Tools for productive hacking

İpek Özdemir

1ST MRS HACKATHON
TORONTO
2023

JOHNS HOPKINS
SCHOOL *of* MEDICINE

iozdemi1@jh.edu
twitter.com/ipekoezdemir

‘Hacking’ in scientific software development

- Never a straight line
- No code without bug
- Constant struggle with the purpose of the software development tools
- Where to start if there is no graphical user interface with buttons to click

A lot of blogs/tutorials explaining all the tools needed for software development

➢ Today’s purpose: **struggle together produce together!**

➢ Just like a neuroscientist would say: **fire together wire together!**

What tools and why to use them?

- Basic unix commands
  - Repetetive tasks can be automated
  - Most computer clusters and high performance hardware

- What can you do with basic unix command?
  - Faster Navigating/Copying and Removing Files and Directories
  - Loops and Conditional Statements
  - Shell Scripts and Automated Analysis

https://andysbrainbook.readthedocs.io/en/latest/unix/Unix_Intro.html#

Hands on unix tutorial

- 10 min.

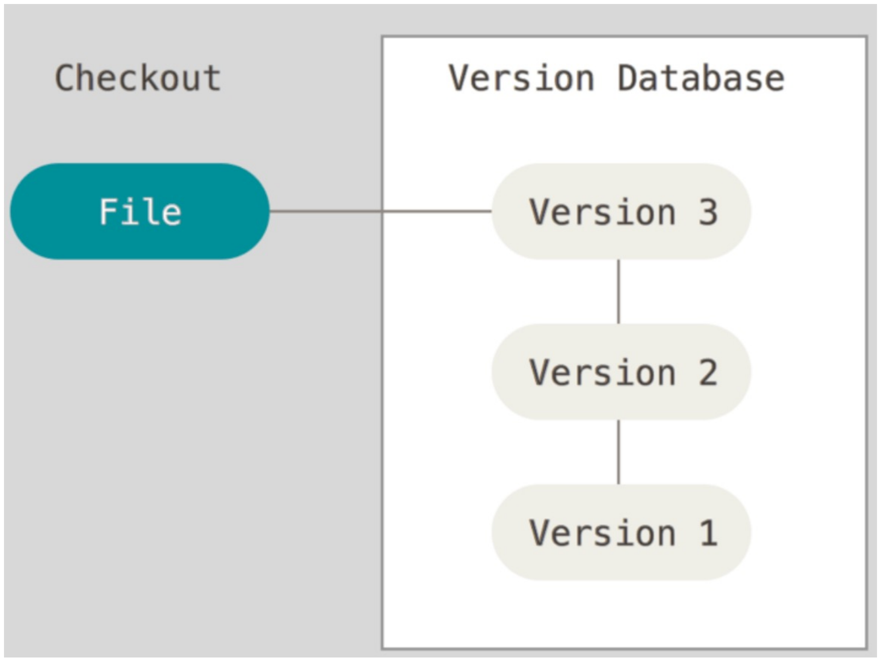
What did you learn?

- uname of a cluster to learn its OS
- cd
- rm
- mv
- The flags of a command
- Man pages for documentation
- Useful resource for a neuroscientist to learn more about unix

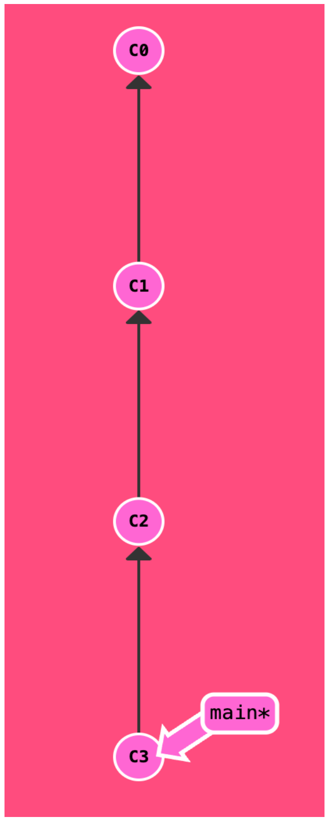
How do you “hack” your code?

- Design – write – test – success – document – process – present
- Write – test – success – process – present – forget
- Write – test – fail – improve – test – fail – improve – test – fail – did I improve this in my previous trial already?? – improve/fail loop
- Write – test – fail – give it a **`version number`** and **`short comment`** – improve – test – fail – give it a **`version number`** and **`short comment`** – improve – test fail – give it a **`version number`** and **`short comment`** – improve (add the first improvement and third together but remove the second one) – test – success!

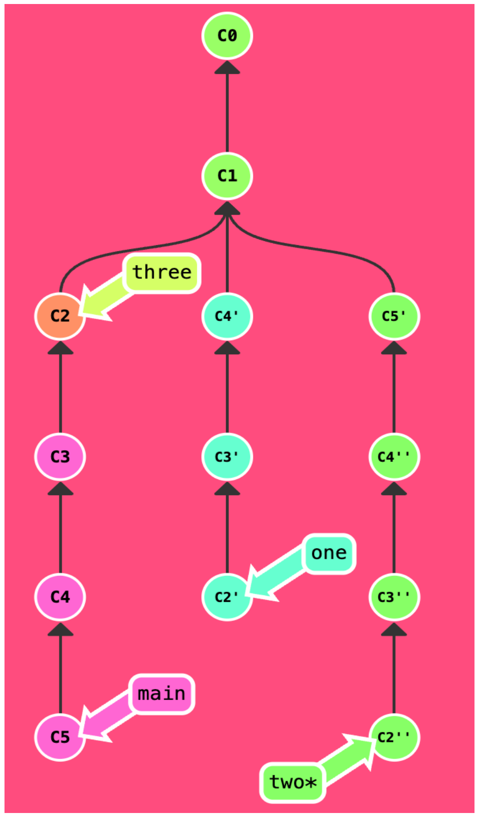
‘Hacking’ Like a Legend using git



simple



complex



https://git-scm.com/book
https://learngitbranching.js.org/

Hands on git tutorial

- 10 min.

What did you learn?

- How to commit
- How to create a branch
- What is a HEAD
- How to go back to an earlier commit
- Difference between git / gitlab / github
- Interactive resource for a neuroscientist to learn more about git