

Отчёт по лабораторной работе №1

Математическое моделирование

Самсонова Мария Ильинична

Содержание

Цель работы	1
Задание	1
Теоретическое введение	1
Выполнение лабораторной работы	2
Вывод.....	9

Цель работы

Подробное изучение идеологии и применение различных средств контроля версий, освоение практических навыков по работе с git.

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.

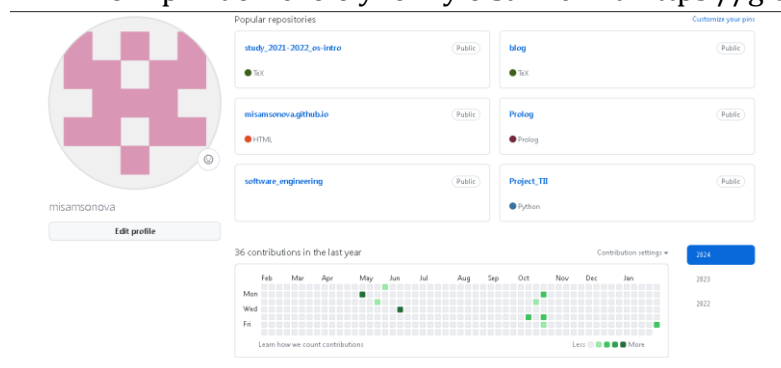
Теоретическое введение

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. При внесении изменений в содержание проекта система контроля версий позволяет их фиксировать, совмещать изменения, произведённые разными участниками проекта, производить откат к 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. Участник 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 дельтакомпрессию — 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 (слить) изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию, отменить 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. Кроме того, 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.

Выполнение лабораторной работы

1. Открываем свою учётную запись на <https://github.com>: (рис. [-@fig:001])



Учетная запись на github

2. Задаём имя и email владельца репозитория данными командами `git config --global user.name "Name Surname"` и `git config --global user.email "work@mail"`, настраиваем utf-8 в выводе сообщений `git` командой `git config --global core.quotePath false`, а также задаём имя начальной ветки (будем называть её master): (рис. [-@fig:002])

```
[misamsonova@fedora ~]$ git config --global user.Maria Samsonova
[misamsonova@fedora ~]$ git config --global user.1032216526@rudn.ru
[misamsonova@fedora ~]$ git config --global core.quotepath false
[misamsonova@fedora ~]$ git config --global core.quotepath
false
[misamsonova@fedora ~]$ git config --global init.defaultBranch master
[misamsonova@fedora ~]$ git config --global core.autocrlf input
[misamsonova@fedora ~]$ git config --global core.savecrlf warn
```

Задание имени и email пользователя, настройка utf-8 в выводе сообщений, задание имени начальной ветки

{#fig:002 width=70% }

3. Создаём ключей ssh по алгоритму rsa размером 4096 бит и по алгоритму ed25519: (рис. [-@fig:003])

```
[misamsonova@fedora ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/misamsonova/.ssh/id_rsa):
Created directory '/home/misamsonova/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/misamsonova/.ssh/id_rsa
Your public key has been saved in /home/misamsonova/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Kv7wqd/X00gzWfVHluxemywPKvZTNwtX77D96SSKvQ4 misamsonova@fedora
The key's randomart image is:
+---[RSA 4096]-----+
|          .oo |
|         ... o|
|        . . . |
|       .      +|
|      S.    ..*|
|     ..    =.Bo|
|    o .o E oo.B*o|
|   . +.o+oBoo +o+|
|  o+=.+=+Bo .oo|
+-----[SHA256]-----+
```

Создание ключа ssh по алгоритму rsa с ключём размером 4096 бит

{#fig:003 width=70% }

(рис. [-@fig:004])

```
[misamsonova@fedora ~]$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/misamsonova/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/misamsonova/.ssh/id_ed25519
Your public key has been saved in /home/misamsonova/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:LsiE0f9WAvKJ0Be4cLkBQiBVj5IjwNi0zEtbw1YhR7Y misamsonova@fedora
The key's randomart image is:
+--[ED25519 256]--+
|@+.oo=.          |
|=o.=++          |
|=o@ *Eo          |
|.B.# * o         |
|. @ = + S .      |
| o = . o o       |
|   o . +         |
|       o         |
|                  |
+-----[SHA256]-----+
[misamsonova@fedora ~]$
```

