

Российский университет дружбы народов

Факультет физико-математических и естественных наук
(«Фундаментальная информатика и информационные технологии»)

ОТЧЕТ по лабораторной работе №2

«Архитектура компьютеров и операционные системы»

Студент: Абдулфазова Лейла Али гызы
Группа: НКАБд-01-23
№ 1032235809

Содержание отчета:

1. Цель работы
2. Задания
3. Теоретическое введение
4. Выполнение лабораторной работы
5. Ответы на задания для самостоятельной работы
6. Выводы

1. Цель работы

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

2. Задания

- 1) Настройка GitHub
- 2) Базовая настройка git
- 3) Создание SSH-ключа
- 4) Создание рабочего пространства
- 5) Создание репозитория на основе шаблона
- 6) Настройка каталога курса

3. Теоретическое введение

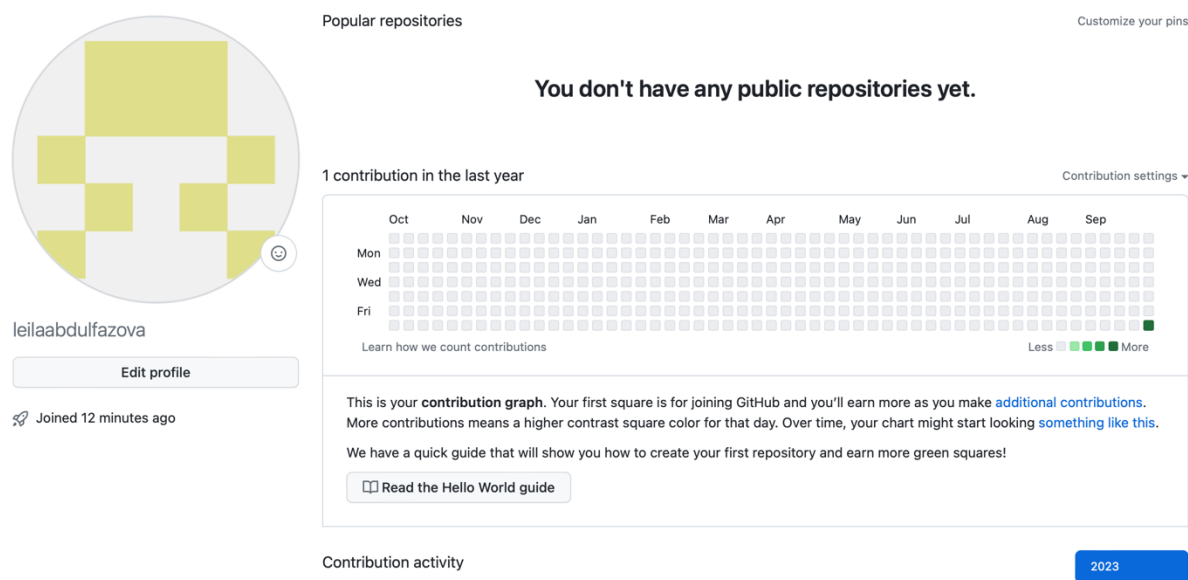
Система контроля версий (Version Control System, VCS) применяют при работу нескольких человек над одним проектом. Обычно основное дерево проекта хранится в локальном или удалённом репозитории, к которому настроен доступ для участников проекта. При внесении изменений в содержание проекта система контроля версий позволяет их фиксировать, совмещать изменения, произведённые разными участниками проекта, производить откат в любой более ранней версии проекта, если это требуется. В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Выполнение большинства функций по управлению версиями осуществляется специальным сервером. Участники проекта (пользователь) перед началом работы посредством определённых команд получает нужную ему версию файлов. После внесения изменений пользователь размещает новую версию в хранилище. При этом предыдущие версии не удаляются из центрального хранилища и к ним можно вернуться в любой момент. Сервер может сохранять не полную версию изменённых файлов, а производить так называемую дельта-компрессию – сохранять только изменения между последовательными версиями, что позволяет уменьшить объём хранимых данных. Системы контроля версий поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить изменения вовсе или заблокировать файлы для изменения. В зависимости от настроек блокировка не позволяет другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом привилегированный доступ только одному пользователю, работающему с файлом. Системы контроля версий также могут обеспечивать дополнительные, более гибкие функциональные возможности. Например, они могут поддерживать работу с несколькими версиями одного файла, сохраняя общую историю изменений до точки ветвления версий и собственные истории изменений каждой ветви. Обычно доступна информация о том, кто из участников, когда и какие изменения вносил. Обычно такого рода информация хранится в журнале изменений, доступ к которому можно ограничить. В отличие от классических, в распределённых системах контроля версий центральный репозиторий не является обязательным. Среди классических VCS наиболее известны CVS,

Subversion, а среди распределённых - Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд.

