

# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

## ОТЧЕТ

### ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2 \_\_\_\_\_

дисциплина:     Архитектура компьютера

Студент: Савенкова А.Е.

Группа: НММбд-03-24

МОСКВА

2024\_\_ г.

## Содержание

1. Цель работы .....	3
2. Задание .....	4
3. Теоретическое введение .....	5
4. Выполнение лабораторной работы .....	6
5. Выводы .....	13
6. Источники .....	14

## **1. Цель работы**

Цель данной лабораторной работы заключается в изучении идеологии и применении средств контроля версий, а также в приобретении практических навыков по работе с системой git.

## **2. Задание**

1. Настройка GitHub.
2. Базовая настройка Git.
3. Создание SSH-ключа.
4. Создание рабочего пространства и репозитория курса на основе шаблона.
5. Создание репозитория курса на основе шаблона.
6. Настройка каталога курса.
7. Выполнение заданий для самостоятельной работы

### 3. Теоретическое введение

Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом. Обычно основное дерево проекта хранится в локальном или удалённом репозитории, к которому настроен доступ для участников проекта.

В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов.

Среди распределённых VCS наиболее известны Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе 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.

Работа пользователя со своей веткой начинается с проверки и получения изменений из центрального репозитория. Затем можно вносить изменения в локальном дереве и/или ветке. После завершения внесения какого-то изменения в файлы и/или каталоги проекта необходимо разместить их в центральном репозитории. Для этого необходимо проверить, какие файлы изменились к текущему моменту.

Команда	Описание
<code>git init</code>	создание основного дерева репозитория
<code>git pull</code>	получение обновлений (изменений) текущего дерева из центрального репозитория
<code>git push</code>	отправка всех произведённых изменений локального дерева в центральный репозиторий
<code>git status</code>	просмотр списка изменённых файлов в текущей директории
<code>git diff</code>	просмотр текущих изменения
<code>git add .</code>	добавить все изменённые и/или созданные файлы и/или каталоги
<code>git add имена_файлов</code>	добавить конкретные изменённые и/или созданные файлы и/или каталоги
<code>git rm имена_файлов</code>	удалить файл и/или каталог из индекса репозитория (при этом файл и/или каталог остаётся в локальной директории)

Таблица основных команд `git`

## 4. Выполнение лабораторной работы

### 4.1. Настройка github

Сначала создаю учётную запись на сайте <https://github.com/>, заполняю основные данные (рис. 1.1).

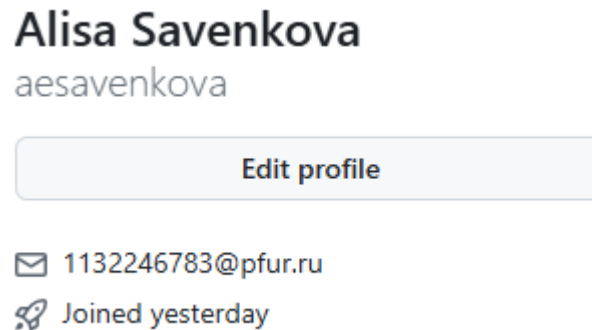


Рис. 1.1: Создание профиля на github

### 4.2. Базовая настройка git

Затем делаю предварительную конфигурацию git, открываю терминал и ввожу команды, указав свое имя и email (рис. 2.1).

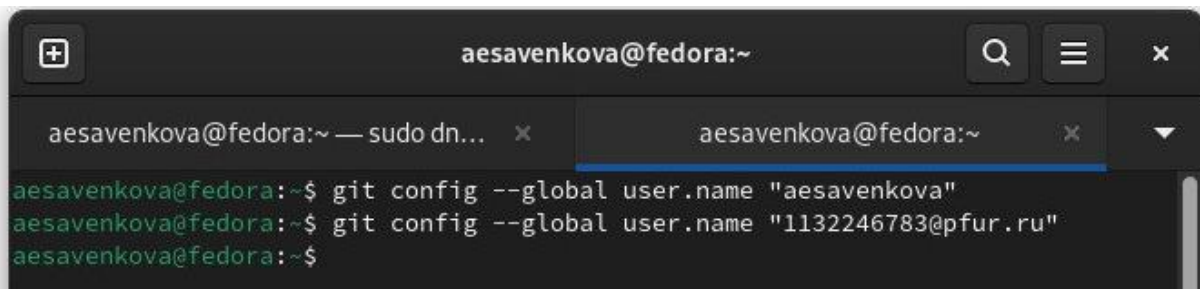


Рис. 2.1: Предварительная конфигурация git

Следующим шагом настраиваю utf-8 в выводе сообщений git (рис. 2.2).