Создание ключа ssh по алгоритму ed25519

{#fig:004 width=70% }

4. Создаём ключ GPG, генерируя его и определяя различные характеристики: (рис. [-@fig:005])

```
[misamsonova@fedora ~]$ gpg --full-generate-key
gpg (GnuPG) 2.3.2; Copyright (C) 2021 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

gpg: создан каталог '/home/misamsonova/.gnupg'
gpg: создан щит с ключами '/home/misamsonova/.gnupg/pubring.kbx'
Выберите тип ключа:
  (1) RSA and RSA
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (9) ECC (sign and encrypt) *default*
 (10) ECC (только для подписи)
 (14) Existing key from card
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
Запрошенный размер ключа - 4096 бит
Выберите срок действия ключа.
    0 = не ограничен
    <n> = срок действия ключа - n дней
    <n>w = срок действия ключа - n недель
    <n>m = срок действия ключа - n месяцев
```

{#fig:005 width=70% }

5. Добавляем GPG ключ в GitHub, выводя список ключей и копируя отпечаток приватного ключа: (рис. [-@fig:006])

```
[misamsonova@fedora ~]$ gpg --list-secret-keys --keyid-format LONG
gpg: проверка таблицы доверия
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: глубина: 0 достоверных: 1 подписанных: 0 доверие: 0-, 0q, 0n, 0m, 0f
, 1u
/home/misamsonova/.gnupg/pubring.kbx
-----
sec   rsa4096/B0F5F76EB4F08EFE 2022-04-23 [SC]
      436A3CB94C856DB6E8096C06B0F5F76EB4F08EFE
uid   [ абсолютно ] Maria <1032216526@rudn.ru>
ssb   rsa4096/027B84C5E48C4958 2022-04-23 [E]

[misamsonova@fedora ~]$
```

Данные о ключе GPG

{#fig:006 width=70% }

6. Получаем содержимое ключа GPG и скопируем наш сгенерированный ключ в буфер обмена: (рис. [-@fig:007]) и (рис. [-@fig:008])

```
[misamsonova@fedora ~]$ gpg --armor --export B0F5F76EB4F08EFE | xclip -sel clip
[misamsonova@fedora ~]$ gpg --armor --export B0F5F76EB4F08EFE
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBGJkExABEADXHRvfl+cg9IQBrPywbFMkeXDr10v9koMDUYTIizWRJVgG8CIa
I8P3biUEjIUyTLoVjv3+LYAi4zAx2ZRzk5wLITS/ln94pE2StHa05/k5/Z6zZM3z
fzKcD0ftISxpdSxB4H+uo0VxGP86pKD0QCnkk/ijIAR77GS9PSUW0b+46bDWBZQ3
HKSDVp57psJFoZn1o12ggM+YE9cIusnUZpuV6NkPGI9NI2hGBMd2HoJjUdMDybPY
gx6u5xCj2MJwpDEBM5LthByYHdFPSWhySXmFaNWXIDil9DWEBzP1ECnsvH/rKoZc
AqC8QYVT/3ggjwKaNC205UoAQ5PH+V2B09un7RZA2Yew8fULPCo+qo1TTT5YfLLk
yJgYfxkrMka5Gg2QT5yMZavKa5ynsJb8CVj+ETM6f+SMCs9RaxsspiLFKU50yG0o
YsGZXSc2BkpNq7PfHZRrc0eOLHQ90IlhxyTQffGz3AbLoU0YR9NK/zGRC0qpdofB
XR9I3P/Vo7hSwFCPWD8M0fSeTBvSLv5hPmfiWu6uo4PC4IMguL6z5dhbf2ToQAA
XLckxKcU3v9peHwQhQCDDAGopjc9u0aXay0yDskNCUemCYPxVxeR9qg9DXGD002Z
h79S3Sik0hDB9g9AI33+vLxJQY/vMCV3CP8l+HgwcwUpt7900fVPRzfZwARAQAB
tBpNYXJpYSA8MTAzMjIxNjUyNkBydWRuLnJlPokCUGQTAQgAPBYhBENqPLlMhW22
6AlsBrD192608I7+BQJiZBMQAhsDBQsJCAcCAyICAQYVVCgkICwIEFgIDAQIEBwIX
gAAKRCRCw9fdutPCO/s0zD/0e3blYToBbTrkuYK9DQi+WsocplNUq0Bm3bdcKvgUv
HRutniLgNUAdlYPdR8XPrU4sovu9DdukIWyxCBsrRt5D1ttM10SLmN319eEXCMfg
b2A8BnP/ESxy22xbHQWmElvH7zY8dfVCw9dp0A0ruGIR+Q/F7bfV2ASV3zrejSvv
7FpW3boav0zQAS16a700Yp+RB39Gm5qytaBTWow75wBn0o4ZFIEIoWLXG18Ds6RCa
c0GVrUDN2Z5ntCQviKQWl8mctZahcLeglLxDzE2yv0nWIftczWnK30Fd1QsnEkpB
yI48Vph8zPqZqrzPMZjNNuqH7RKzIjpmcEIYPDVd1Zfd7yXcDRXgbmu8yH0hXKu
CvsbBAejrbvD9EgaxXQUHvKT59BEtG58uoVidv+Cd4Iu+8YYDFtNTGpuqX39z7L8
```

