

# **Отчет по лабораторной работе № 2**

**дисциплина: Архитектура компьютера**

Самойлова Софья Дмитриевна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Теоретическое введение</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>4</b>	<b>Выводы</b>	<b>13</b>
	<b>Список литературы</b>	<b>14</b>

## Список иллюстраций

3.1	работа в терминале . . . . .	7
3.2	работа в терминале . . . . .	7
3.3	работа в терминале . . . . .	8
3.4	работа в терминале . . . . .	8
3.5	работа в терминале . . . . .	8
3.6	работа в терминале . . . . .	9
3.7	работа в терминале . . . . .	9
3.8	работа в терминале . . . . .	10
3.9	работа в терминале . . . . .	10
3.10	работа в терминале . . . . .	11
3.11	работа в терминале . . . . .	11
3.12	работа в терминале . . . . .	12
3.13	работа в терминале . . . . .	12
3.14	работа в терминале . . . . .	12

## **Список таблиц**

# 1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий, а также приобрести практические навыки по работе с системой git

## 2 Теоретическое введение

Системы контроля версий (VCS) используются для совместной работы над проектами. Обычно основное дерево проекта хранится в локальном или удалённом репозитории с доступом для участников. VCS фиксирует изменения, объединяет их и позволяет откатываться к предыдущим версиям. В классических системах применяется централизованная модель с единственным репозиторием, где сервер управляет версиями. Пользователь получает нужную версию файлов, вносит изменения и загружает новую версию, при этом старые версии сохраняются. Сервер может использовать дельта-компрессию, сохраняя только изменения между версиями, что позволяет уменьшить объём хранимых данных. Системы контроля версий предлагают гибкие функции, такие как поддержка нескольких версий одного файла с сохранением общей истории изменений и индивидуальных ветвей. Они фиксируют информацию о том, кто и когда вносил изменения, что хранится в журнале, доступ к которому можно ограничить. В распределённых системах центральный репозиторий не обязателен. Из классических VCS известны CVS и Subversion, а среди распределённых — Git, Bazaar и Mercurial. Принципы их работы схожи, различия в синтаксисе команд.

Система контроля версий Git представляет собой набор программ командной строки. Доступ к ним можно получить из терминала посредством ввода команды `git` с различными опциями. Благодаря тому, что Git является распределённой системой контроля версий, резервную копию локального хранилища можно сделать простым копированием или архивацией.

### 3 Выполнение лабораторной работы

1. Базовая настройка git 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. Создаем учётную запись на сайте <https://github.com/> и заполняем основные данные (рис.3.1).

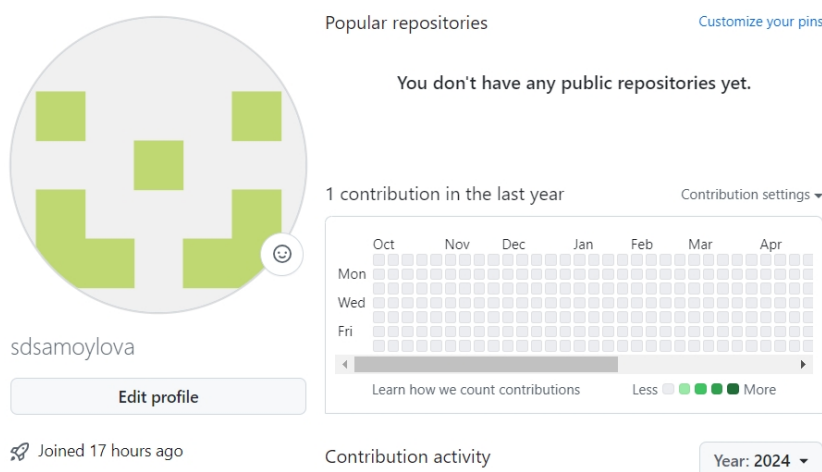


Рис. 3.1: работа в терминале

Сделаем предварительную конфигурацию git. Откроем терминал и введем следующие команды, указав имя и email владельца репозитория (рис.3.2).