4. Выполнение лабораторной работы

1) Настройка GitHub

1.1 Создаю учётную запись, ввожу свои данные и подтверждаю свою учётную запись (рис 1)



(рис 1) Создание аккаунта на GitHub

2) Базовая настройка git

2.1 Открываю терминал и ввожу следующие команды, указав имя и mail владельца репозитория с помощью команды: `git config --global user.name "<leilaabdufzova>"`
`git config --global user.mail "<leilaaligyzy@mail.ru>"` (рис 2)

```
mansur@ubuntu:~$ git config --global user.name "<leilaabdufzova>"
mansur@ubuntu:~$ git config --global user.mail "<leilaaligyzy@mail.ru>"
```

(рис 2) Предварительная конфигурация git

2.2 Настраиваю utf-8 в выводе сообщений команды `"git config --global core.quotepath false"` и задаю имя начальной ветки (будем называть её master) (рис 3)

```
mansur@ubuntu:~$ git config --global core.quotepath false
mansur@ubuntu:~$ git config --global init.defaultBranch master
```

(рис 3) Настройка utf-8 и имя начальной ветки

2.3 Задаю параметр autocrlf со значением input (рис 4)

```
mansur@ubuntu:~$ git config --global core.autocrlf input
```

(рис 4) Параметр autocrlf

2.4 Задаю параметр safecrlf со значением warn, так Git будет определять преобразование на обратимость (рис 5)

```
mansur@ubuntu:~$ git config --global core.safecrlf warn
```

(рис 5) Параметр safecrlf

3) Создание SSH-ключа

3.1 Для последующей идентификации пользователя на сервере перозитория необходимо сгенерировать пару ключей (приватный и открытый), используя команду: `ssh-keygen -C «Имя Фамилия <work@mail>»` (рис 6). Ключ автоматически сохранится в каталоге `~/.ssh`

```
mansur@ubuntu:~$ ssh-keygen -C "leilaabdufzova <leilaaligyzy@mail.ru>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/mansur/.ssh/id_rsa): Leilanauralke
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in Leilanauralke
Your public key has been saved in Leilanauralke.pub
The key fingerprint is:
SHA256:HlgoFOA1RdiTVocPLYk/3xdTe2pYMJ3E0AWYLSAIPxA leilaabdufzova <leilaaligyzy@mail.ru>
The key's randomart image is:
+----[RSA 3072]-----+
| ..E0o=0+o..Xo+..|
| . o.+B.=o. * * .|
| . ..+O.+ + ..|
| . +o . +..|
| . So . o +..|
| . .. o +|
| . o|
| ..|
+----[SHA256]-----+
```

