

**Министр науки и высшего образования Российской  
Федерации**

**Федеральное государственное автономное  
образовательное учреждение высшего образования**

**«Национальный исследовательский университет  
ИТМО»**

Факультет информационных технологий и программирования

**i. Лабораторная работа № 1**

**Выполнила студент группы № М3114**

Нгуен Нгуен Тхинь

**Подпись:**

**Проверил:**

Санкт-Петербург, 2021г

Лабораторная работа предназначена для углубленного ознакомления с возможностями Git.

Подразумевается, что у студента уже есть собственный GitHub аккаунт и репозиторий по предмету - программирование.

Задача студента произвести клонирование репозитория и выполнить задания на копии репозитория.

- 1) Организовать модель ветвления, где каждая лабораторная - отдельная ветка.
- 2) Отправка, принятие Pull Request - ваш код должен пройти код ревью вашего одногруппника. Обязательное наличие комментариев, правок. Разрешается, в целях обучения, допущение преднамеренных ошибок в коде.
- 3) Использование тегов для обозначения версий.
- 4) Установить приложение графического интерфейса для гит.

5) Новое клонирование репозитория из пункта 1. И изначального коммита, используя модель ветвления GitFlow, реорганизуем свои ветки. Для этого использовать только команды гита. Запрещается копировать и удалять в папках. В отчет необходимо предоставить последовательность и описание используемых команд.

Данная лабораторная работа допускается для сдачи при наличии не менее 3 лабораторных работ по программированию

## 1) Организовать модель ветвления, где каждая лабораторная - отдельная ветка.

```
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming
(A)
$ git branch Lab1

Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming
(main)
$ git checkout Lab1
Switched to branch 'Lab1'

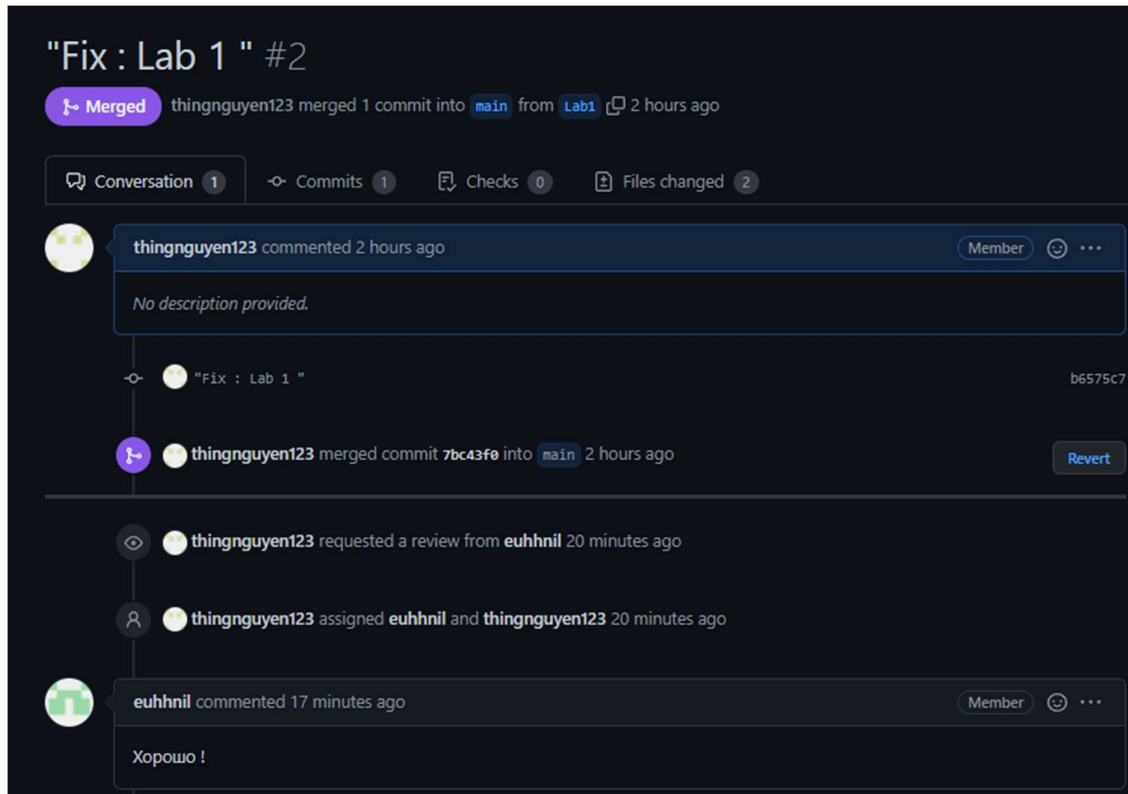
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming
(Lab1)
$ git add .

Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming
(Lab1)
$ git status
On branch Lab1
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   A.txt

Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming
(Lab1)
$ git commit
[Lab1 a699499] " Lab 1 "
 1 file changed, 44 insertions(+)
 create mode 100644 A.txt

Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming
(Lab1)
$ git push -u origin Lab1
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 367.03 KiB | 30.58 MiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
remote:
remote: Create a pull request for 'Lab1' on GitHub by visiting:
remote:   https://github.com/Github-2101/Progamming/pull/new/Lab1
remote:
To https://github.com/Github-2101/Progamming
 * [new branch]      Lab1 -> Lab1
Branch 'Lab1' set up to track remote branch 'Lab1' from 'origin'.
```

