

# **Отчёт по лабораторной работе**

**Дисциплина: Научное программирование**

Живцова Анна, 1132249547

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
3.1	Git . . . . .	7
3.2	SSH PGP . . . . .	8
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>10</b>
<b>5</b>	<b>Выводы</b>	<b>15</b>
<b>6</b>	<b>Список литературы</b>	<b>16</b>

# Список иллюстраций

4.1	Личная учетная запись на github . . . . .	10
4.2	Проверка наличия программного обеспечения . . . . .	10
4.3	Базовая настройка git . . . . .	11
4.4	Создание ключей ssh . . . . .	11
4.5	Создание ключа pgr . . . . .	12
4.6	Подключение github через pgr . . . . .	12
4.7	Настройка подписи к коммитам с pgr . . . . .	13
4.8	Создание и настройка репозитория . . . . .	13
4.9	Итоговый репозиторий . . . . .	14

## **Список таблиц**

# 1 Цель работы

- Изучить идеологию и применение средств контроля версий.
- Освоить умения по работе с git.
- Настроить базовую конфигурацию для работы с git.

## 2 Задание

- Создать базовую конфигурацию для работы с git.
- Создать ключ SSH.
- Создать ключ PGP.
- Настроить подписи git.
- Зарегистрироваться на Github.
- Создать локальный каталог для выполнения заданий по предмету.

## 3 Теоретическое введение

### 3.1 Git

Git – распределённая система управления версиями.

Github – крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки. Веб-сервис основан на системе контроля версий Git.

В ходе работы были использованы следующие команды для конфигурации git:

Установка имени

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

Установка адреса электронной почты

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

Настройка utf-8 в выводе сообщений git

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

Задание имени начальной ветки

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

Установка параметра autocrlf

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

Немного о параметре autocrlf. Параметр “core.autocrlf” придумали для обеспечения работы над одним проектом программистов из разных операционных систем. Предполагается, что программист в операционной системе «Windows» будет работать с файлами, в которых окончания строк только вида CRLF. При этом предполагается, что он включит для проекта настройку “core.autocrlf” со значением “true”. Тогда он будет работать в своей папке проекта с файлами, в которых

окончания строк будут вида CRLF, при этом в базе данных “Git” эти же файлы будут сохранены с окончаниями вида LF. Программист в операционной системе “Windows” этого даже не заметит, ведь конвертация происходит автоматически.

В тот же момент программист в Unix-подобной операционной системе будет работать с той же базой данных “Git”, но у него для проекта будет включена настройка «core.autocrlf» со значением “input” (или со значением “false”). Он будет получать из базы данных файлы с окончаниями строк вида LF, как и принято в Unix-подобных операционных системах.

Установка параметра safecrlf

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

Если core.safecrlf установлен на “true” или “warn”, Git проверяет, если преобразование является обратимым для текущей настройки core.autocrlf. core.safecrlf true - отвержение необратимого преобразования lf<->crlf. Полезно, когда специфические бинарники похожие на текстовые файлы. core.safecrlf warn - печать только предупреждение, но принимает необратимый переход.

Установка параметра commit.gpgsign для автоматической подписи коммитов gpg ключем

```
git config --global user.signingkey [PGP Fingerprint]; git config --global commit.gpgsign true
```

Более подробно про систему Git [1],

## 3.2 SSH PGP

SSH (от англ. Secure Shell) — криптографический сетевой протокол, предназначенный для удалённого доступа к операционной системе и осуществления безопасного удалённого управления в рамках незащищённой сети (например, через интернет).

SSH обеспечивает защищённый канал связи между клиентом и сервером, через который можно:



- передавать данные (почтовые, видео, файлы);
- работать в командной строке;
- удалённо запускать программы, в том числе графические.

PGP (англ. Pretty Good Privacy) — компьютерная программа, также библиотека функций, позволяющая выполнять операции шифрования и цифровой подписи сообщений, файлов и другой информации, представленной в электронном виде, в том числе прозрачное шифрование данных на запоминающих устройствах, например, на жёстком диске.

Шифрование PGP осуществляется последовательно хешированием, сжатием данных, шифрованием с симметричным ключом, и, наконец, шифрованием с открытым ключом, причём каждый этап может осуществляться одним из нескольких поддерживаемых алгоритмов. Симметричное шифрование производится с использованием одного из семи симметричных алгоритмов (AES, CAST5, 3DES, IDEA, Twofish, Blowfish, Camellia) на сеансовом ключе. Сеансовый ключ генерируется с использованием криптографически стойкого генератора псевдослучайных чисел. Сеансовый ключ зашифровывается открытым ключом получателя с использованием алгоритмов RSA или Elgamal (в зависимости от типа ключа получателя). Каждый открытый ключ соответствует имени пользователя или адресу электронной почты. Первая версия системы называлась Сеть доверия и противопоставлялась системе X.509, использовавшей иерархический подход, основанной на удостоверяющих центрах, добавленный в PGP позже. Современные версии PGP включают оба способа.