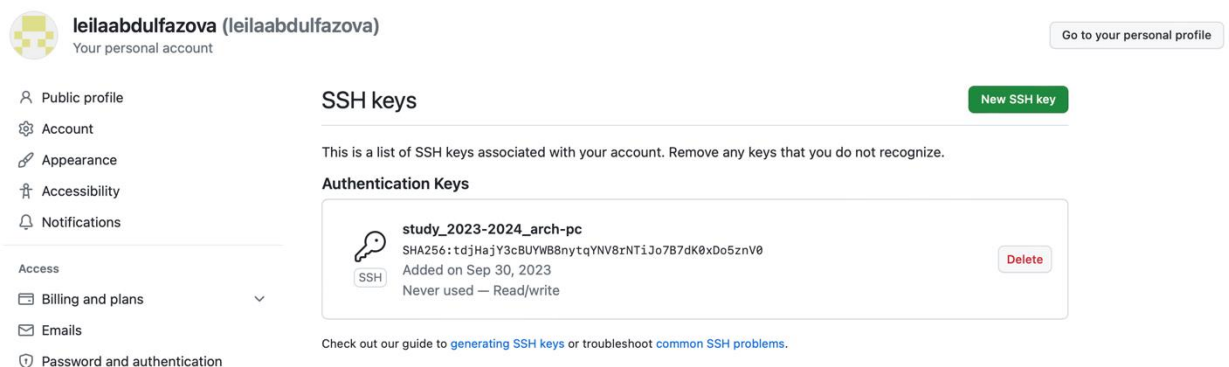
(рис 6) Создание SSH-ключа

3.2 Копирую открытый ключ из директории, в которой он был сохранён, используя утилиту “xclip” (рис 7)

```
mansur@ubuntu:~$ cat ~/.ssh/id_rsa.pub | xclip -sel clip
```

(рис 7) Копирование ключа

3.3 Далее необходимо загрузить сгенерированный открытый ключ. Для этого захожу на сайт <http://github.org/> под своей учётной записью и перехожу в меня Settings. После этого выбираю в боковом меню «SSH and GPG keys» и нажимаю кнопку «New SSH key». После чего вставляю скопированный ключ, указываю имя в поле “Title” и добавляю SSH-ключ (рис 8)



(рис 8) Добавление ключа



3.4 Проверяю созданный ключ (рис 9)

SSH keys

New SSH key

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

Authentication Keys

leilaabulfazova

SHA256:tdjHajY3cBUYWB8nytqYNV8rNTiJo7B7dK0xDo5znV0

Added on Sep 30, 2023

Never used — Read/write

Delete

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH problems](#).

(рис 9) Проверка ключа

4) Создание рабочего пространства

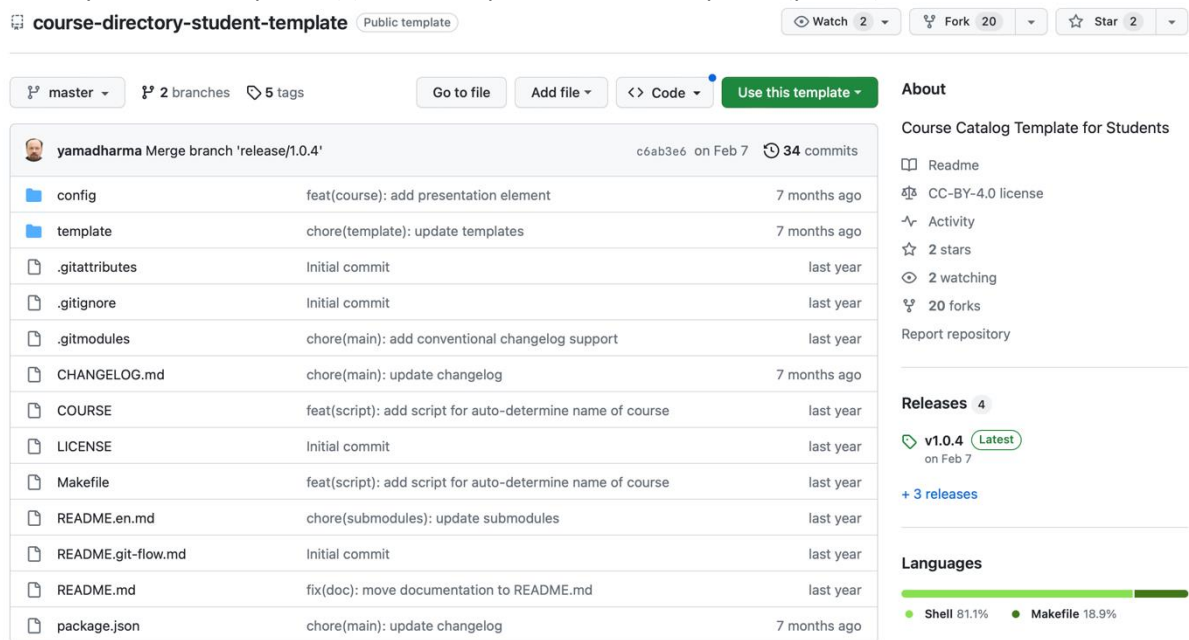
4.1 Выхожу из браузера, открываю терминал, создаю директорию, рабочее пространство, используя утилиту «mkdir». С помощью ключа -p создаю все директории после домашней ~/work/study/2023-2024/'Архитектура компьютера' рекурсивно. С помощью проверяю создание всех каталогов и подкаталогов (рис 10)

```
mansur@ubuntu:~$ cd ~/work/study/2023-2024/'Архитектура компьютера'
```