- 2) Отправка, принятие Pull Request - ваш код должен пройти код ревью вашего одноклассника. Обязательное наличие комментариев, правок. Разрешается, в целях обучения, допущение преднамеренных ошибок в коде.



- 3) Использование тегов для обозначения версий.

```
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming (Lab1)
$ git tag -a "V.1.0" -m " This is the first version"

Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming (Lab1)
$ git tag
V.1.0
```

```

Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming
(Lab1)
$ git show V.1.0
tag V.1.0
Tagger: thingnguyen123 <thinhisssomething22@gmail.com>
Date: Fri Nov 26 16:59:50 2021 +0300

This is the first version

commit d64d9e8e4a48687159709a5d43455bcd40872162 (HEAD -> Lab1, tag: V.1.0, origin/Lab1)
Author: thingnguyen123 <thinhisssomething22@gmail.com>
Date: Fri Nov 26 16:53:09 2021 +0300

    " Lab 1 "

diff --git a/A.txt b/A.txt
index 67aa0df..f0de669 100644
--- a/A.txt
+++ b/A.txt
@@ -1,44 +1,14 @@
-#include <bits/stdc++.h>
-
+#include<iostream>
+
+using namespace std;
+...skipping...
tag V.1.0
Tagger: thingnguyen123 <thinhisssomething22@gmail.com>
Date: Fri Nov 26 16:59:50 2021 +0300

This is the first version

commit d64d9e8e4a48687159709a5d43455bcd40872162 (HEAD -> Lab1, tag: V.1.0, origin/Lab1)
Author: thingnguyen123 <thinhisssomething22@gmail.com>
Date: Fri Nov 26 16:53:09 2021 +0300

    " Lab 1 "

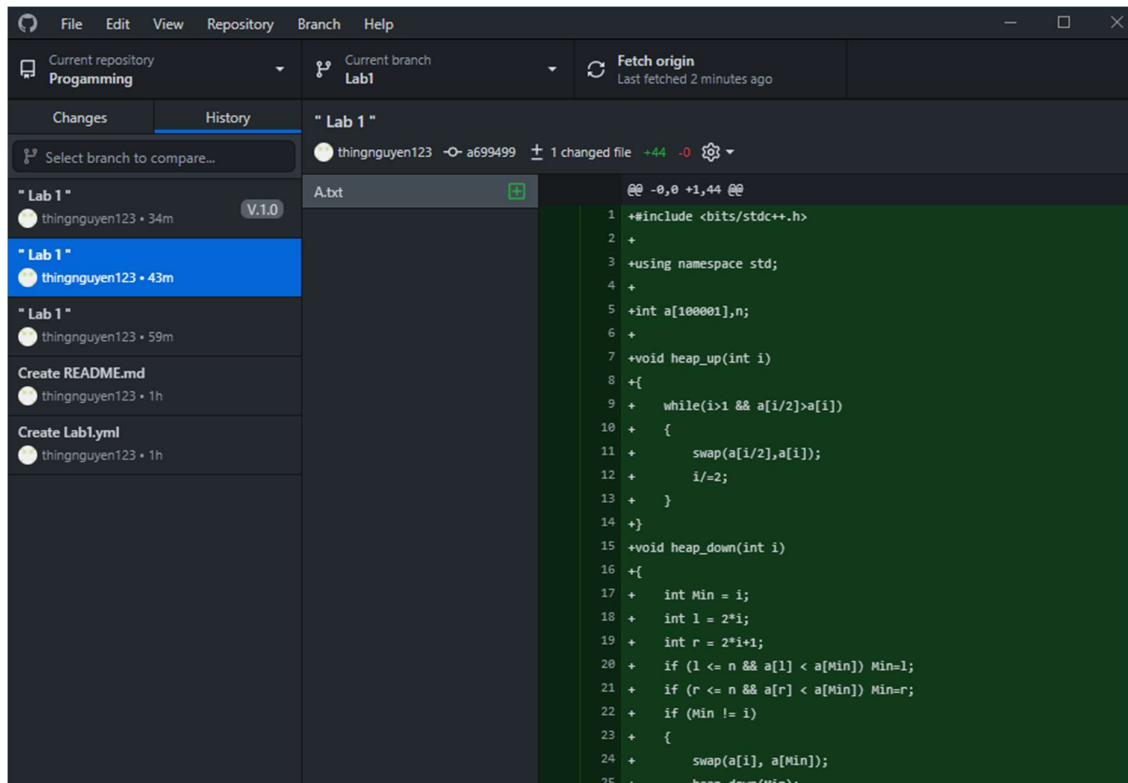
diff --git a/A.txt b/A.txt
index 67aa0df..f0de669 100644
--- a/A.txt
+++ b/A.txt
@@ -1,44 +1,14 @@
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming (
Lab1)
$ git tag
V.1.0

Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Programming (
Lab1)
$ git push origin V.1.0
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 180 bytes | 180.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Github-2101/Programming
 * [new tag]          V.1.0 -> V.1.0

