

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: Мурашов И. В.

Группа: НКАбд-04-23

МОСКВА

2023 г.

Содержание

1 Цель работы	3
2 Выполнение лабораторной работы	4
3 Выводы	10

1 Цель работы

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

2 Выполнение лабораторной работы

1. Настройка github

Поскольку у меня есть учётная запись на сайте <https://github.com/> и основные данные заполнены (рис. 1), перехожу к следующему шагу лабораторной работы.

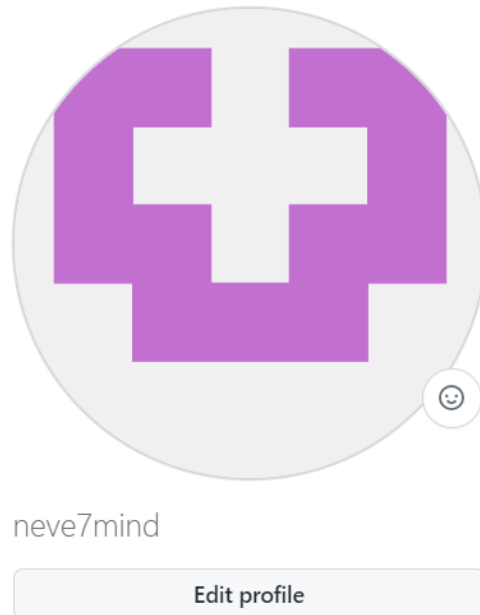


Рис. 1. Мой профиль на сайте github

2. Базовая настройка git

Сначала я делаю предварительную конфигурацию git. Открываю терминал и ввожу следующие команды, указав имя и email моего репозитория (рис. 2.1).

```
[ivmurashov@fedora ~]$ git config --global user.name "<neve7mind>"  
[ivmurashov@fedora ~]$ git config --global user.email "<ivan.murashov1200@gmail.com>"
```

Рис. 2.1. Предварительная конфигурация git

Затем настраиваю utf-8 в выводе сообщений git, задаю имя начальной ветки (буду называть её master), настраиваю параметры autocrlf и safecrlf (рис. 2.2).

```
[ivmurashov@fedora ~]$ git config --global core.quotepath false
[ivmurashov@fedora ~]$ git config --global init.defaultBranch master
[ivmurashov@fedora ~]$ git config --global core.autocrlf input
[ivmurashov@fedora ~]$ git config --global core.safecrlf warn
```

Рис. 2.2. Предварительная конфигурация git

3. Создание SSH ключа

Для последующей идентификации пользователя на сервере репозитория необходимо сгенерировать пару ключей (приватный и открытый). Для этого я ввожу команду ‘ssh-keygen -C "Имя Фамилия <work@mail>"’, указывая имя пользователя и электронную почту (рис. 3.1). Ключ автоматически сохраняется в каталоге ~/.ssh/.

```
[ivmurashov@fedora ~]$ ssh-keygen -C "neve7mind <ivan.murashov1200@gmail.com>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ivmurashov/.ssh/id_rsa):
Created directory '/home/ivmurashov/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ivmurashov/.ssh/id_rsa
Your public key has been saved in /home/ivmurashov/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:DGQmbL0lJYieZmzY6mN2h1I7bHdNQX1PvNCwf40B3Xs neve7mind <ivan.murashov1200@gmail.com>
The key's randomart image is:
+---[RSA 3072]---+
|  o.o=... .+ |
|  . +=o.o . =,= |
| = o . = o.* o |
| . 0 .o. . = . |
| =      S. ..oE |
| . . . . .o |
| .o.. o      |
| .*=o. . .   |
| +.+o .      |
+---[SHA256]-----+
```

Рис. 3.1. Генерация SSH-ключа

Далее необходимо загрузить сгенерированный открытый ключ. Для этого копирую из локальной консоли ключ в буфер обмена с помощью команды ‘cat ~/.ssh/id_rsa.pub | xclip -sel clip’ (рис. 3.2).

```
[ivmurashov@fedora ~]$ cat ~/.ssh/id_rsa.pub | xclip -sel clip
```

Рис. 3.2. Копирование открытого ключа из консоли

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. Вставляю ключ в появившееся на сайте поле и указываю для ключа имя (Title) (рис. 3.3).

