Seminar on Computer-assisted mathematics

Judith Ludwig and Florent Schaffhauser

Heidelberg University - Summer semester 2025

Session 3 - May 8, 2025

Today, we will learn about Git and GitHub.



- What Git is: a version control system originally authored by Linus Torvalds in 2005.
- Git's main characteristics are that it is distributed, fast, and efficient in its use of memory space.
- What Git does: it tracks changes in computer files.
- What Git is often used for: coordinating work among people working collaboratively on a project.

GitHub

GitHub is a private company that was started in 2008. It became a subsidiary of Microsoft in 2018 (alternatives are: GitLab, Gittea, etc).



- What GitHub is: an Internet hosting service for software development and version control using Git.
- What GitHub does: it provides infrastructure and automation for large projects (such as Mathlib).
- What GitHub is usually used for: host collaborative software development projects.

Repositories

- The successive versions of a project administered via Git are stored in a **Git repository**, which in practice is just the hidden folder .git in your workspace.
- A **GitHub repository** (or repo) means something slightly different: it is basically an interface showing the latest version of a project administered via Git. But the whole history of the project is also accessible via that interface.
- A public GitHub repository can be **cloned** or **forked** by anyone who wants to use it:
 - If you clone it, then you will be working directly on it (if you are authorised to do so).
 - If you fork it, then you create a new copy and you can decide later if you want to contribute to the original repo (requires managing two repos).

Our seminar's repository

- Our seminar's repo is hosted on GitHub but it is a private repo, so we will add you as collaborators.
- To do that, we will need you GitHub username to do that (please send it to us via Zulip, either on the channel or via DM).
- We want you to clone the repo (not fork it).

E C matematifio / CompAssistedMath202	G. Type // to	search	8 • + • O n 🕰 🖲
↔ Code ⊙ Issues 11 Pull requests 1) Actions 🖽 Projects 🖽 Wiki 🛈 Security 🗠	⊻ Insights 🛞 Se	ttings
CompAssistedMath2025 (Private)		⊙ Unwatch 3	▼ ♥ Fork 0 ▼ ☆ Star 2 ▼
P main → P 13 Branches 🛇 0 Tags	Q. Go to file	+ Code +	About 🛞
(e) matematifio fix: correct comments and formation	tting in Logic.lean 🗸 442fded - yesterdar	y 🕚 49 Commits	Seminar on Computer-assisted mathematics, Heidelberg University,
CompAssistedMath2025	Initial commit	last month	Summer 2025
PracticeFiles	fix: correct comments and formatting in Logic.lean	yesterday	matematino.gitnub.io/Compassistedm
QR_codes	feat: add script to generate QR code for Zulip invitatio	3 weeks ago	C Decidere
Slides	feat: update Session 3 presentation slides	yesterday	Apache-2.0 license
index_files/libs	Add syntax highlighting styles and implement tabset f	3 weeks ago	Ar Activity
🗅 .gitignore	feat: add Slides_prep to .gitignore to exclude slide pre	3 weeks ago	
CompAssistedMath2025.lean	Initial commit	last month	Y 0 forks

Branches

- A characteristic feature of Git is the existence of branches.
- Branches allow for distributed and parallel workflows that can later be merged back into the main workflow.
- Use cases: branches can for instance be used to develop new features or fix bugs.



Collaborating on a repository

- Branches are very flexible, allowing for several patterns.
- They can be used for collaborative work, typically on software development projects.



- In the seminar, you will create your own branch to:
 - Work independently on the Lean files we provide for you.
 - 2 Submit assignments and project work via **pull requests**.

Pull requests are used constantly when collaborating on a project.

- A pull request is a proposal to merge a set of changes from one branch into another (often into the main branch).
- In a pull request, collaborators can review and discuss the proposed set of changes before they integrate the changes into the main codebase.
- Pull requests display the differences, or diffs, between the content in the source branch and the content in the target branch.

For more details, we refer you to the GitHub documentation.

