

Git 指令與觀念 (一)

與大家合作做專案之前必學技能之一



RyanLin 林昕銳
(@ryanlinjui)

上課時間：19:00 - 20:30
2025.04.01 Tue.

```
org = filterByOrg ? study.lead_organization === filterByOrg : true  
status = filterByStatus ? study.status === filterByStatus : true  
(matchStatus) {  
  // ...  
}  
  
function filterStudies(studies, filterByOrg, filterByStatus) {  
  return studies.filter(study => {  
    return filterByOrg === study.lead_organization &&  
    filterByStatus === study.status &&  
    (matchStatus) {  
      // ...  
    }  
  })  
}
```

本日Outline

// 19:00 - 20:30

| | | |
|-------|-------------------|-----|
| 前置準備 | 踏入 Git 的第一步 | p3 |
| 19:00 | 關於「多人開發專案」 | p5 |
| 19:15 | Git 與版本控制的基本觀念 | p11 |
| 19:30 | 用 Git 指令上傳你的成果吧 | p17 |
| 19:50 | 中場休息 | |
| 20:00 | GitHub: 全球工程師開源平台 | p27 |
| 20:10 | 用 GitHub 貢獻你的成果吧 | p36 |
| 20:25 | 結語 | p41 |



踏入 Git 的第一步

前置準備

- 註冊 GitHub 帳號 [\[Link\]](#)
- 下載 Git 相關工具 [\[Link\]](#)
- 得到一個好用的文件編輯器，這裡一律推薦 VSCode [\[Link\]](#)
- 拿取 GitHub API Token, 就像你的專屬鑰匙 [\[Link\]](#)

(以上前置準備一定要在課程開始前準備好哦！我 18:00 就會在現場，有問題的人可提早到現場問問題)

關於「多人開發專案」

什麼是「開發程式專案」？

眾多的程式碼的集合體，由一人或多人共同撰寫而成

- 網頁設計
- 手機 APP
- 遊戲開發

(只要是需要撰寫程式碼的系統或應用都可以稱之)



EXPLORER

- SOURCE CONTROL REPOSITORIES
- SOURCE CONTROL
- PRIVY-COUNCIL
 - src
 - lib
 - component
 - server
 - generate.ts
 - log.ts
 - prisma.ts
 - verify.ts
 - !18n.ts
 - routes
 - @[namespace]
 - ask
 - auth
 - logout
 - +page.server.ts
 - +page.svelte
 - create-team
 - +page.server.ts
 - +page.svelte
 - home
 - +server.ts
 - trending
 - +layout.server.ts
 - +layout.svelte
 - +layout.ts
 - +page.svelte
 - app.css
 - OUTLINE
 - TIMELINE

src > lib > server > TS log.ts > log

```
1 import debug from "debug";
2
3
4 const console_log = debug("app:log");
5
6 export async function log(do_what: string, done_by: string, namespace: string) {
7   const user_info = await prisma.user.findUnique({
8     where: {
9       email: done_by,
10     },
11   });
12   console_log(user_info);
13
14   if (user_info?.namespace_name === namespace) {
15     await prisma.log.create({
16       data: {
17         content: do_what,
18         time: new Date(),
19         user_id: user_info?.id,
20       },
21     });
22     return { success: true };
23   } else {
24     const team_info = await prisma.team.findUnique({
25       where: {
26         namespace_name: namespace,
27       },
28     });
29     await prisma.log.create({
30       data: {
```