(рис 10) Создание рабочего пространства

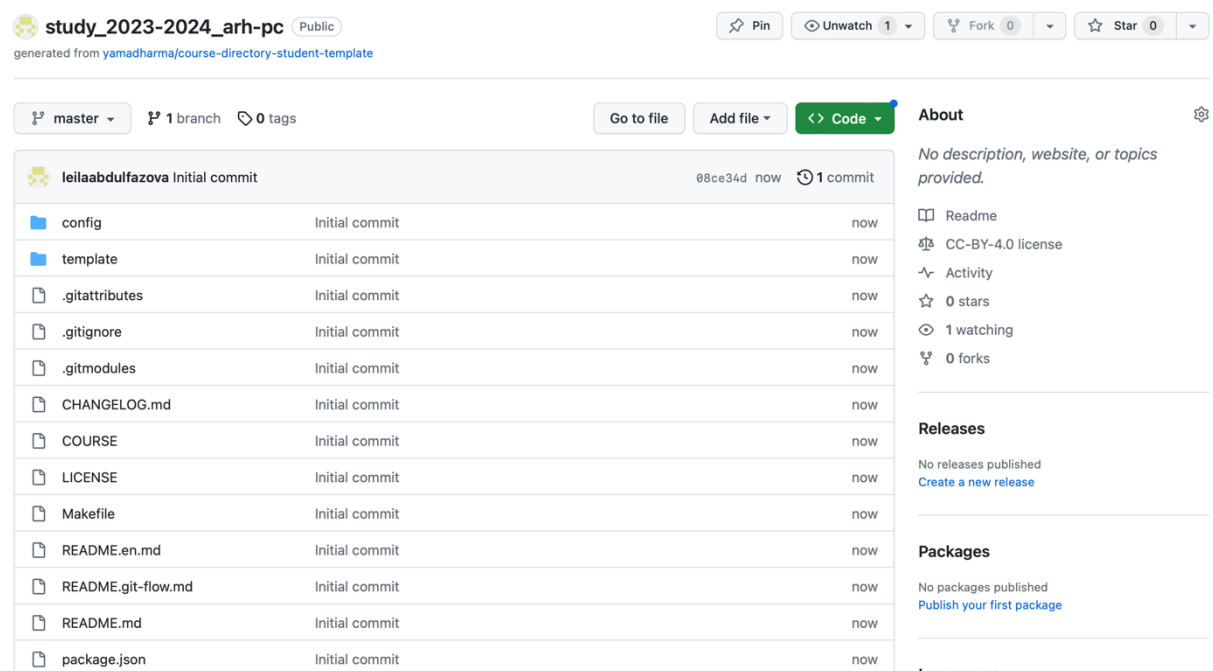
5) Создание репозитория на основе шаблона

5.1 Перехожу на страницу репозитория с шаблоном курса “https://github.com/yamadharma.course-directory-student-template”. Далее выбираю “Use this template” (рис 11)



(рис 11) Страница шаблона для репозитория

5.2 В открывшемся окне задаю имя репозитория (Repository name) study_2023-2024_arch-pc и создаю репозиторий (кнопка Create repository from template) (рис 12)



(рис 13) Созданный репозиторий

5.3 С помощью утилиты перехожу в созданный каталог курса (рис 14)

```
mansur@ubuntu:~$ cd ~/work/study/2023-2024/'Архитектура компьютера'
```

(рис 14) Перемещение между директориями

5.4 Клонирование созданного репозитория с помощью команды: `git clone --recursive`