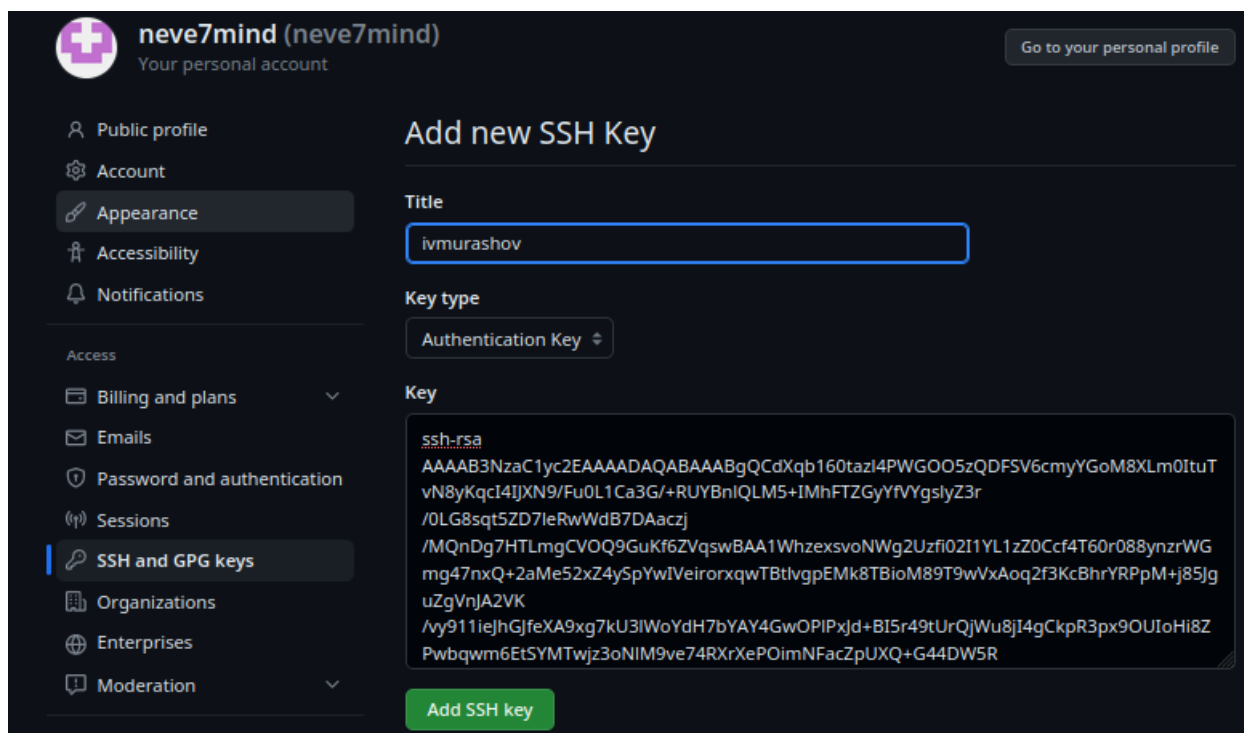


Рис. 3.3. Добавление SSH-ключа в github

4. Создание рабочего пространства и репозитория курса на основе шаблона

Открываю терминал и создаю каталог для предмета «Архитектура компьютера» с помощью команды 'mkdir'. Проверяю наличие каталога в корневом каталоге с помощью команды 'ls' (рис. 4).

```
[ivmurashov@fedora ~]$ mkdir -p ~/work/study/2023-2024/"Архитектура компьютера"
[ivmurashov@fedora ~]$ ls
DataWindows  Видео      Загрузки   Музыка     'Рабочий стол'
work         Документы  Изображения  Общедоступные  Шаблоны
```

Рис. 4. Создание каталога и проверка его наличия

5. Создание репозитория курса на основе шаблона

Репозиторий на основе шаблона можно создать через web-интерфейс github. Перехожу на страницу репозитория с шаблоном курса <https://github.com/yamadharm/course-directory-student-template>. Выбираю Use this template. В открывшемся окне задаю имя репозитория (Repository name) study_2023–2024_arh-pc и создаю репозиторий (Create repository from template) (рис. 5.1).

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 (*).

Owner * neve7mind / Repository name * study_2023-2024_arh-pc
 study_2023-2024_arh-pc is available.

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

Description (optional)

☒ **Public**
 Anyone on the Internet can see this repository. You choose who can commit.

☐ **Private**
 You choose who can see and commit to this repository.

🔔 You are creating a public repository in your personal account.