Данные о ключе GPG

{#fig:007 width=70% }

```
Y+enB0W0A7oxckfq6B6vBErdsZcpsFbC5wAXdwhQoluhALLjPo/LDRfjdF5Aybev  
esgVhBEclCdG2LdD7ASYAatBlhqipKWdSXgVpe0KLUTKslS403+UKpptA+u24E8  
q1P0X9U/TbR2Tn8RfEX0J4mk5LDbEVmT01Z//f1506vUlFMNnZFh+jfsLzk25eR5  
YlAZW2+zvJfxKE7qYW7KXw08m3bxF9rq7Pvw2RmDvPX3GEHN6WbgZuLruwL6xaTz  
PEij+tLnLXdRTEfulPrZjVcFFt0ZV4bo359p6ANJxDyvEjbHhflLqvXF4fusl7Sz  
Xy+8Pwq/ibBBaAZQDPLmotTq0f6jv7yxWf6Ks2BRVlTA3ItlLuxfbIi2RNUyNj3J  
USSC0sagBEBVPTJaellyF7biYT82UBpxKsN2GKbbRnHPyH8ITsFcD6MeWQFXx+27  
/bdXmsqGXZgPjBBjxPLHe6RPGIa5iEZhd1FfIC27vBdKn7/Xg1kD1LWSbDGzIBym  
UhC84Qh8wLvP0tzh60fh4ND7cyGldlLpQzF/9+lPC/qG/AYd6ahYyngBhXsAEQEA  
AYkCNgQYAQgAIBYhBENqPLMhW226AlsBrD192608I7+BQJiZBMQAhsMAAoJELD1  
92608I7+26kQAJ39P7ixWX7S6DdsVQaLACGBxKPydb2JVtdf165aIK5Io7F3uFzy  
JE/38Dt14Tw02xRzBatbbISncpUpU8R4qjE02lbk1xKH9VEMcWYP0CHHnUtmDUvA  
0DFaXuTdYHS4Zga1GEJEK6tD2Q0/u8oCDYbURAH80p2XNQkeimqVwvLkb6goHsLS  
9zDVx7+U/cX8nRHGEereekpqt2n9FhKWjQDTbqh4zTIuHLzpJZTBV+sb4q9b1amf  
ku/KrYkBVp+wnOrqhNSZXTcbSV/YQZ76FMmop0gb5Jrq4/DFSopmJwE+6osqs8A  
NHDB2MuwdQ0MkKfdyghl1E3UysBVZS0yTVj3WaN/RcP3K6F0h5VYp7UHnx8UmrI  
R30424Q1t3xv55xGIk3cvs6dQXYmYPFaJe+Ex2T2V1wl4aeIh8fFRbhM2m3N0MYF  
vHhTq0t8uGig+Nws5bb+lgWzBDYlSwkqVoc/ApCIpUrr0cND0iI36pAMVCfBQA7Q  
ESvGiQhn4jmNrZVx07y9eegzfQMLSMqQ6T9WQIKkeJtcqHLGLw+hLyhK10QJ9zLK  
6TFvbAcgvzy/fBBJxxFMeOrwJCKzrh/h6ZkQK2fWcrwQf5u0JKbjwYre1Kh6IFB/  
4Ql+U/5Utn1lcXdhNphsRF2ZM+XKdNkwH90fUBUa200iM1cTVF1x4Unk  
=2Ry0  
-----END PGP PUBLIC KEY BLOCK-----
```

Данные о ключе GPG

{#fig:008 width=70% }

7. Переходим в настройки GitHub, нажимаем на кнопку New GPG key и вставляем полученный ключ в поле ввода. Получаем: (рис. [-@fig:009])

```
[misamsonova@fedora ~]$ git config --global user.signingkey B0F5F76EB4F08EFE  
[misamsonova@fedora ~]$ git config --global commit.gpgsign true  
[misamsonova@fedora ~]$ git config --global gpg.program $(which gpg2)
```

