjr0fyY9nlprwU8WgVg7xvAqYTNhvXZs4L7DN89wQN1PRbU160xMKonixYj+mZfBpSNgAfXS+tNjLZFaNC/aYmZFbfCHYiZfpK
3cAG2/PB8L4iSvql0aX8i9rK55O3anTiAfoDQjZ+qdX3ZQ+OWMv7uuXbanh1ts0AuuARsNOTJ/8ngHj+Tp8= y03233029@163.com
```

Add SSH key

Рисунок 2.3 – Добавление ключа

2.4.4. Сознание рабочего пространства и репозитория курса на основе шаблона

При выполнении лабораторных работ следует придерживаться структуры рабочего пространства. Рабочее пространство по предмету располагается в следующей иерархии:

~/work/study/

└ < учебный год >/

└ < название предмета >/

└ < код предмета >/

Например, для 2023–2024 учебного года и предмета «Архитектура компьютера» (код предмета arch-pc) структура каталогов примет следующий вид:

~/work/study/

└ 2023–2024/

└─ архитектура компьютера/

└─ arch-pc/

└─ labs/

└─ lab01/

└─ lab02/

└─ lab03/

...

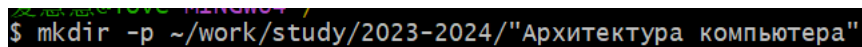
- Каталог для лабораторных работ имеет вид labs.
- Каталоги для лабораторных работ имеют вид lab<номер>, например:

lab01, lab02 и т.д. название проекта на хостинге git имеет вид:
study_<учебный год>_<код предмета>

Например, для 2023–2024 учебного года и предмета «Архитектура компьютера» (код предмета arch-pc) название проекта примет следующий вид: study_2023–2024_arch-pc

Откройте терминал и создайте каталог для предмета «Архитектура компьютера»:

```
mkdir -p ~/work/study/2023-2024/"Архитектура компьютера"
```



```
$ mkdir -p ~/work/study/2023-2024/"Архитектура компьютера"
```

Рисунок 2.4 – Создание каталогов

2.4.5. Создание репозитория курса на основе шаблона

Репозиторий на основе шаблона можно создать через web-интерфейс github.

Перейдите на страницу репозитория с шаблоном курса <https://github.com/yamadharma/course-directory-student-template>. Далее

выберите Use this template.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Repository template

 yamadharma/course-directory-student-template ▾

Start your repository with a template repository's contents.

☐ Include all branches

Copy all branches from yamadharma/course-directory-student-template and not just the default branch.

Owner *

 15374773029 ▾

Repository name *

/ study_2023-2024_arhpc

✓ Your new repository will be created as study_2023-2024_arhpc.

The repository name can only contain ASCII letters, digits, and the characters ., -, and _.

Great repository names are short and memorable. Need inspiration? How about **sturdy-spork** ?

Description (optional)

Рисунок 2.5 – Выбор шаблона

В открывшемся окне задайте имя репозитория (Repository name) study_2023-2024_arhpc и создайте репозиторий (кнопка Create repository from template).

Откройте терминал и перейдите в каталог курса: `cd ~/work/study/2023-2024/"Архитектура компьютера"` клонируйте созданный репозиторий:

```
git clone --recursive git@github.com:<user_name>/study_2023-2024_arh-pc.git
arch-pc
```

Ссылку для клонирования можно скопировать на странице созданного репозитория Code -> SSH:

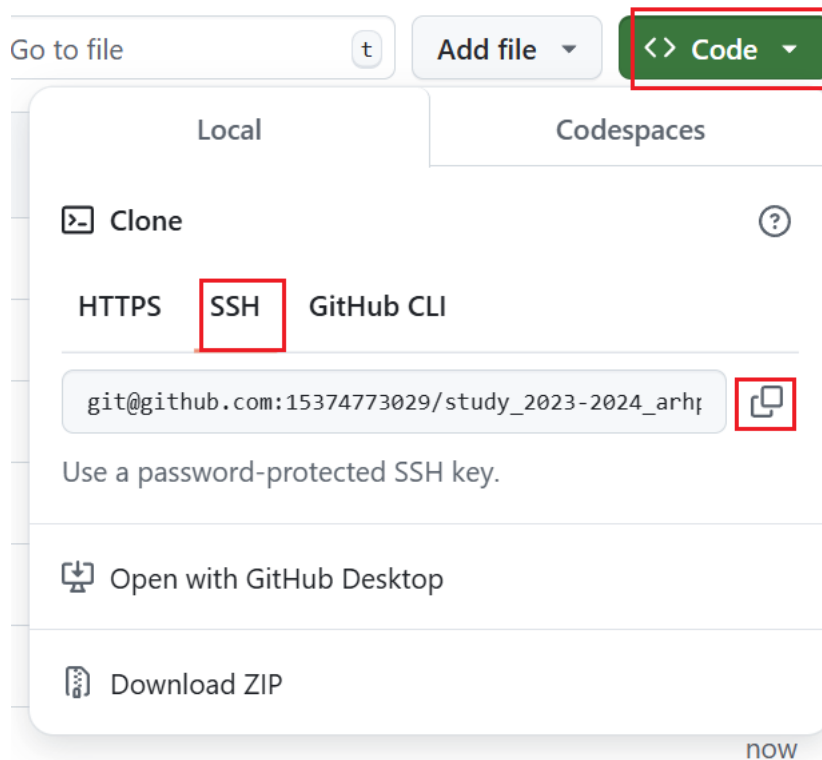


Рисунок 2.2 – Клонирование репозитория

4. Задание для самостоятельной работы

- 1) Создайте отчет по выполнению лабораторной работы в соответствующем каталоге рабочего пространства (labs>lab02>report).
- 2) Скопируйте отчеты по выполнению предыдущих лабораторных работ в соответствующие каталоги созданного рабочего пространства.
- 3) Загрузите файлы на github.

5. Содержание отчёта

Отчёт должен включать:

- Титульный лист с указанием номера лабораторной работы и ФИО студента.

- Формулировка цели работы.
- Описание результатов выполнения лабораторной работы:
 - описание выполняемого задания;
 - скриншоты (снимки экрана), фиксирующие выполнение заданий лабораторной работы;
 - комментарии и выводы по результатам выполнения заданий.
- Описание результатов выполнения заданий для самостоятельной работы:
 - описание выполняемого задания;
 - скриншоты (снимки экрана), фиксирующие выполнение заданий;
 - комментарии и выводы по результатам выполнения заданий.
- Выводы, согласованные с целью работы.

Отчёт по выполнению лабораторной работы оформляется в любом текстовом процессоре (OpenOffice, Libreoffice и др.) с последующей конвертацией в формат pdf.

6. Вопросы для самопроверки

1. Дайте определение командной строки. Приведите примеры.

Командная строка — это текстовый интерфейс пользователя для взаимодействия с операционной системой компьютера и/или другим программным обеспечением с помощью команд, вводимых с клавиатуры. Терминал в linux, командная строка windows.

2. Как получить информацию об интересующей вас команде?

Ввести команду с параметром `-help`.

3. Чем относительный путь к файлу отличается от абсолютного?

Относительный путь – путь от текущего каталога. Абсолютный путь – путь от корневого каталога.

4. Как определить абсолютный путь к текущей директории?

С помощью команды `pwd`.

5. При помощи каких команд можно удалить файл и каталог? Можно ли это сделать одной и той же командой?

Удалить файл и каталог можно командой `rm` с параметром `-R`. Удалять каталоги можно командой `rmdir`, удалять файлы можно командой `rm`.

6. Как можно запустить нескольких команд в одной строке?

Приведите примеры.

С помощью логических операторов `&&` (И), `||` (ИЛИ), или точки с запятой. При логическом И, команды выполняются последовательно если предыдущая выполнена успешно. При логическом ИЛИ, команды выполняются последовательно если предыдущая команда не выполнилась. При точке с запятой команды выполняются последовательно, независимо от результата предыдущей.

Примером приведу создание и удаление папки (рисунок 6.1).

```
bash-5.2$ mkdir lotus; rm -R lotus; mkdir lotus && rm -R lotus
```

Рисунок 6.1 – Одновременное создание и удаление папки lotus одной командой

7. Какая информация выводится на экран о файлах и каталогах, если используется опция

-l в команде ls?

Выводится дополнительная информация о файлах (права доступа, владельцы и группы, размеры файлов и время последнего доступа).

8. Каким образом отобразить информацию о скрытых файлах? Приведите примеры.

С помощью команды ls -a (рисунок 6.2).

```
bash-5.2$ ls -a
.
..
.bash_history
.cache
.config
.local
.vboxclient-clipboard-tty2-control.pid
.vboxclient-clipboard-tty2-service.pid
.vboxclient-draganddrop-tty2-control.pid
.vboxclient-draganddrop-tty2-service.pid
.vboxclient-hostversion-tty2-control.pid
.vboxclient-seamless-tty2-control.pid
.vboxclient-seamless-tty2-service.pid
Видео
Документы
Загрузки
Изображения
Музыка
Общедоступные
```

Рисунок 6.2 – Вывод скрытых файлов в домашнем каталоге

9. Какая клавиша или комбинация клавиш служит для автоматического дополнения вводимых команд?

Клавиша TAB автоматически дополняет введенную команду.

7. Заключение

Я приобрел практические навыки работы с операционной системой на уровне командной строки (организация файловой системы, навигация по файловой системе, создание и удаление файлов и директорий).

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).