

Отчёт по лабораторной работе №1

Математическое моделирование

Настройка рабочего пространства. Система контроля версий Git. Язык разметки Markdown

Выполнила: Коняева Марина Александровна,
НФИбд-01-21, 1032217044

Содержание

Цель работы	1
Задание	1
Теоретическое введение	1
Выполнение лабораторной работы	2
Вывод.....	7
Список литературы. Библиография	7

Цель работы

Настроить рабочее пространство для лабораторной работы. Изучить систему контроля версий Git и язык разметки Markdown.

Задание

Создать директорию, создать репозиторий, настроить связь между своим компьютером и GitHub по SSH-ключу. При помощи Makefile сконвертировать из файла .md файлы отчетов в форматах docx и pdf. Запустить все готовые отчеты на Github.

Теоретическое введение

Git — система управления версиями с распределенной архитектурой. В отличие от некогда популярных систем вроде CVS и Subversion (SVN), где полная история версий проекта доступна лишь в одном месте, в Git каждая рабочая копия кода сама по себе является репозиторием. Это позволяет всем разработчикам хранить историю изменений в полном объеме.

Markdown — облегчённый язык разметки, созданный с целью обозначения форматирования в простом тексте, с максимальным сохранением его читаемости

человеком, и пригодный для машинного преобразования в языки для продвинутых публикаций (HTML, Rich Text и других).

Назван

ие

коман

ды

Описание команды

git clone	Клонирование репозитория на ПК
--------------	--------------------------------

git commit -m "Initial Commit "	Оставление коммита
--	--------------------

git push	Загрузка изменений на гит
-------------	---------------------------

make	Конвертация файла .md
------	-----------------------

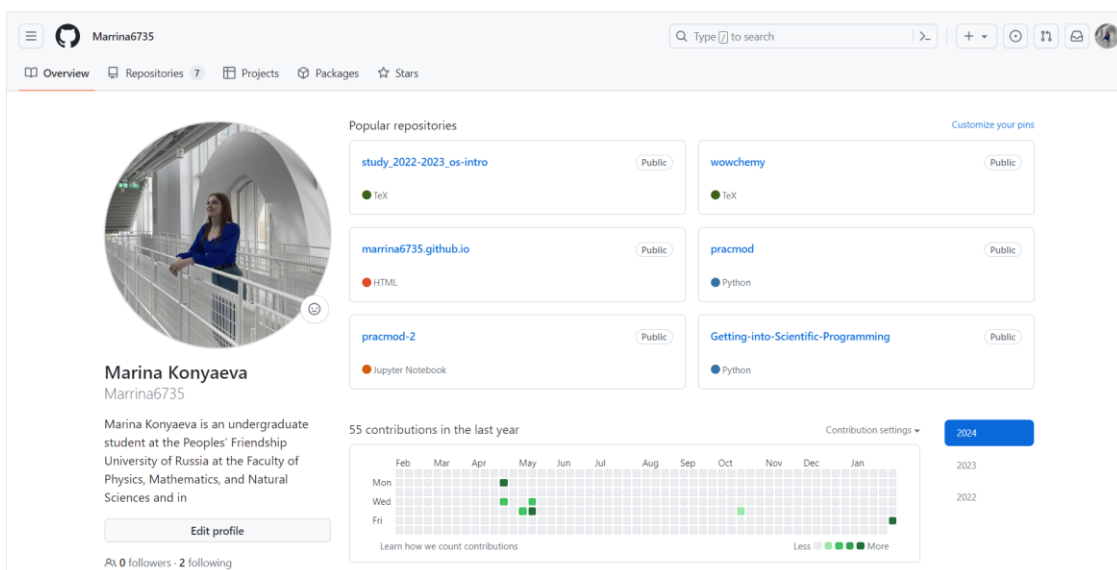
Выполнение лабораторной работы

1. Создадим директорию на своем компьютере по шаблону:

```
~/work/study/  
├── <учебный год>/  
│   ├── <название предмета>/  
│   │   └── <код предмета>/
```