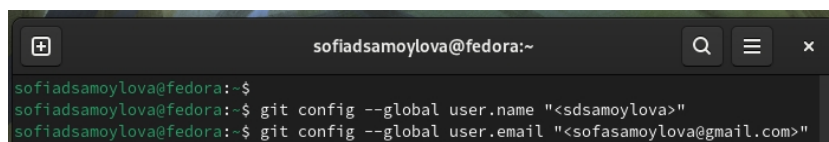
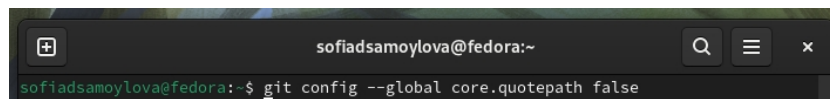


Рис. 3.2: работа в терминале

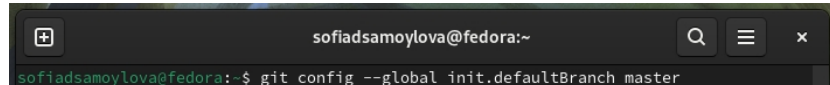
Настроим utf-8 в выводе сообщений git ((рис.3.3).

A terminal window with a dark background. The title bar shows 'sofiadsamoylova@fedora:~'. The command prompt is 'sofiadsamoylova@fedora:~\$' followed by the command 'git config --global core.quotePath false'.

```
sofiadsamoylova@fedora:~$ git config --global core.quotePath false
```

Рис. 3.3: работа в терминале

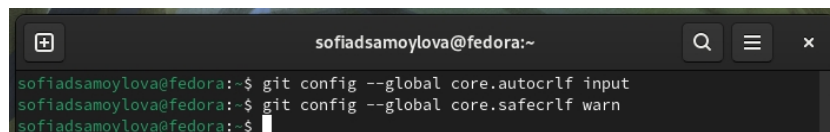
Зададим имя начальной ветки (будем называть её master) (рис.3.4).

A terminal window with a dark background. The title bar shows 'sofiadsamoylova@fedora:~'. The command prompt is 'sofiadsamoylova@fedora:~\$' followed by the command 'git config --global init.defaultBranch master'.

```
sofiadsamoylova@fedora:~$ git config --global init.defaultBranch master
```

Рис. 3.4: работа в терминале

Параметр autocrlf и safecrlf (рис.3.5).

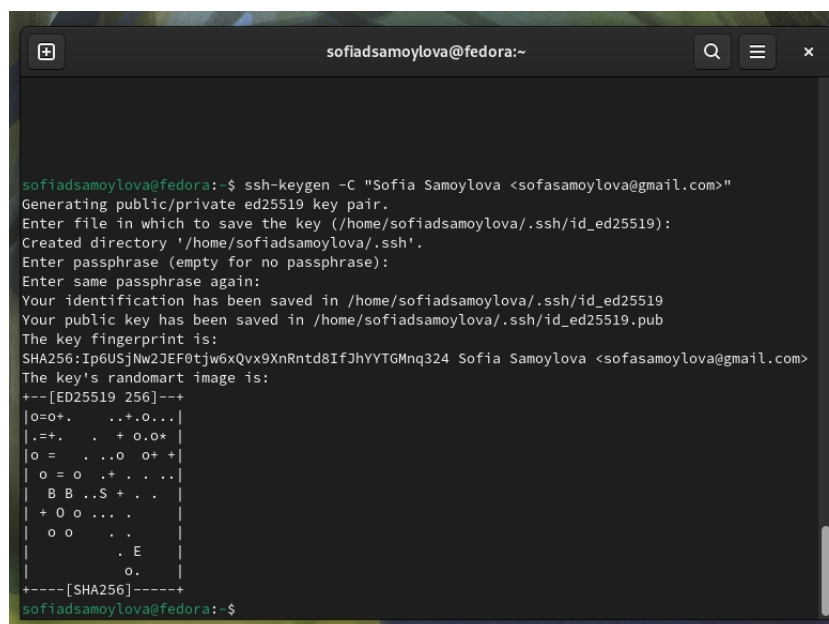
A terminal window with a dark background. The title bar shows 'sofiadsamoylova@fedora:~'. The command prompt is 'sofiadsamoylova@fedora:~\$' followed by two commands: 'git config --global core.autocrlf input' and 'git config --global core.safecrlf warn'.

```
sofiadsamoylova@fedora:~$ git config --global core.autocrlf input
sofiadsamoylova@fedora:~$ git config --global core.safecrlf warn
sofiadsamoylova@fedora:~$
```

Рис. 3.5: работа в терминале

2. Создание SSH ключа Для последующей идентификации пользователя на сервере репозитория необходимо сгенерировать пару ключей (приватный и открытый) (рис.3.6).