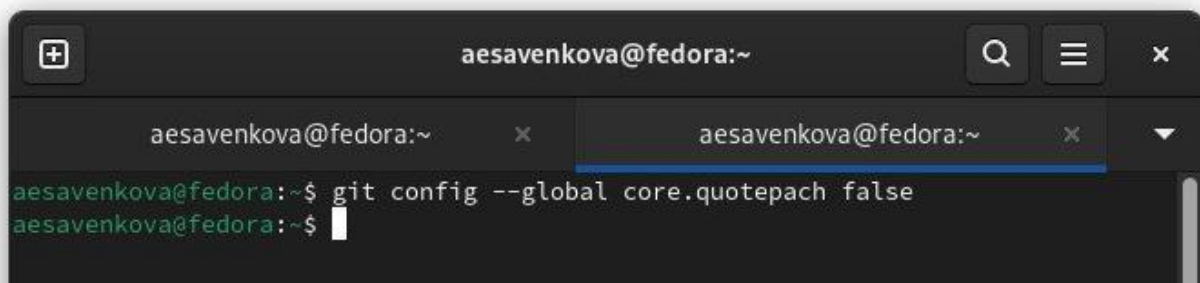
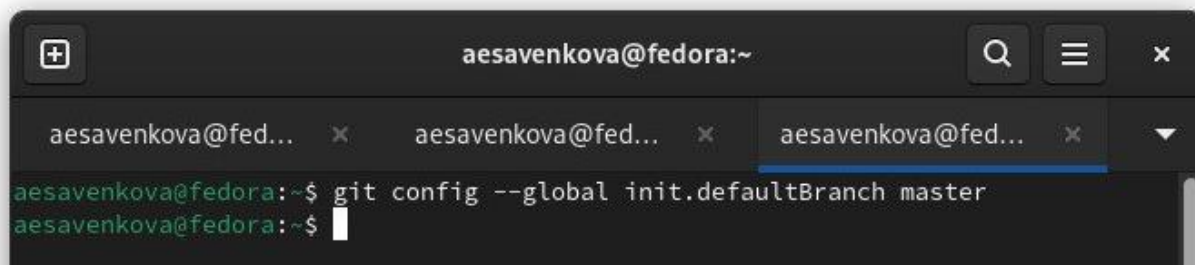


Рис. 2.2: Настройка кодировки вывода в консоль

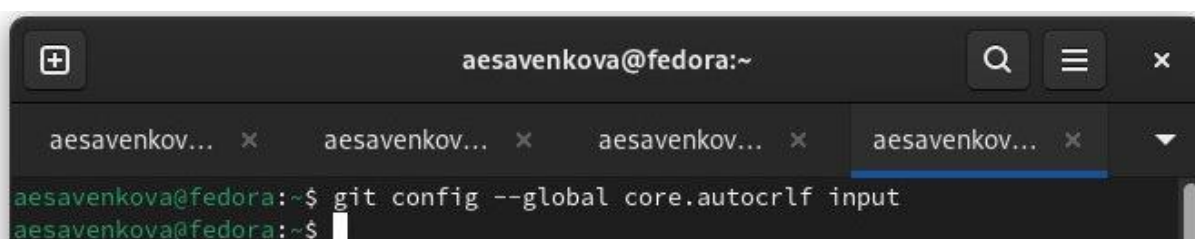
Далее задаю имя master для начальной ветки (рис. 2.3).

A terminal window titled 'aesavenkova@fedora:~' with three tabs. The active tab shows the command 'git config --global init.defaultBranch master' being entered at the prompt.