`git@github.com:<user name>/study_2023-2024-arh-pc.git`, которую можно скопировать на сайте (рис 15)

```
Mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера$ git clone --recursive git@github.com:leilaabdufzova/study_2023-2024_arh-pc.git
Cloning into 'study_2023-2024_arh-pc'...
remote: Enumerating objects: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 27 (delta 1), reused 11 (delta 0), pack-reused 0
Receiving objects: 100% (27/27), 16.94 KiB | 4.23 MiB/s, done.
Resolving deltas: 100% (1/1), done.
Submodule 'template/presentation' (https://github.com/yamadharma/academic-presentation-markdown-template.git) registered for path 'template/presentation'
Cloning into '/home/mansur/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/template/presentation'...
remote: Enumerating objects: 82, done.
remote: Counting objects: 100% (82/82), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 82 (delta 28), reused 77 (delta 23), pack-reused 0
Cloning into '/home/mansur/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/template/report'...
remote: Enumerating objects: 101, done.
remote: Counting objects: 100% (101/101), done.
remote: Compressing objects: 100% (70/70), done.
remote: Total 101 (delta 40), reused 88 (delta 27), pack-reused 0
Receiving objects: 100% (101/101), 327.25 KiB | 2.29 MiB/s, done.
Resolving deltas: 100% (40/40), done.
Submodule path 'template/presentation': checked out 'b1be3800ee91f5809264cb755d316174540b753e'
Submodule path 'template/report': checked out '1d1b61dcac9c287a83917b82e3aef11a33b1e3b2'
Mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера$
Mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера$ git clone --recursive git@github.com:leilaabdufzova/study_2023-2024_arh-pc.git
Cloning into 'study_2023-2024_arh-pc'...
remote: Enumerating objects: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 27 (delta 1), reused 11 (delta 0), pack-reused 0
Receiving objects: 100% (27/27), 16.94 KiB | 4.23 MiB/s, done.
Resolving deltas: 100% (1/1), done.
Submodule 'template/presentation' (https://github.com/yamadharma/academic-presentation-markdown-template.git) registered for path 'template/presentation'
Cloning into '/home/mansur/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/template/presentation'...
remote: Enumerating objects: 82, done.
remote: Counting objects: 100% (82/82), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 82 (delta 28), reused 77 (delta 23), pack-reused 0
Cloning into '/home/mansur/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/template/report'...
remote: Enumerating objects: 101, done.
remote: Counting objects: 100% (101/101), done.
remote: Compressing objects: 100% (70/70), done.
remote: Total 101 (delta 40), reused 88 (delta 27), pack-reused 0
Receiving objects: 100% (101/101), 327.25 KiB | 2.29 MiB/s, done.
Resolving deltas: 100% (40/40), done.
Submodule path 'template/presentation': checked out 'b1be3800ee91f5809264cb755d316174540b753e'
Submodule path 'template/report': checked out '1d1b61dcac9c287a83917b82e3aef11a33b1e3b2'
Mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера$
```

(рис 15) Клонирование репозитория

6) Настройка каталога курса

6.1 Перехожу в каталог курса и удаляю лишние файлы (рис 16)

```
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера$ cd study_2023-2024_arh-pc
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc$ rm package.json
```

(рис 16) Перемещение по директории и удаление лишних файлов

6.2 Создаю необходимые каталоги (рис 17)

```
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc$ echo arch-pc > COURSE
```

(рис 17) Создание необходимых каталогов

6.3 Отправляю созданные каталоги на сервер. Добавляю созданные каталоги, используя «git add», сохраняю изменения на сервере как добавление курса, с помощью команды «git commit» (рис 18)

```
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc$ git commit -am 'feat(main): make course structure'
[master 45e9423] feat(main): make course structure
199 files changed, 54725 insertions(+), 14 deletions(-)
create mode 100644 labs/README.md
create mode 100644 labs/README.ru.md
create mode 100644 labs/lab01/presentation/Makefile
create mode 100644 labs/lab01/presentation/image/kulyabov.jpg
create mode 100644 labs/lab01/presentation/presentation.md
create mode 100644 labs/lab01/report/Makefile
create mode 100644 labs/lab01/report/bib/cite.bib
create mode 100644 labs/lab01/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab01/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab01/report/report.md
create mode 100644 labs/lab02/presentation/Makefile
create mode 100644 labs/lab02/presentation/image/kulyabov.jpg
create mode 100644 labs/lab02/presentation/presentation.md
create mode 100644 labs/lab02/report/Makefile
create mode 100644 labs/lab02/report/bib/cite.bib
create mode 100644 labs/lab02/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab02/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab02/report/report.md
create mode 100644 labs/lab03/presentation/Makefile
create mode 100644 labs/lab03/presentation/image/kulyabov.jpg
create mode 100644 labs/lab03/presentation/presentation.md
create mode 100644 labs/lab03/report/Makefile
```