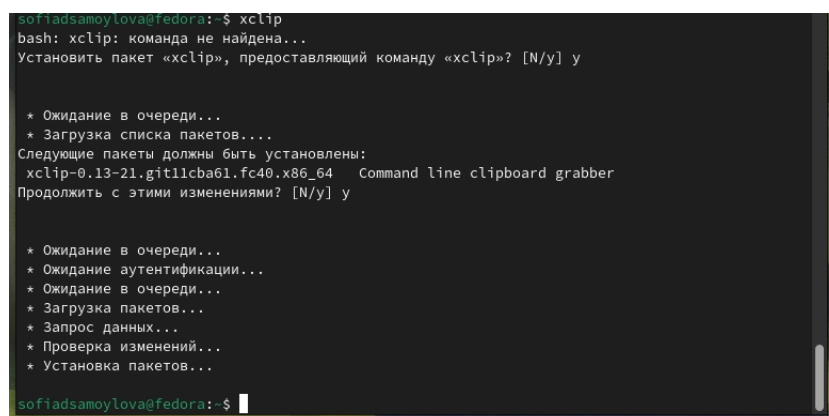




```
sofiadsamoylova@fedora:~$ ssh-keygen -C "Sofia Samoylova <sofasamoylova@gmail.com>"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/sofiadsamoylova/.ssh/id_ed25519):
Created directory '/home/sofiadsamoylova/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/sofiadsamoylova/.ssh/id_ed25519
Your public key has been saved in /home/sofiadsamoylova/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:Ip6USjNw2JEF0tjw6xQvx9XnRntd8IfJhYYTGMnq324 Sofia Samoylova <sofasamoylova@gmail.com>
The key's randomart image is:
+--[ED25519 256]--+
|o=o+. .+.o...|
|.=+. . +o.o* |
|o = . ..o o+ +|
| o = o .+ . . .|
| B B ..S + . .|
| + O o . . . .|
| o o . . . .|
| . E . . . .|
| o . . . .|
+-----[SHA256]-----+
sofiadsamoylova@fedora:~$
```

Рис. 3.6: работа в терминале

Xclip – утилита, позволяющая скопировать любой текст через терминал. Ока-  
зывается, в дистрибутиве Linux Kali ее сначала надо установить. Устанавливаю  
xclip с помощью команды apt-get install с ключом -у от имени суперпользователя,  
введя в начале команды sudo (рис.3.7).



```
sofiadsamoylova@fedora:~$ xclip
bash: xclip: команда не найдена...
Установить пакет «xclip», предоставляющий команду «xclip»? [N/y] y

* Ожидание в очереди...
* Загрузка списка пакетов...
Следующие пакеты должны быть установлены:
xclip-0.13-21.git11cba61.fc40.x86_64 Command line clipboard grabber
Продолжить с этими изменениями? [N/y] y

* Ожидание в очереди...
* Ожидание аутентификации...
* Ожидание в очереди...
* Загрузка пакетов...
* Запрос данных...
* Проверка изменений...
* Установка пакетов...
sofiadsamoylova@fedora:~$
```

Рис. 3.7: работа в терминале

Ключи сохраняются в каталоге ~/.ssh/. Далее необходимо загрузить сгенерённый  
открытый ключ. Для этого заходим на сайт <http://github.org/> под своей учётной  
записью и переходим 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 и вставляем ключ в появившееся на сайте поле и указываем для ключа имя (Title) (рис.3.8).

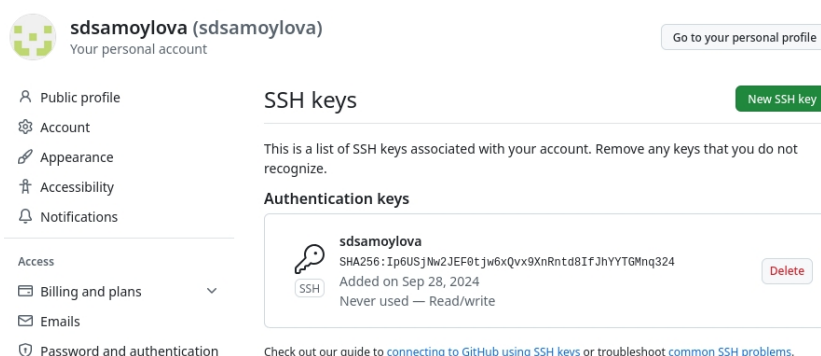


Рис. 3.8: работа в терминале

### 3. Создание рабочего пространства и репозитория курса на основе шаблона

При выполнении лабораторных работ следует придерживаться структуры рабочего пространства. Название проекта на хостинге git имеет вид: study\_\_ Например, для 2023–2024 учебного года и предмета «Архитектура компьютера» (код предмета arch-pc) название проекта примет следующий вид: study\_2023–2024\_arch-pc