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

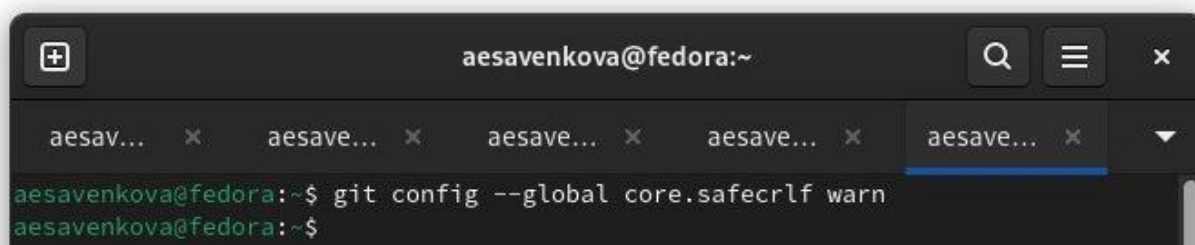
Рис. 2.3: Имя начальной ветки

Затем задаю параметры `autocrlf`, чтобы в главном репозитории все переводы строк текстовых файлов были одинаковы, и `safecrlf` для предотвращения автозамены LF в CRLF (рис. 2.4-2.5).

A terminal window titled 'aesavenkova@fedora:~' with four tabs. The active tab shows the command 'git config --global core.autocrlf input' being entered at the prompt.

```
aesavenkova@fedora:~$ git config --global core.autocrlf input
aesavenkova@fedora:~$
```

Рис. 2.4: Параметр `autocrlf`

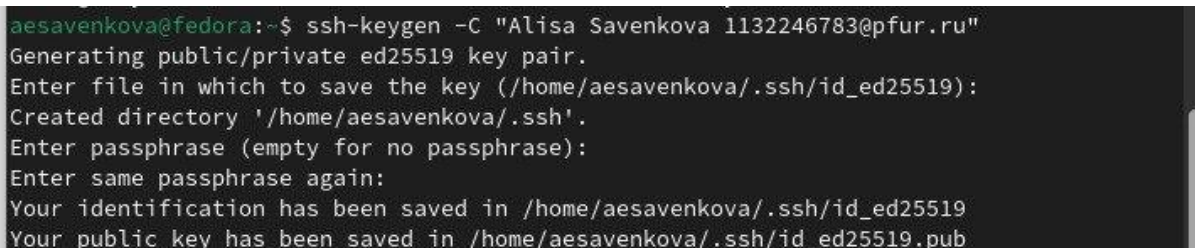
A terminal window titled 'aesavenkova@fedora:~' with five tabs. The active tab shows the command 'git config --global core.safecrlf warn' being entered at the prompt.

```
aesavenkova@fedora:~$ git config --global core.safecrlf warn
aesavenkova@fedora:~$
```

Рис. 2.5: Параметр `safecrlf`

### 4.3. Создание SSH ключа

Для последующей идентификации на сервере репозитория генерирую пару ключей (приватный и открытый) (рис. 3.1).

A terminal window showing the output of the 'ssh-keygen' command. It prompts for a file name, creates a directory, asks for a passphrase, and confirms the key generation.

```
aesavenkova@fedora:~$ ssh-keygen -C "Alisa Savenkova 1132246783@pfur.ru"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/aesavenkova/.ssh/id_ed25519):
Created directory '/home/aesavenkova/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/aesavenkova/.ssh/id_ed25519
Your public key has been saved in /home/aesavenkova/.ssh/id_ed25519.pub
```

Рис. 3.1: Генерация ssh-ключа

Далее загружаю сгенерённый открытый ключ. Для этого захожу на сайт <http://github.org/> под своей учётной записью и перехожу в меню **Settings**. После этого выбираю в боковом меню **SSH and GPG keys** и нажимаю кнопку **New SSH key**.

Скопировав из локальной консоли ключ в буфер обмена, вставляю ключ в появившееся на сайте поле и задаю ключу имя key1 (рис. 3.2-3.3).