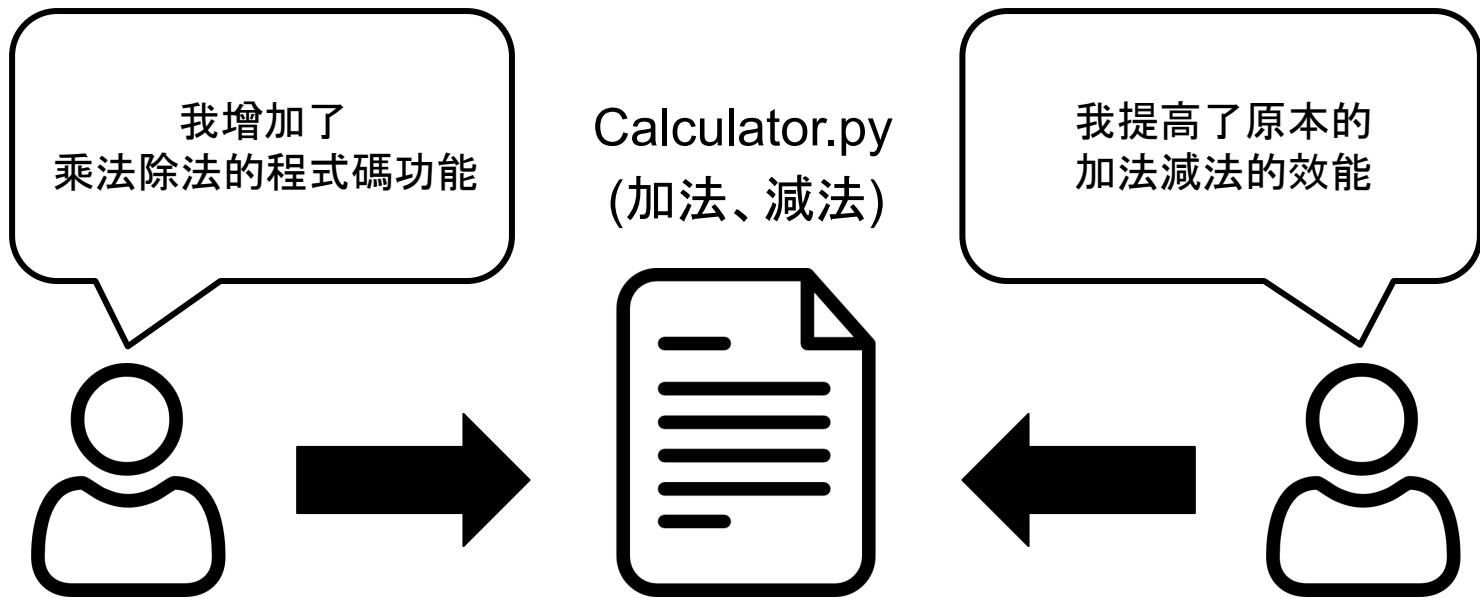
Ln 13, Col 1 Tab Size: 4 UTF-8 LF {} TypeScript

多人共同開發遇到問題

- 多人同時修改想同一檔案
- 合併開發成果而造成程式碼衝突 (Merge Conflict)
- 專案爛掉了, 想要回復 (Revert) 到先前穩定版本



共同開發 (想像一下..... 同時上傳工作成果)



版本控制 - 你是否想找之前的程式碼

Calculator.py (v1)

一週前版本



加法

更新



Calculator.py (v2)

三天前版本

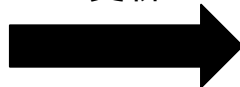


加法、乘法

回復



更新



Calculator.py (v3)

現在版本



(x) 程式改到壞掉了

Git 與版本控制的基本觀念

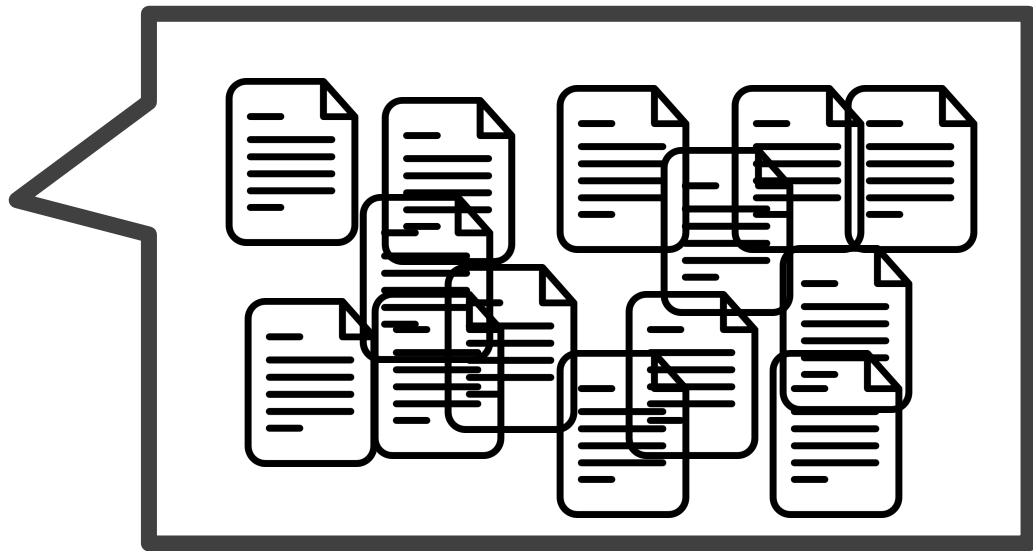
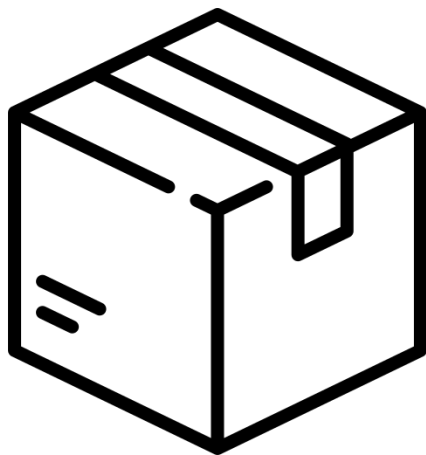
在 Git 裡常提到的一些名詞