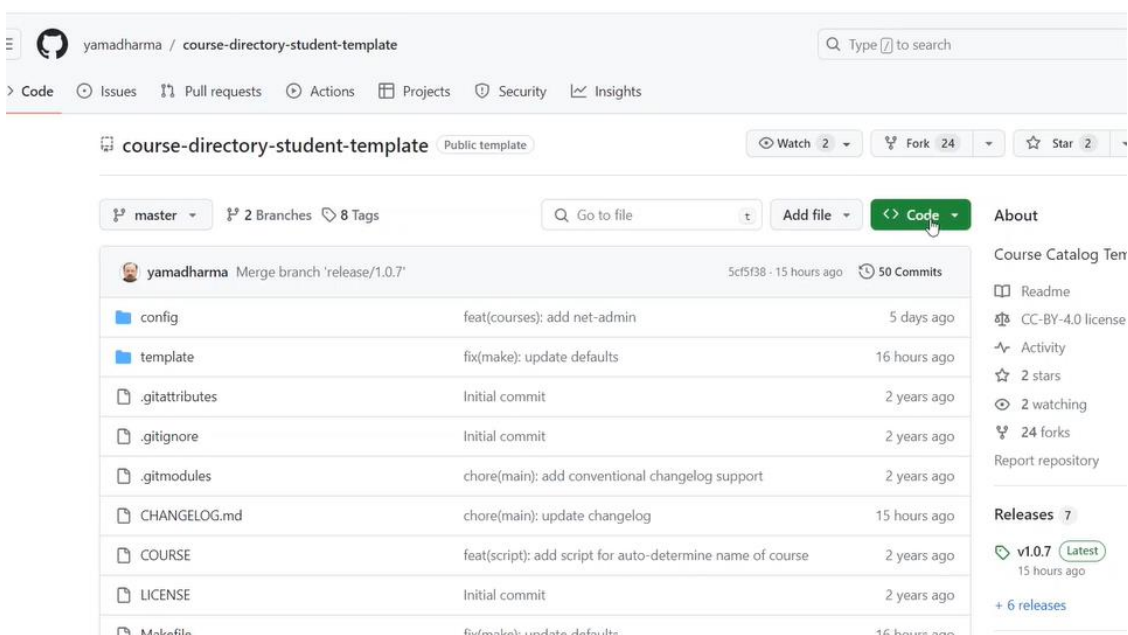
(рис. 1. Шаблон директории)

2. Авторизируемся на Github:



(рис. 2. Авторизация)

3. Перейдем к шаблону репозитория и создадим по нему свой новый репозиторий:



(рис. 3. Шаблон репозитория)

4. Создадим и настроим репозиторий:

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 (*).

Repository template

 yamadharma/course-directory-student-template ▾

Start your repository with a template repository's contents.

☒ Include all branches

Copy all branches from yamadharma/course-directory-student-template and not just the default branch.

Owner *

 Marrina6735 ▾

Repository name *

mathmod

✓ mathmod is available.

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

Description (optional)

Математическое моделирование 2023-2024

☒ 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. 4. Создание репозитория)

5. Установим make:

```
PS C:\WINDOWS\system32> choco list
Chocolatey v2.2.2
Chocolatey 2.2.2
1 packages installed.
PS C:\WINDOWS\system32> choco install make
Chocolatey v2.2.2
Installing the following packages:
make
By installing, you accept licenses for the packages.
Progress: Downloading make 4.4.1... 100%
make v4.4.1 [Approved]
make package files install completed. Performing other installation steps.
ShimGen has successfully created a shim for make.exe
The install of make was successful.
Software installed to 'C:\ProgramData\chocolatey\lib\make'
Chocolatey installed 1/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).
Enjoy using Chocolatey? Explore more amazing features to take your
experience to the next level at
https://chocolatey.org/compare
PS C:\WINDOWS\system32>
```

(рис. 5. Установка make)

6. Установим git:

```

PS C:\WINDOWS\system32> choco install git
Chocolatey v2.2.2
Installing the following packages:
git
By installing, you accept licenses for the packages.
Progress: Downloading chocolatey-compatibility.extension 1.0.0... 100%

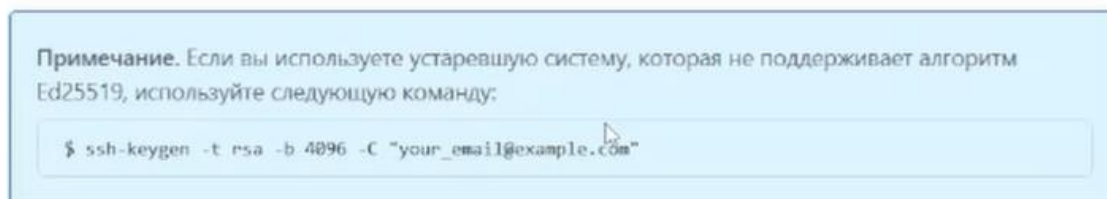
chocolatey-compatibility.extension v1.0.0 [Approved]
chocolatey-compatibility.extension package files install completed. Performing other installations...
Installed/updated chocolatey-compatibility extensions.
The install of chocolatey-compatibility.extension was successful.
Software installed to 'C:\ProgramData\chocolatey\extensions\chocolatey-compatibility'
Progress: Downloading chocolatey-core.extension 1.4.0... 100%

chocolatey-core.extension v1.4.0 [Approved]
chocolatey-core.extension package files install completed. Performing other installations...
Installed/updated chocolatey-core extensions.
The install of chocolatey-core.extension was successful.
Software installed to 'C:\ProgramData\chocolatey\extensions\chocolatey-core'
Progress: Downloading git.install 2.43.0... 100%