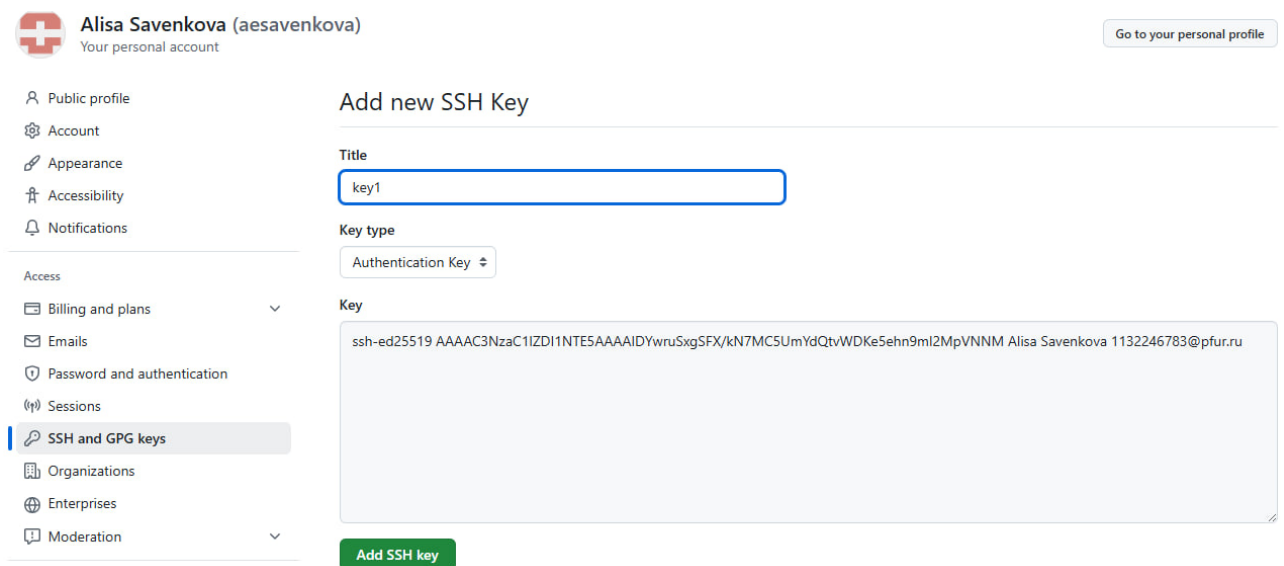


Рис. 3.2: Генерация ssh-ключа

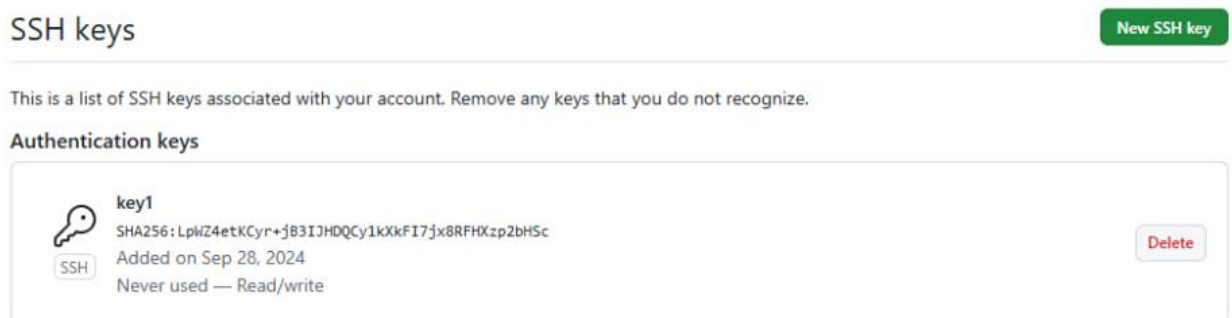


Рис. 3.3: Ключ в профиле на github

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

Открываю терминал и создаю каталог для 2024–2025 учебного года и предмета «Архитектура компьютера», с помощью ls проверяю выполнение команды (рис. 4.1).