```

#### 4) Установить приложение графического интерфейса для гит.

-установить клиент github GUI



**5) Новое клонирование репозитория из пункта 1. И изначального коммита, используя модель ветвления GitFlow, реорганизуем свои ветки. Для этого использовать только команды гита. Запрещается копировать и удалять в папках. В отчет необходимо предоставить последовательность и описание используемых команд.**

**Шаг 1. Подготовка**

1. Создать Organization

Tell us about your organization

## Set up your organization

**Organization account name \***

Github-2101 ✓

This will be the name of your account on GitHub.  
Your URL will be: <https://github.com/Github-2101>.

**Contact email \***


thinissomething22@gmail.com ✓

**This organization belongs to: \***

☒ **My personal account**  
I.e., thingnguyen123

☐ **A business or institution**  
For example: GitHub, Inc., Example Institute, American Red Cross

**Verify your account**



**Next**

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.

2. Добавляем участников в учебную группу, например на картинке:  
(Допускаются члены в review source code и оцените, обсудите это)

Start collaborating

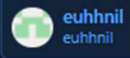
# Welcome to Github-2101

## Add organization members

Organization members will be able to view repositories, organize into teams, review code, and tag other members using @mentions.

[Learn more about permissions for organizations →](#)

Search by username, full name or email address



Complete setup

[Skip this step](#)

Tell us about your team

How many people do you expect to actively work within this GitHub organization?

0



1-5



6-15



25+



What type of work do you plan to use this organization for?

Please select all that apply



Open source projects



Education projects



Personal projects



Work projects

Other

Please describe

## What do you expect to do on GitHub in the next seven days?

Choose up to 3



Manage code



Collaborate on code



Plan and track work



Set up CI/CD



Improve security



Other

Please describe

## Do you have an existing repository for your project?

Yes



No



Submit



### 3. Создать 1 хранилище


## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


---

Owner <sup>\*</sup>

Repository name <sup>\*</sup>

 Github-2101 ▾


/

Programming 


Great repository names are short and memorable. Need inspiration? How about **sturdy-fortnight**?

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.

---

**Initialize this repository with:**  
Skip this step if you're importing an existing repository.

☐ **Add a README file**  
This is where you can write a long description for your project. [Learn more.](#)




☐ **Add .gitignore**  
Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**  
A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository

Search or jump to...

[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)

Github-2101 / Programming Public

[Watch](#) 0 [Star](#) 0 [Fork](#) 0

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

main ▾ 1 branch 0 tags

Go to file

Add file ▾

Code ▾

thinguyent23 Create README.md

escccc now 2 commits

M3114\_Hyey\_Hyey\_Tuon Create Lab1.yml

1 minute ago

README.md Create README.md

now

README.md

Programming

About

No description, website, or topics provided.