```

(рис. 6. Установка git)

7. Запросим SSH-ключ:



(рис. 7. Запрос ключа)

8. Получим SSH-ключ:

```

PS C:\Users\User\OneDrive\Документы\2023-2024\Математическое моделирование\mathmod> git -v
git version 2.43.0.windows.1
PS C:\Users\User\OneDrive\Документы\2023-2024\Математическое моделирование\mathmod> ssh-keygen -t rsa -b 4096 -C koniaeva2003@mail.ru
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\User/.ssh/id_rsa):
Created directory 'C:\Users\User/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\User/.ssh/id_rsa.
Your public key has been saved in C:\Users\User/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:NqH1u/gAmKKnQ5t48gDe9kT9WvxjMoAJ5gwvNWD7nQg koniaeva2003@mail.ru
The key's randomart image is:
+---[RSA 4096]-----+
|
|  .  .
|  .  .O+O .
|  O +E++O +
|  OO +=++O.=
|  * ..+O.O$ .
|  +O . O O + +
|  += . . + .
|  O=OO .
|  .+.+.
+---[SHA256]-----+
PS C:\Users\User\OneDrive\Документы\2023-2024\Математическое моделирование\mathmod>

```

(рис. 8. Получение ключа)

9. Получим id SSH-ключа:

```

PS C:\WINDOWS\system32> ssh-add C:\Users\User/.ssh/id_rsa
Identity added: C:\Users\User/.ssh/id_rsa (koniaeva2003@mail.ru)
PS C:\WINDOWS\system32>

```

(рис. 9. Получение id ключа)

10. Добавим связку по SSH-ключу на сайт:

Marina Konyaeva (Marrina6735)
Your personal account

- Public profile
- Account
- Appearance
- Accessibility
- Notifications

Access

- Billing and plans
- Emails
- Password and authentication
- Sessions
- SSH and GPG keys**
- Organizations
- Enterprises
- Moderation

Code, planning, and automation

- Repositories

Add new SSH Key

Title

Key type
Authentication Key

Key
Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp521', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com' or 'sk-ssh-ed25519@openssh.com'

Add SSH key

(рис. 10. Добавление ключа)

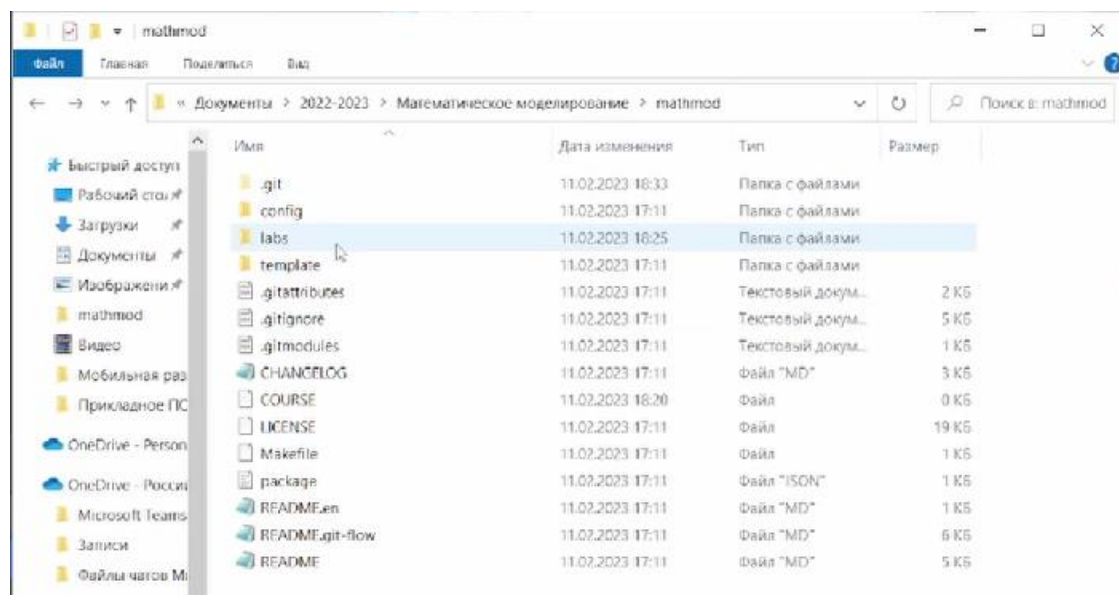
11. Клонировем репозиторий:

```
git clone --recursive git@github.com:malashenkomv/mathmod
```

```
PS C:\Users\Марина\Documents\2022-2023\Математическое моделирование\mathmod> git clone --recursive git@
```