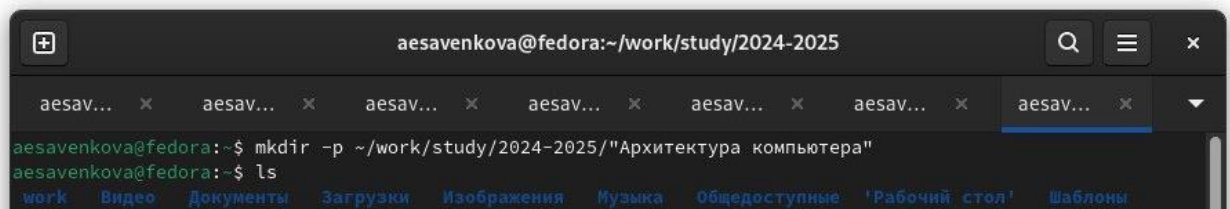


Рис. 4.1: Создание каталога «Архитектура компьютера»

#### 4.5. Создание репозитория курса на основе шаблона

Перехожу на страницу репозитория с шаблоном курса



<https://github.com/yamadharma/course-directory-student-template> (рис. 5.1).

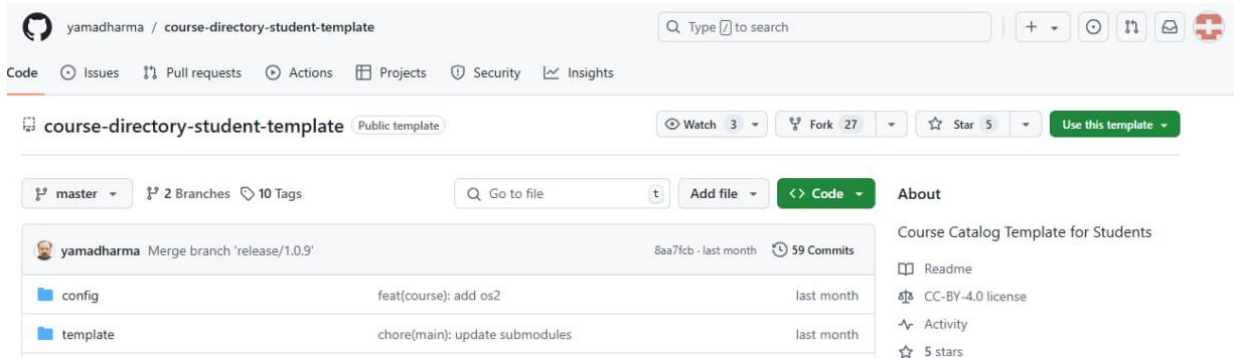


Рис. 5.1: Шаблон курса

В открывшемся окне задаю имя репозитория `study_2024-2025_arh-pc` и создаю репозиторий (рис. 5.2).

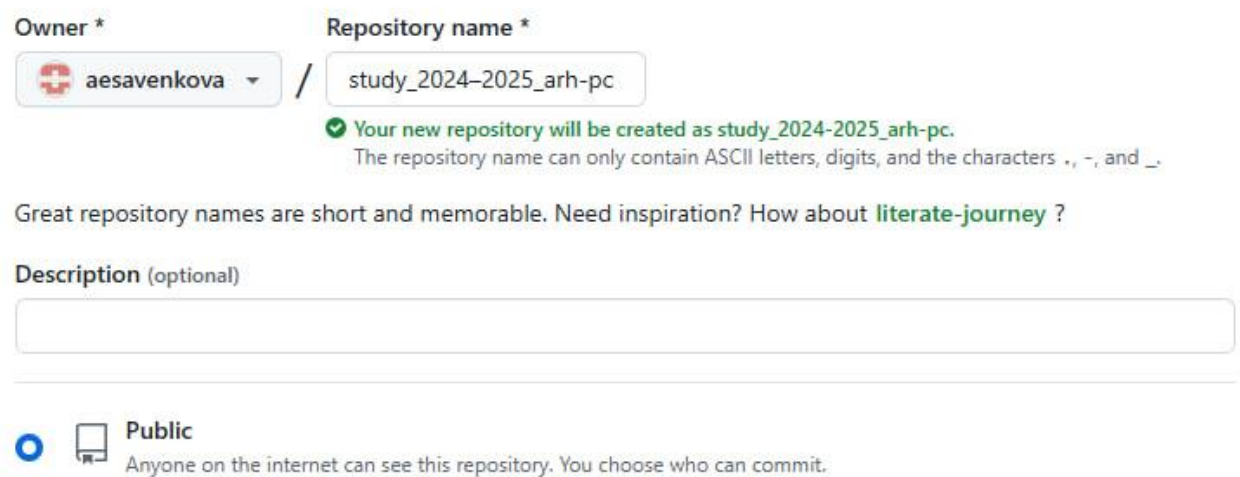


Рис. 5.2: Создание репозитория

Затем открываю терминал, перехожу в каталог курса и клонирую созданный репозиторий (рис. 5.3), перед этим скопировав Ссылку для клонирования на странице созданного репозитория (5.4).