(рис 18) Добавление и сохранение изменений на сервере

6.4 Отправляю все изменения на сервер, используя команду «push» (рис 19)

```
mansur@mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 302 bytes | 302.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:leilaabdufzova/study_2023-2024_arh-pc.git
   dd69045..0061880  master -> master
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc$
```

(рис 19) Отправление всех данных на сервер

6.5 Проверяю правильность выполнения задания на самом сайте GitHub (рис 20)

study_2023-2024_arh-pcPublicgenerated from yamadharma/course-directory-student-template

PinUnwatch1Fork0Star0

master1 branch0 tagsGo to fileAdd fileCode

leilaabdufzova feat(main): make course structure45e94234 minutes ago2 commits

config	Initial commit	27 minutes ago
labs	feat(main): make course structure	4 minutes ago
presentation	feat(main): make course structure	4 minutes ago
template	Initial commit	27 minutes ago
.gitattributes	Initial commit	27 minutes ago
.gitignore	Initial commit	27 minutes ago
.gitmodules	Initial commit	27 minutes ago
CHANGELOG.md	Initial commit	27 minutes ago
COURSE	feat(main): make course structure	4 minutes ago
LICENSE	Initial commit	27 minutes ago
Makefile	Initial commit	27 minutes ago
README.en.md	Initial commit	27 minutes ago
README.git-flow.md	Initial commit	27 minutes ago
README.md	Initial commit	27 minutes ago
prepare	feat(main): make course structure	4 minutes ago

About

No description, website, or topics provided.

Readme

CC-BY-4.0 license

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Languages

Python93.5%

TeX5.2%

Makefile1.1%

Shell0.2%

(рис 20) Проверка репозитория

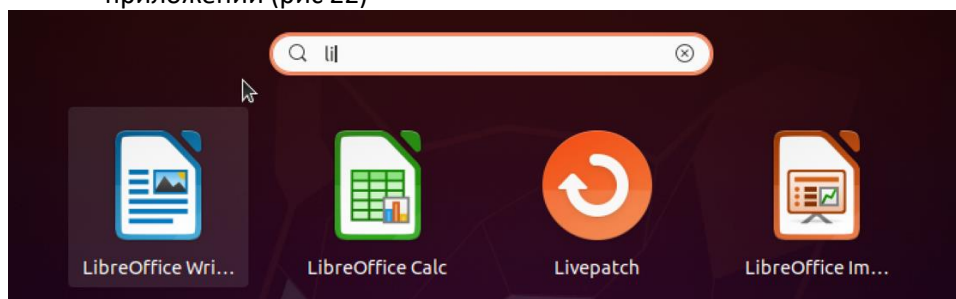
5. Ответы на задания для самостоятельной работы

- 1) Создаю файл для отчёта по лабораторной работе №2 с помощью утилиты «touch» (рис 21)

```
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-  
pc/labs/lab02/report$ touch лаб2  
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-  
pc/labs/lab02/report$ ls  
bib image Makefile pandoc report.md лаб2  
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-  
pc/labs/lab02/report$
```