## 4 Выполнение лабораторной работы

1. Настройка github. Я работе я использовала свою учетную запись (AnnaZhiv) на github (см рис. 4.1).

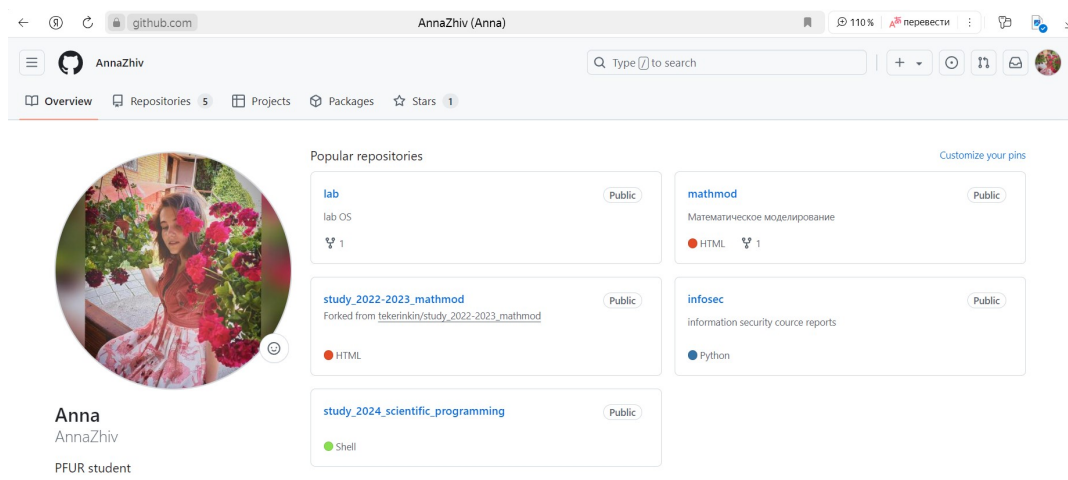


Рис. 4.1: Личная учетная запись на github

2. Установка программ. К счастью, основное программное обеспечение (git, gh) также было установлено на моем компьютере (см рис. 4.2).

```
PS C:\Users\annaz\Documents\study\scientific_programming\sc_prog> git --version
git version 2.42.0.windows.2
PS C:\Users\annaz\Documents\study\scientific_programming\sc_prog> gh --version
gh version 2.32.1 (2023-07-24)
https://github.com/cli/cli/releases/tag/v2.32.1
```

Рис. 4.2: Проверка наличия программного обеспечения

3. Базовая настройка git. Для базовой настройки git использовала команды, приведенные на рисунке 4.3. Объяснение выполненных команд содержится в теоретической части отчета.

```

PS C:\Users\annaz\Documents\study\scientific_programming> git config --global user.name "AnnaZhiv"
PS C:\Users\annaz\Documents\study\scientific_programming> git config --global user.email "annazhiv93@mail.ru"
PS C:\Users\annaz\Documents\study\scientific_programming> git config --global core.quotepath false
PS C:\Users\annaz\Documents\study\scientific_programming> git config --global init.defaultBranch master
PS C:\Users\annaz\Documents\study\scientific_programming> git config --global core.autocrlf input
PS C:\Users\annaz\Documents\study\scientific_programming> git config --global core.safecrlf warn

```

Рис. 4.3: Базовая настройка git

4. Ключи ssh. Ключи ssh были созданы (см рис. 4.4), но не использовались.

```

PS C:\Users\annaz\Documents\study\scientific_programming> ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\annaz\.ssh/id_rsa): ssh_for_sc_pr.txt
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in ssh_for_sc_pr.txt.
Your public key has been saved in ssh_for_sc_pr.txt.pub.
The key fingerprint is:
SHA256:HBH1MVTk5tJcUC8n62xXeR8pZ090J0F+twD0FkKrOrY annaz@DESKTOP-VTVTQ4A
The key's randomart image is:
+---[RSA 4096]-----+
|      ooo==+..      |
|      o.O+==.+      |
|      .O+o= =       |
|      .==oo *+      |
|      S.. +oo**      |
|      . . o=oo       |
|      +      +.+     |
|      . O      ..     |
|      E              |
+---[SHA256]-----+
PS C:\Users\annaz\Documents\study\scientific_programming> ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (C:\Users\annaz\.ssh/id_ed25519): another_ssh_for_sc_pr