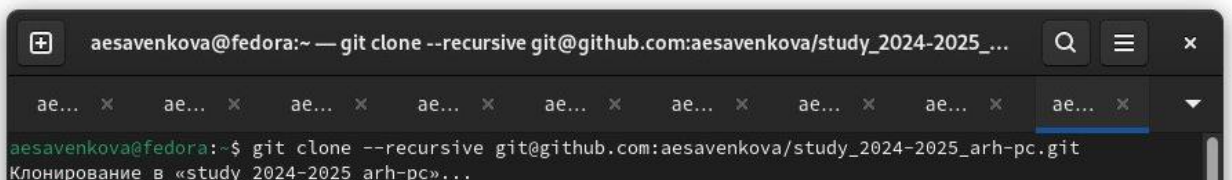


Рис. 5.3: Клонирование репозитория

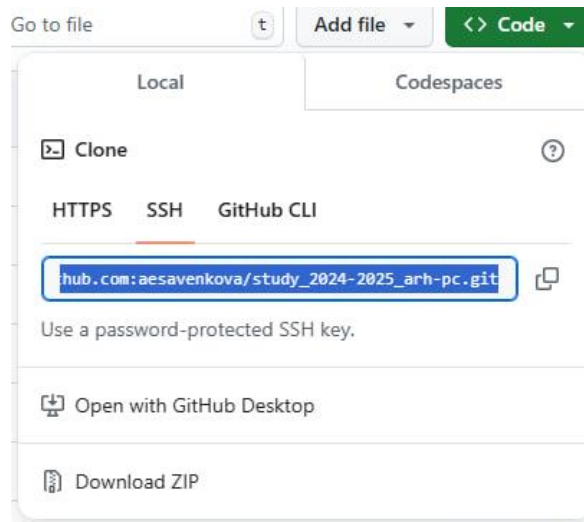


Рис. 5.4: Ссылка для клонирования

## 4.6. Настройка каталога курса

Перехожу в каталог курса и удаляю лишние файлы (рис. 6.1).

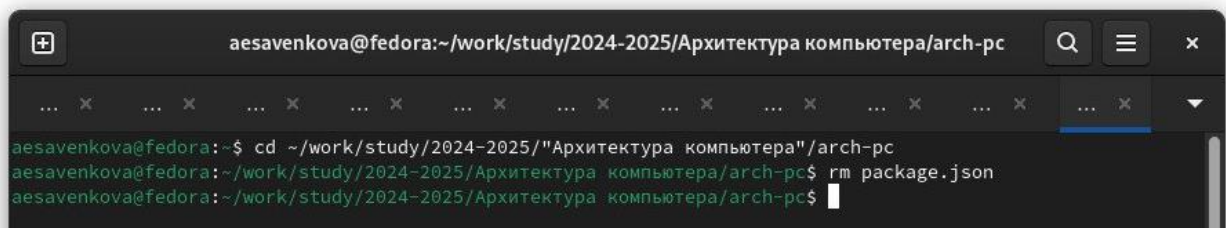


Рис. 6.1: Удаление файла package.json

Создаю необходимые каталоги (рис. 6.2).