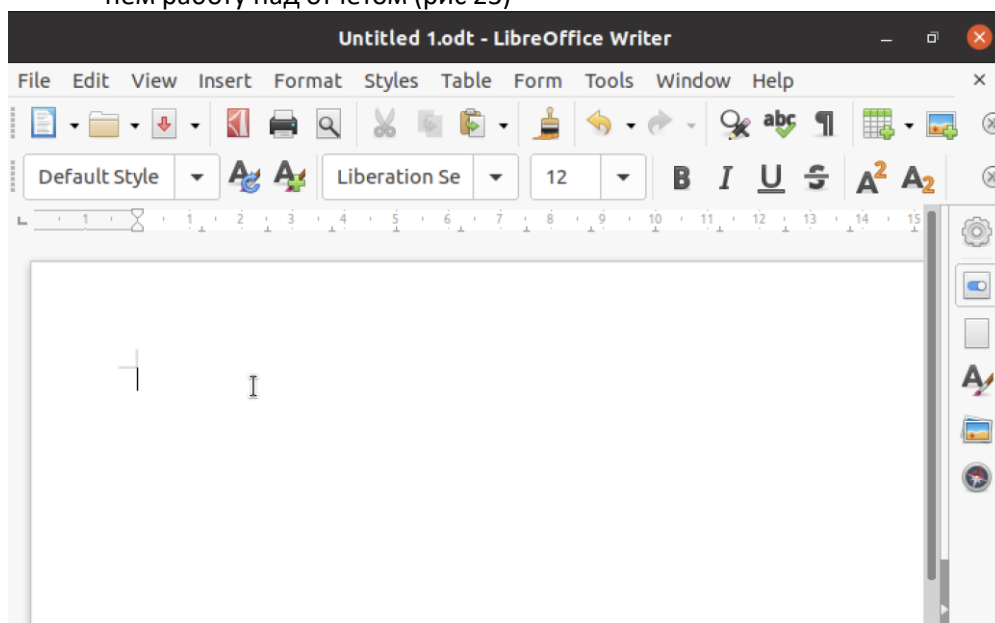
(рис 21) Создание файла

- 2) Составлять отчёт я могу в текстовом процессоре LibreOffice Writer, найдя его в меню приложений (рис 22)



(рис 22) Меню приложений

- 3) После открытия текстового процессора открываю в нём созданный файл и могу начать в нём работу над отчётом (рис 23)



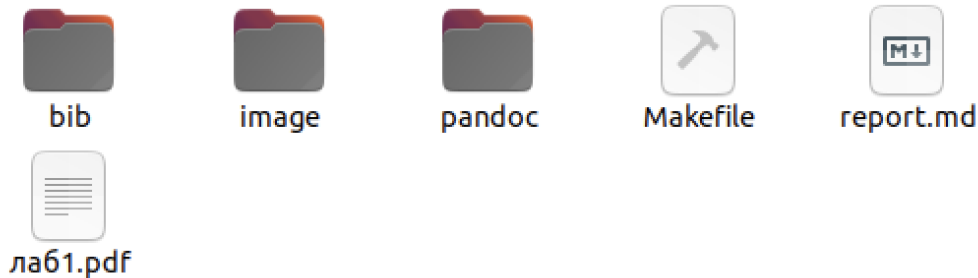
(рис 23) Работа над отчётом в текстовом процессоре

4) Перехожу из подкаталога lab02/report в подкаталог lab01/report (рис 24)

```
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab01/report$
```

(рис 24) Перемещение между директориями

5) Копирую первую лабораторную и проверяю выполненную команду с помощью ls (рис 25)



(рис 25) Проверка местонахождения файла

6) Перехожу в директорию labs/lab02/report (рис 25)

```
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc$ cd labs/lab02/report
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab02/report$
```

(рис 25) Перемещение по директории

7) Добавляю файл лаб2 (рис 26)

```
mansur@ubuntu:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab02/report$ git add лаб2.pdf
```

(рис 26) Добавление файла на сервер

8) Проверяю на сайте GitHub правильность выполнения заданий. Вижу, что обе лабораторные были добавлены (рис 27)

study_2023-2024_arh-pc / labs / lab01 / report /			Add file	...
leilaabdulfazova Add files via upload			cedf0 - 1 hour ago History	
Name	Last commit message	Last commit date		
..				
bib	feat(main): make course structure	15 hours ago		
image	feat(main): make course structure	15 hours ago		
pandoc	feat(main): make course structure	15 hours ago		
Makefile	feat(main): make course structure	15 hours ago		
report.md	feat(main): make course structure	15 hours ago		
лаб1.pdf	Add files via upload	1 hour ago		

study_2023-2024_arh-pc / labs / lab02 / report /			Add file	...
leilaabdulfazova Add files via upload			89febf - now History	
Name	Last commit message	Last commit date		
..				
bib	feat(main): make course structure	16 hours ago		
image	feat(main): make course structure	16 hours ago		
pandoc	feat(main): make course structure	16 hours ago		
Makefile	feat(main): make course structure	16 hours ago		
report.md	feat(main): make course structure	16 hours ago		
лаб2.pdf	Add files via upload	now		

6 . Выводы

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