(рис. 11. Клонирование)

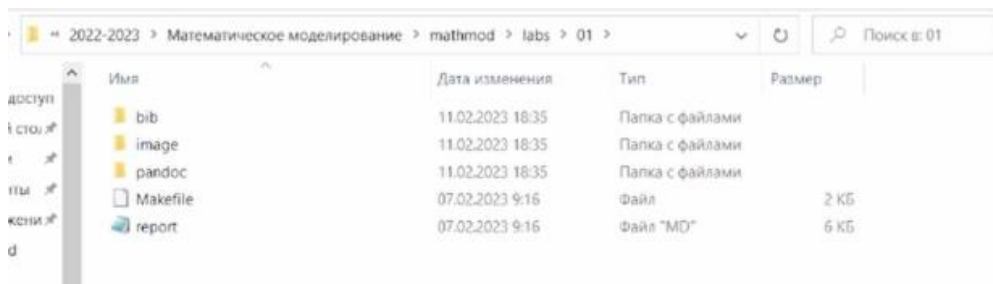
12. Репозиторий скопирован:



(рис. 12. Репозиторий в директории)

Создадим папку Labs с внутренней папкой Lab01. Внутри папки Lab01 папки report и presentation.

13. Вид папки Lab01/report:



(рис. 13. Папка лабораторной работы)

14. Конвертируем .md файл в docx командой make:

```
PS C:\Users\Марина\Documents\2022-2023\Математическое моделирование\mathmod\labs\01> make
pandoc "report.md" --filter pandoc/filters/pandoc_fignos.py --filter pandoc/filters/pandoc_eqnos.py --filter pandoc/
/filters/pandoc_secnos.py --number-sections --citeproc -o "report.docx"
--main--: Bad reference: @fig:001.
pandoc "report.md" --filter pandoc/filters/pandoc_fignos.py --filter pandoc/filters/pandoc_eqnos.py --filter pandoc/
/filters/pandoc_secnos.py --pdf-engine=lualatex --pdf-engine-opt=--shell-escape --citeproc --number-sections -o "r
```

(рис. 14. Конвертация в docx)

15. Затем получили docx файл, а для конвертации .md файла в pdf потребуется установка TeX Live. Установим MiKTeX как альтернативу TeX Live для LaTeX, после конвертируем .md файл в pdf командой:

```
pandoc report.md -o report.pdf --pdf-engine=lualatex -V mainfont="Times New Roman" -V sansfont="DejaVu Sans" -V monofont="DejaVu Sans Mono"
```

18. Получили pdf файл и итоговый вид папки отчета лабораторной работы, также конвертируем .md файл презентации в pdf презентации командой:

```
pandoc presentation.md -o presentation.pdf --pdf-engine=lualatex -V mainfont="Times New Roman" -V sansfont="DejaVu Sans" -V monofont="DejaVu Sans Mono" -t beamer --slide-level=2
```

21. Получили pdf файл презентации, отправим все изменения на GitHub командами и все изменения отправляем в репозиторий с указанным коммитом.

Вывод

Мы настроили рабочее пространство для лабораторной работы. Изучили систему контроля версий Git и язык разметки Markdown.

Список литературы. Библиография

- Документация по Git: <https://git-scm.com/book/ru/v2>
- Документация по Markdown: <https://learn.microsoft.com/ru-ru/contribute/markdown-reference>

- Документация по MiKTeX: https://kpfu.ru/staff_files/F2077692752/Inst_MiKTeX.pdf