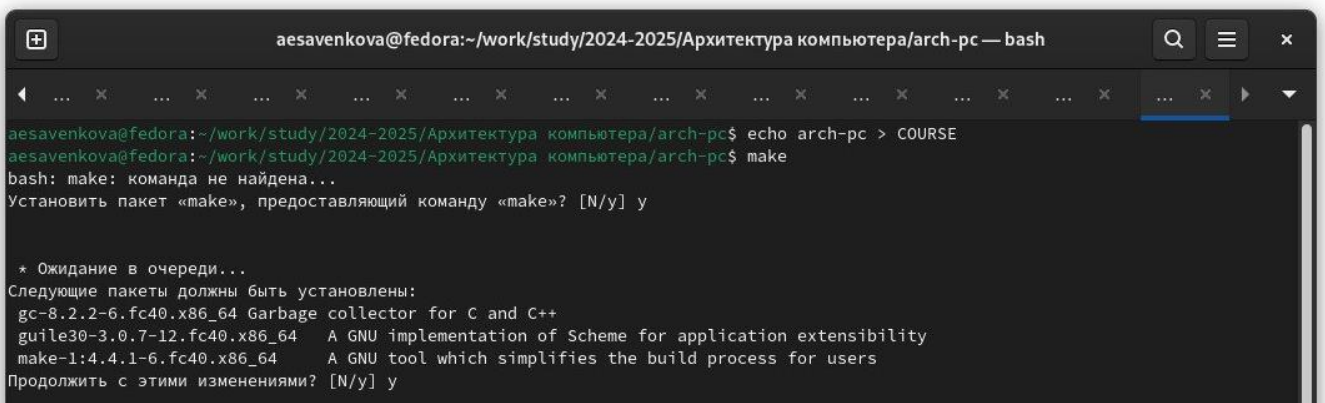


Рис. 6.2: Создание каталогов

Затем отправляю файлы на сервер (рис. 6.3)

```

aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git add .
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git commit -am 'feat(main): make course structure'
Текущая ветка: master
Ваша ветка опережает «origin/master» на 1 коммит.
(используйте «git push», чтобы опубликовать ваши локальные коммиты)

нечего коммитить, нет изменений в рабочем каталоге
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git push
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 282 байта | 282.00 КиБ/с, готово.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:aesavenkova/study_2024-2025_arh-pc.git
 1f75b46..7a3df13 master -> master

```

Рис. 6.3: Отправление файлов на сервер

Далее убеждаюсь в правильности создания иерархии рабочего пространства на странице github.

#### 4.7. Выполнение заданий для самостоятельной работы

Перехожу в директорию labs/lab02 с помощью cd. Создаю в каталоге файл для отчета по второй лабораторной работе, проверяю наличие файла (рис. 7.1).

```

aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs$ ls lab02
report Л02_Савенкова_отчет.doc
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs$

```

Рис. 7.1: Создание файла с отчетом

Копирую отчеты по выполнению предыдущих лабораторных работ в соответствующие каталоги созданного рабочего пространства и проверяю с помощью команды ls (рис. 7.2).

```

aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch...
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs$ ls
lab01 lab02
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs$ ls lab01
Л01_Савенкова_отчет.pdf
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs$ ls lab02
report Л02_Савенкова_отчет.doc
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs$

```

Рис. 7.2: Отчеты по лабораторным работам

Загружаю файл с отчетом по предыдущей лабораторной работе на github (рис. 7.3).

```
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git add .
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git commit -m "
Added lab01 report and its directory, and other directories"
[master b43ae93] Added lab01 report and its directory, and other directories
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 "labs/lab01/report/\320\233\320\2361_\320\241\320\260\320\262\320\265
\320\275\320\272\320\276\320\262\320\260_\320\276\321\202\321\207\321\221\321\202.pdf"
 create mode 100644 "labs/lab02/report/\320\233\320\2362_\320\241\320\260\320\262\320\265
\320\275\320\272\320\276\320\262\320\260_\320\276\321\202\321\207\320\265\321\202.doc"
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git push
Перечисление объектов: 10, готово.
Подсчет объектов: 100% (10/10), готово.
Сжатие объектов: 100% (7/7), готово.
Запись объектов: 100% (9/9), 1.06 МиБ | 3.88 МиБ/с, готово.
Total 9 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:aesavenkova/study_2024-2025_arh-pc.git
 7a3df13..b43ae93 master -> master
aesavenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$
```

Рис. 7.3: Загрузка отчета на гитхаб

## **5. Выводы**

В ходе данной лабораторной работы я изучила идеологию и применение средств контроля версий и приобрела практические навыки по работе с системой git.

## 6. Источники

1. <https://esystem.rudn.ru/mod/resource/view.php?id=1030826>
2. <https://github.com/>