[Readme](#)

Releases

No releases published

[Create a new release](#)

Packages

No packages published

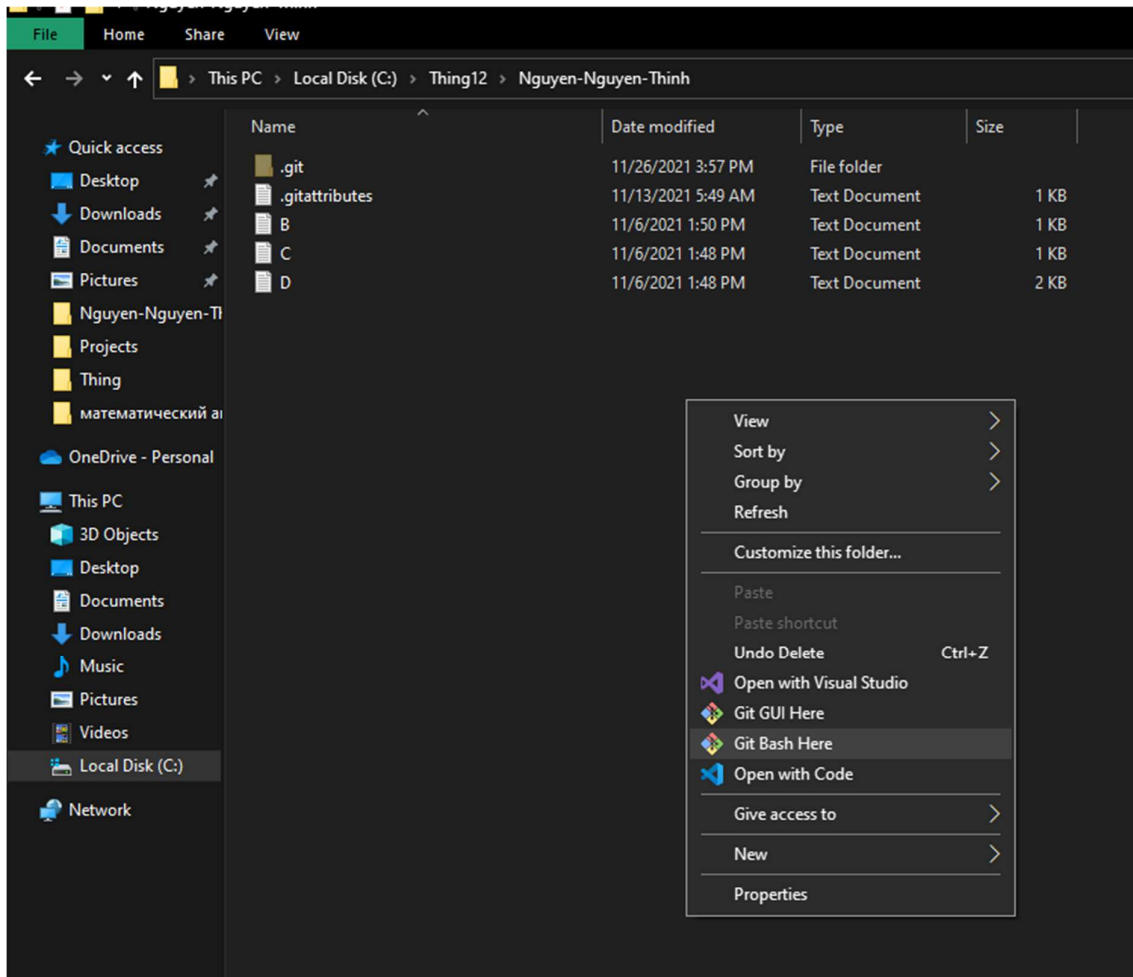
[Publish your first package](#)

© 2021 GitHub, Inc.

[Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

## Шаг 2. Решение

- Создайте папку на своем компьютере и используйте Git Bash



- Войдите в учетную запись GitHub в Git Bash

```
MINGW64/c/Thing12/Nguyen-Nguyen-Thinh
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh (main)
$ git config --global user.name thingnguyen123
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh (main)
$ git config --global user.email thinhsomething22@gmail.com
```

- git clone : Clone repository Programming

```
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh (main)
$ git clone https://github.com/Github-2101/Programing
```

- git add

```
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh (main)
$ git add .
```

- git commit :

```
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Progammimg (
main)
$ git commit
[main f2e38cd] " Program B "
1 file changed, 42 insertions(+)
create mode 100644 B.txt
```

- реорганизуем свои ветки

```
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Progammimg (
main)
$ git checkout Lab1
Switched to branch 'Lab1'
Your branch is up to date with 'origin/Lab1'.
```

```
Thing Nguyen@DESKTOP-C3PC0HH MINGW64 /c/Thing12/Nguyen-Nguyen-Thinh/Progammimg (
Lab1)
$ git merge Lab1
Already up to date.
```