[Create repository](#)

Рис. 5.1. Создание репозитория на github

Открываю терминал и, переходя в каталог курса: `cd ~/work/study/2023–2024/"Архитектура компьютера"`, клонирую созданный репозиторий (рис. 5.2).

```
[ivmurashov@fedora ~]$ git clone --recursive git@github.com:neve7mind/study_2023-2024_arhpc.git
Клонирование в «study_2023-2024_arhpc»...
```

Рис. 5.2. Клонирование репозитория

6. Настройка каталога курса

Перехожу в каталог курса, удаляю лишние файлы, создаю необходимые каталоги и отправляю файлы на сервер (рис. 6.1).

```
[ivmurashov@fedora Архитектура компьютера]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/study_2023-2024_arhpc
[ivmurashov@fedora study_2023-2024_arhpc]$ rm package.json
[ivmurashov@fedora study_2023-2024_arhpc]$ echo arhpc > COURSE
[ivmurashov@fedora study_2023-2024_arhpc]$ git add .
[ivmurashov@fedora study_2023-2024_arhpc]$ git commit -am 'feat(main): make course structure'
[master 52505fb] feat(main): make course structure
 2 files changed, 1 insertion(+), 14 deletions(-)
 delete mode 100644 package.json
[ivmurashov@fedora study_2023-2024_arhpc]$ git push
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 288 байтов | 288.00 КиБ/с, готово.
Всего 3 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:neve7mind/study_2023-2024_arhpc.git
 37c949a..52505fb master -> master
```

Рис. 6.1. Перемещение, действия с файлами и каталогами, отправка данных на сервер

Проверяю правильность создания иерархии рабочего пространства в локальном репозитории и на странице github (рис. 6.2).

neve7mind feat(main): make course structure 24e1a1d · now History

Name	Last commit message	Last commit date
..		
lab01	feat(main): make course structure	now
lab02	feat(main): make course structure	now
lab03	feat(main): make course structure	now
lab04	feat(main): make course structure	now
lab05	feat(main): make course structure	now
lab06	feat(main): make course structure	now
lab07	feat(main): make course structure	now
lab08	feat(main): make course structure	now
lab09	feat(main): make course structure	now
lab10	feat(main): make course structure	now
lab11	feat(main): make course structure	now
README.md	feat(main): make course structure	now
README.ru.md	feat(main): make course structure	now

Рис. 6.2. Страница репозитория

7. Выполнение заданий для самостоятельной работы

1. Создаю отчёт по выполнению лабораторной работы в каталоге labs>lab02>report (рис. 7.1). Работать я буду в LibreOffice Writer.

```
[ivmurashov@fedora report]$ touch Л02_Мурашов_отчет
```

Рис. 7.1. Создание файла

2. Копирую отчет по выполнению первой лабораторной работы в подкаталог report каталога lab01, перенося его из ОС Windows, в которой я работал над отчетом ранее, в ОС Linux с помощью общей папки. Затем, проверяю наличие отчёта с помощью команды 'ls' (рис. 7.2).

```
[ivmurashov@fedora lab01]$ ls report
Л01_Мурашов_отчет.pdf
```

Рис. 7.2. Просмотр содержимого каталога

3. Загружаю файл Л01_Мурашов_отчет.pdf на github с помощью следующих команд: 'git add' – для добавления файла, 'git commit -am 'feat(main): make course structure' – для сохранения всех добавленных изменений и 'git push' – для отправки всех произведённых изменений в репозиторий (рис. 7.3).

```
[ivmurashov@fedora report]$ git add Л01_Мурашов_отчет.pdf
[ivmurashov@fedora report]$ git commit -am 'feat(main): make course structure'
[master 19df81f] feat(main): make course structure
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100755 labs/lab01/report/Л01_Мурашов_отчет.pdf
[ivmurashov@fedora report]$ git push
Перечисление объектов: 8, готово.
Подсчет объектов: 100% (8/8), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (5/5), готово.
Запись объектов: 100% (6/6), 643.53 КиБ | 3.39 МиБ/с, готово.
Всего 6 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:neve7mind/study_2023-2024_arhpc.git
a900be0..19df81f master -> master
```

Рис. 7.3. Загрузка файла на github через командную строку

3 Выводы

В ходе выполнения лабораторной работы я изучил идеологию и применение средств контроля версий, а также приобрел практические навыки по работе с системой git.