Открываем терминал и создаем каталог для предмета «Архитектура компьюте-ра»:mkdir -p ~/work/study/2024-2025/“Архитектура компьютера” (рис.3.9).

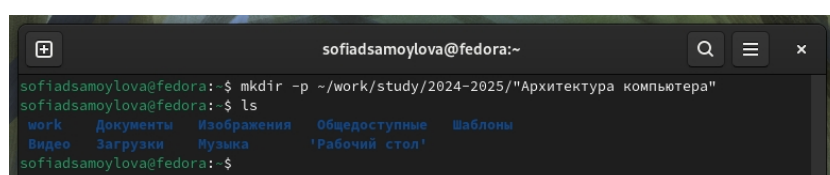


Рис. 3.9: работа в терминале

Репозиторий на основе шаблона можно создать через web-интерфейс github. Переходим на страницу репозитория с шаблоном курса <https://github.com/yamadharma/course-directory-student-template> (рис.3.10).

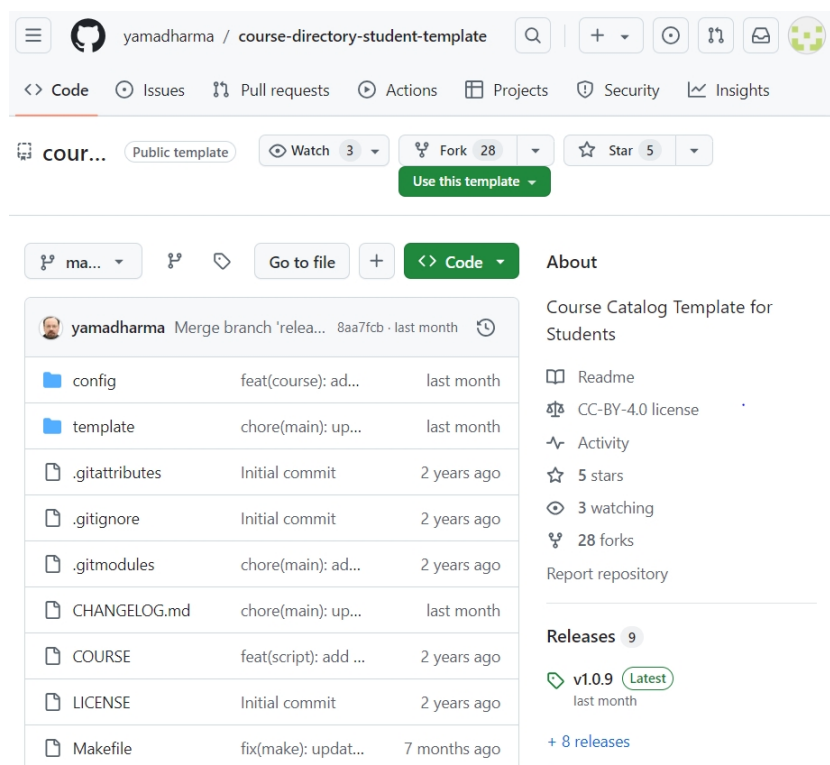


Рис. 3.10: работа в терминале

Далее выбираем Use this template.

В открывшемся окне задаем имя репозитория (Repository name) study\_2024–2025\_arch-  
pc и создаем репозиторий (кнопка Create repository from template).(рис.3.11).

### 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**

yamadharm/course-directory-student-template ▾

Start your repository with a template repository's contents.

☐ **Include all branches**  
Copy all branches from yamadharm/course-directory-student-template and not just the default branch.