GitHub Desktop

- If you are just getting started with Git, you may find it convenient to use GitHub Desktop.
- GitHub Desktop is an application that enables you to perform all basic Git operations via a graphical interface.
- For instance, if you go to a repo's webpage on GitHub, you have the option to open that repo in GitHub Desktop, which means cloning it on your machine.



Using Git in an IDE

- You can also use an IDE such as VS Code to perform Git operations on your repo. This is quite convenient but perhaps slightly confusing at first.
- We will see next several basic operations that you will need to perform on the seminar repo. If you are a beginner at this, GitHub Desktop provides a simplified framework to handle these tasks.



Cloning a repo

- Cloning a repo means creating a copy of it on your machine. When you do that, you also copy the entire history of the repo.
- The command line to clone a repo is:

git clone https://github.com/matematiflo/CompAssistedMath2025

• You can also clone a remote repo (or add a local one) from within GitHub Desktop. If you click on History, it should look like this:

Current Repository CompAssistedMath2025	같 Current Branch ← C Fetch origin Last fetched 17 minutes ago		
Changes History	fix: correct comments and formatting in Logic.lean ‡		
P Select Branch to Compare	matematic of welded () +17-21		
fix: correct comments and formatting in Logic.lean	1 changed file		
📵 matematifio - yesterday	PracticeFiles/Logic.lean	۰	
feat: update Session 3 presentation slides (இ) matematifio - yesterday			
feat: update seminar schedule with Git and GitHub introduction and (a) matematifio - yesterday			
feat: add Session 3 presentation slides in PDF format (e) matematifio - yesterday			
fix: update copyright year to 2025 in Logic.lean () matematifio - yesterday			
feat: add link to md file for Logic and convergent sequences in REA (a) matematifio - yesterday			
feat: add md version for Logic and convergent sequences assignment (a) matematifio - yesterday			
added exercises on convergent sequences (#1) () judithludwig, matematifio - yesterday			

Creating a branch

- Creating a branch enables you to use the repo's history to test or develop new features.
- We will use branches a lot for the seminar (either just to practice, or work on assignments, and later to develop your projects).
- The command line to create a new branch is:
 - git branch Assignment1-FlorentJudith



Tracking a file's history

- In Git, there are in principle three workspaces:
 - The working tree, where you work on the files.
 - The index tree, where you prepare the files you want to add to the repo.
 - **③** The **repository**, where you store the various versions of your files.
- To start tracking a file's history, you need to add that file to the index. GitHub Desktop does that automatically (you have to manually uncheck a file if you do not want it to be tracked).
- The above also applies when a file is modified (the modified version may or may not be added to the index). The command line to index a file is the following:

```
git add . -- add all files in the working tree
git add Logic.lean -- only adds Logic.lean
```

- Committing some files means generating a new version of the repo that includes those (new or modified) files.
- By default, GitHub Desktop adds all modified files to the commit, but you can uncheck those that you do not want to add.
- When you commit files to a repo, you must include a commit message. It is good practice to add an *informative* commit message when you do that.

```
git commit -m "Completed Exercise 1 of Assignment 1."
```

• The interface in GitHub Desktop lets you enter a commit message as well as more detailed explanations.

Assignment

Our next session is on May 15th. Your assignment for next week is:

- Install Git (and, if you want, GitHub Desktop) on your machine (if you have VS Code on your machine, then Git should already be installed).
- Clone the seminar repository (you will need an invitation to see the repo on GitHub, so send us your GitHub username via Zulip).
- Immediately create a branch and switch to it. The name of the branch should enable us to identify you. If you have never created a branch, GitHub Desktop can assist you with it.
- Complete as much as you can of the Logic.lean file (there are various sorries to fill).
- Commit your work to your branch of the repo (you can do this as many times as you want during the completion of this assignment).
- Merge your branch back into the main branch (as the main branch of our repo is protected, you will need to open a pull request to do that, which GitHub Desktop can help you with).

If you get stuck during any of that, do not hesitate to ask for help on Zulip!