Настройка подписей коммитов git

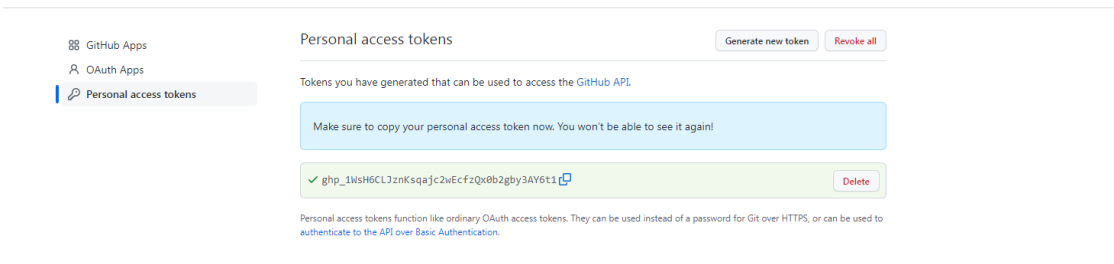
{#fig:009 width=70% }

8. Настраиваем gh и для начала авторизуемся в github. Далее создаём персональный токен, чтобы не нужно было входить в свою учётную запись через веб-сайт: (рис. [-@fig:010]) и (рис. [-@fig:011])

```
[misamsonova@fedora ~]$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Paste an authentication token
Tip: you can generate a Personal Access Token here https://github.com/settings/tokens
The minimum required scopes are 'repo', 'read:org', 'workflow'.
? Paste your authentication token: *****
error validating token: HTTP 401: Bad credentials (https://api.github.com/)
Try authenticating with: gh auth login
[misamsonova@fedora ~]$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Paste an authentication token
Tip: you can generate a Personal Access Token here https://github.com/settings/tokens
The minimum required scopes are 'repo', 'read:org', 'workflow'.
? Paste your authentication token: *****
- gh config set -h github.com git_protocol https
✓ Configured git protocol
✓ Logged in as misamsonova
[misamsonova@fedora ~]$
```

Вход в учётную запись github

{#fig:010 width=70% }



Создание токена для авторизации

{#fig:011 width=70% }

- После чего создаём репозиторий курса на основе шаблона следующим образом:(рис. [-@fig:012]) и (рис. [-@fig:013])

```
gh repo create study_2023-2024_mathmod --template=yamadharma/course-directory-student-template --public
git clone --recursive git@github.com:<owner>/study_2023-2024_mathmod.git mathmod
```

Создание репозитория курса на основе шаблона

{#fig:012 width=70% }


```
[misamsonova@fedora Операционные системы]$ git clone --recursive https://github.com/yamadharma/course-directory-student-template.git os-intro
Клонирование в «os-intro»...
remote: Enumerating objects: 35, done.
remote: Counting objects: 100% (35/35), done.
remote: Compressing objects: 100% (28/28), done.
remote: Total 35 (delta 7), reused 34 (delta 6), pack-reused 0
Получение объектов: 100% (35/35), 15.77 КиБ | 283.00 КиБ/с, готово.
Определение изменений: 100% (7/7), готово.
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/misamsonova/work/study/2021-2022/Операционные системы/os-intro/template/presentation»...
remote: Enumerating objects: 42, done.
remote: Counting objects: 100% (42/42), done.
remote: Compressing objects: 100% (34/34), done.
remote: Total 42 (delta 9), reused 40 (delta 7), pack-reused 0
Получение объектов: 100% (42/42), 31.19 КиБ | 3.12 МиБ/с, готово.
Определение изменений: 100% (9/9), готово.
Клонирование в «/home/misamsonova/work/study/2021-2022/Операционные системы/os-intro/template/report»...
remote: Enumerating objects: 78, done.
```

Создание репозитория курса на основе шаблона

{#fig:013 width=70% }

- Далее настраиваем каталога курса, то есть переходим в каталог курса командой `cd ~/work/study/2023-2024/“Математическое моделирование”/mathmod` и следующим образом удаляем лишние файлы, создаём необходимые каталоги и отправляем файлы на сервер: (рис. [-@fig:014])

```
[misamsonova@fedora os-intro]$ rm package.json
[misamsonova@fedora os-intro]$ make COURSE=os-intro
[misamsonova@fedora os-intro]$ git add .
```

Настройка каталога курса

{#fig:014 width=70% }

Вывод

В процессе выполнения данной лабораторной работы мы приобрели практические навыки применения средств контроля версий и вспомнили, как работать с git.