```

Рис. 4.4: Создание ключей ssh

5. Ключ PGP. Создала ключ PGP (см рис. 4.5) с рекомендованными параметрами и использовала его для подключения к github (см рис. 4.6, 4.7).

```

annaz@DESKTOP-VTVTQ4A MINGW64 ~
$ gpg --full-generate-key
gpg (GnuPG) 2.2.41-unknown: Copyright (C) 2022 g10 Code GmbH
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Please select what kind of key you want:
  (1) RSA and RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (14) Existing key from card
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 4096
Requested keysize is 4096 bits
Please specify how long the key should be valid.
    0 = key does not expire
    <n>  = key expires in n days
    <n>w  = key expires in n weeks
    <n>m  = key expires in n months
    <n>y  = key expires in n years
Key is valid for? (0) 0
Key does not expire at all
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.


Real name: Anna_Zhiv
Email address: annazhiv03@mail.ru
Comment: aaa
You selected this USER-ID:
  "Anna_Zhiv (aaa) <annazhiv03@mail.ru>"

```

Рис. 4.5: Создание ключа pgp

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

---



Scientific programming

Email address: annazhiv03@mail.ru

Key ID: A95ED8F5D6C62B71

Subkeys: 168282481EB70ACC

Added on Sep 1, 2024

GPG

Рис. 4.6: Подключение github через pgp

```

annaz@DESKTOP-VTVTQ4A MINGW64 ~
$ gpg --export --armor 88A500ADE68B9FC6C8F3692EA95ED8F5D6C62B71 > ./
annaz@DESKTOP-VTVTQ4A MINGW64 ~
$ git config --global user.signingkey ^[[200~88A500ADE68B9FC6C8F3692
62B71
annaz@DESKTOP-VTVTQ4A MINGW64 ~
$ git config --global commit.gpgsign true
annaz@DESKTOP-VTVTQ4A MINGW64 ~
$ git config --global gpg.program $(which gpg2)
which: no gpg2 in C:/c/Users/annaz/bin:/mingw64/bin:/usr/local/bin:/u

```

Рис. 4.7: Настройка подписи к коммитам с gpg

6. Настройка рабочего пространства. Следуя инструкциям я получила репозиторий для ведения лабораторных работ.

```

PS C:\Users\annaz\Documents\study\scientific_programming> gh repo create study_2021-2022_os-intro
To get started with GitHub CLI, please run: gh auth login
Alternatively, populate the GH_TOKEN environment variable with a GitHub API authentication token.
PS C:\Users\annaz\Documents\study\scientific_programming> --template=yamadharma/course-directory-student-templ
lichgh repo create study_2021-2022_os-intro ^C
PS C:\Users\annaz\Documents\study\scientific_programming> gh repo create study_2024_scientific_programming --te
madharma/course-directory-student-template --public
To get started with GitHub CLI, please run: gh auth login
Alternatively, populate the GH_TOKEN environment variable with a GitHub API authentication token.
PS C:\Users\annaz\Documents\study\scientific_programming> gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? SSH
? Upload your SSH public key to your GitHub account? C:\Users\annaz\.ssh\id_rsa.pub
? Title for your SSH key: (GitHub CLI)
? Title for your SSH key: GitHub CLI
? How would you like to authenticate GitHub CLI? Login with a web browser
1 First copy your one-time code: C3E5-12A2
Press Enter to open github.com in your browser...
2 Authentication complete.
3 gh config set -h github.com git_protocol ssh
4 Configured git protocol
5 SSH key already existed on your GitHub account: C:\Users\annaz\.ssh\id_rsa.pub
6 Logged in as AnnaZhiv
PS C:\Users\annaz\Documents\study\scientific_programming> gh repo create study_2024_scientific_programming --te
madharma/course-directory-student-template --public
7 Created repository AnnaZhiv/study_2024_scientific_programming c
PS C:\Users\annaz\Documents\study\scientific_programming> git cl

```

Рис. 4.8: Создание и настройка репозитория

study\_2024\_scientific\_programming

Public

generated from [yamadhama/course-directory-student-template](#)

Pin

Unwatch 1

Fork 0

Star 0

master 1 Branch 0 Tags

Go to file

Add file

Code

AnnaZhiv

make course structure

6e51669 · 15 minutes ago

2 Commits

config	Initial commit	2 days ago
labs/lab01	make course structure	15 minutes ago
template	Initial commit	2 days ago
.gitattributes	Initial commit	2 days ago
.gitignore	Initial commit	2 days ago
.gitmodules	Initial commit	2 days ago
CHANGELOG.md	Initial commit	2 days ago
COURSE	Initial commit	2 days 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

Рис. 4.9: Итоговый репозиторий

14

## 5 Выводы

В данной работе мне удалось освежить в памяти основы использования системы контроля версий git и изучить несколько новых команд. В частности использовать PGP ключ для подключения к github. В итоге выполнения лабораторной работы я получила рабочее пространство для выполнения лабораторных работ по дисциплине “научное программирование”.

## **6 Список литературы**

1. Chacon S., Straub B. Pro Git. Apress, 2014.