---

**Owner \*** sdsamoylova ▾ / **Repository name \*** study\_2024–2025\_arch-pc

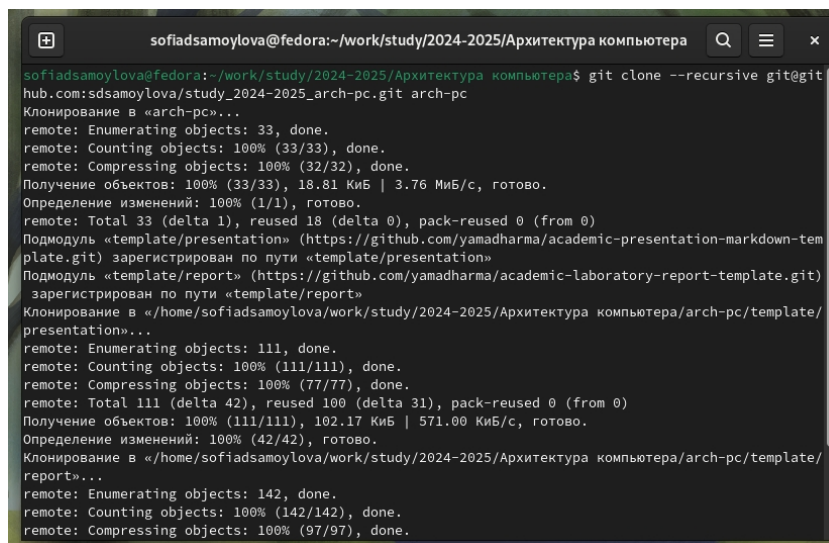
✔ Your new repository will be created as study\_2024-2025\_arch-pc.  
The repository name can only contain ASCII letters, digits, and the characters -, ., and \_.

Great repository names are short and memorable. Need inspiration? How about [friendly-spoon](#) ?

**Description** (optional)

Рис. 3.11: работа в терминале

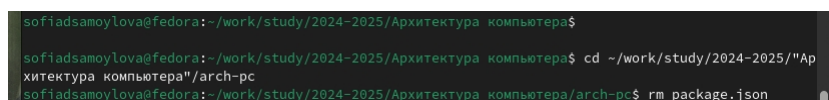
Открываем терминал, переходим в каталог курса и клонируем созданный репозиторий (рис.3.12).



```
sofiadsamoylova@fedora:~/work/study/2024-2025/Архитектура компьютера$ git clone --recursive git@github.com:sdsamoylova/study_2024-2025_arch-pc.git arch-pc
Клонирование в «arch-pc»...
remote: Enumerating objects: 33, done.
remote: Counting objects: 100% (33/33), done.
remote: Compressing objects: 100% (32/32), done.
Получение объектов: 100% (33/33), 18.81 КиБ | 3.76 МБ/с, готово.
Определение изменений: 100% (1/1), готово.
remote: Total 33 (delta 1), reused 18 (delta 0), pack-reused 0 (from 0)
Подмодуль «template/presentation» (https://github.com/yamadharma/academic-presentation-markdown-template.git) зарегистрирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-report-template.git) зарегистрирован по пути «template/report»
Клонирование в «/home/sofiadsamoylova/work/study/2024-2025/Архитектура компьютера/arch-pc/template/presentation»...
remote: Enumerating objects: 111, done.
remote: Counting objects: 100% (111/111), done.
remote: Compressing objects: 100% (77/77), done.
remote: Total 111 (delta 42), reused 100 (delta 31), pack-reused 0 (from 0)
Получение объектов: 100% (111/111), 102.17 КиБ | 571.00 КиБ/с, готово.
Определение изменений: 100% (42/42), готово.
Клонирование в «/home/sofiadsamoylova/work/study/2024-2025/Архитектура компьютера/arch-pc/template/report»...
remote: Enumerating objects: 142, done.
remote: Counting objects: 100% (142/142), done.
remote: Compressing objects: 100% (97/97), done.
```

Рис. 3.12: работа в терминале

4. Настройка каталога курса Переходим в каталог курса и удаляем лишние файлы (рис.3.13).



```
sofiadsamoylova@fedora:~/work/study/2024-2025/Архитектура компьютера$
sofiadsamoylova@fedora:~/work/study/2024-2025/Архитектура компьютера$ cd ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc
sofiadsamoylova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ rm package.json
```

Рис. 3.13: работа в терминале

Создаем необходимые каталоги (рис.3.14).



```
sofiadsamoylova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ echo arch-pc > COURSE
```

Рис. 3.14: работа в терминале

## 4 Выводы

При выполнении данной лабораторной работы я изучила идеологию и применение средств контроля версий, а также приобрела практические навыки по работе с системой git.

## 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](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).