- Repository (儲存庫): 等同於你的專案資料夾。
- Commit (提交): 專案的變更記錄。
- Branch (分支): 專案中的一個可分叉或合併的獨立開發線。

Repository 示意圖

好多的文件、檔案、程式碼

Repository



Commit 示意圖

Caculator.py

三天前版本



加法

Commit



新增乘法功能

Caculator.py

現在版本



加法、乘法

Branch 示意圖

Calculator.py

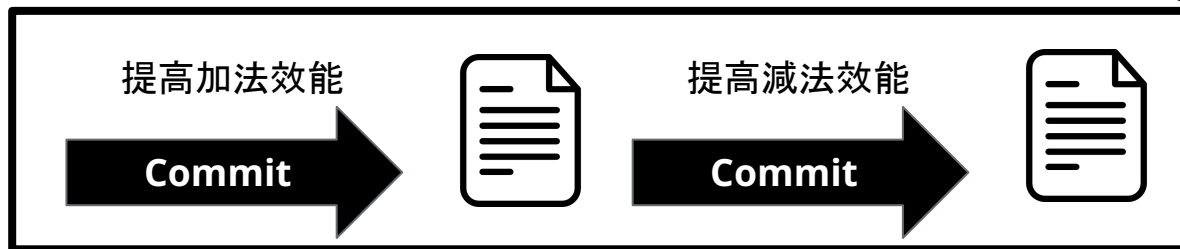


加法、減法

Branch 1



Branch 2



Merge?
合併?

多人共同開發 - 你必須要有的習慣

- 除了初始化 Repository 內容之外，切勿在主要分支 (main branch) 做事
- 當你正要做一個新功能、改動時，請拉一個 branch
- 隨時更新 Repository 內容 (git pull)
- 讓其他人更懂你寫的內容
 - 像是:commit 訊息、branch 名稱、程式碼命名與註解、完整的 README
- 切勿上傳敏感、無意義或龐大資料 (.gitignore)
 - 像是:環境變數檔案 .env、package、.DS_Store

用 Git 指令上傳你的成果吧

實作：在 GitHub 的第一個 Repository

- 前往 GitHub 創建 Repository
- 在電腦上新增資料夾 (好的習慣: 最好將資料夾命名跟 Repository 名稱一樣)
- 開啟終端機到你的資料夾下 (本機、VSCode 都可以), 執行網頁上指令



Start coding with Codespaces

Add a README file and start coding in a secure, configurable, and dedicated development environment.

Create a codespace





Add collaborators to this repository

Search for people using their GitHub username or email address.

Invite collaborators


Quick setup — if you've done this kind of thing before

 Set up in Desktop or [HTTPS](#) [SSH](#) <https://github.com/ryanlinlui/your-repository-name.git> 

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# your-repository-name" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/ryanlinlui/your-repository-name.git
git push -u origin main
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/ryanlinlui/your-repository-name.git
git branch -M main
git push -u origin main
```



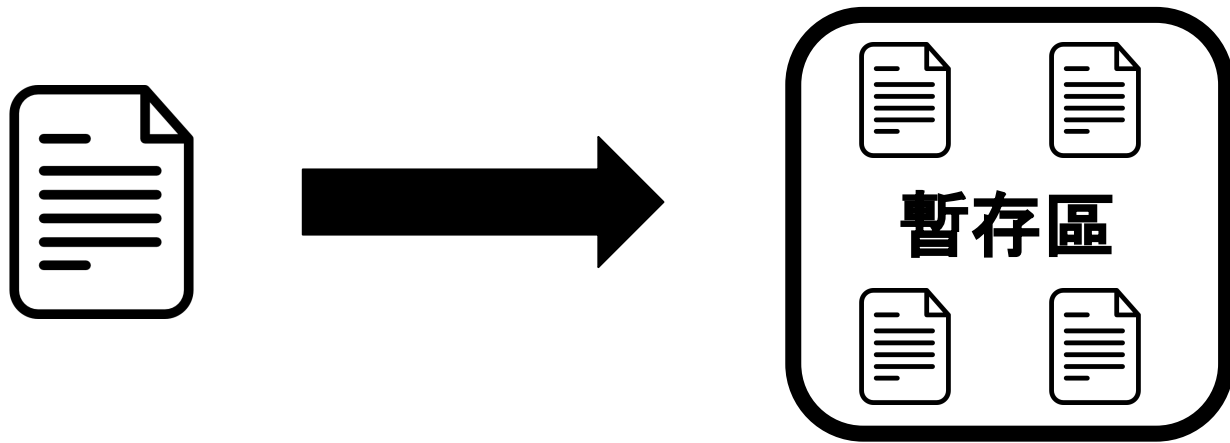


Now It's your Turn !!

```
function filterStudies({ studies, filterByOrg = false, filterByTopic = false }) {  
  return studies.filter(study => {  
    if (filterByOrg) {  
      return study.organizat
```

git add <your-files>

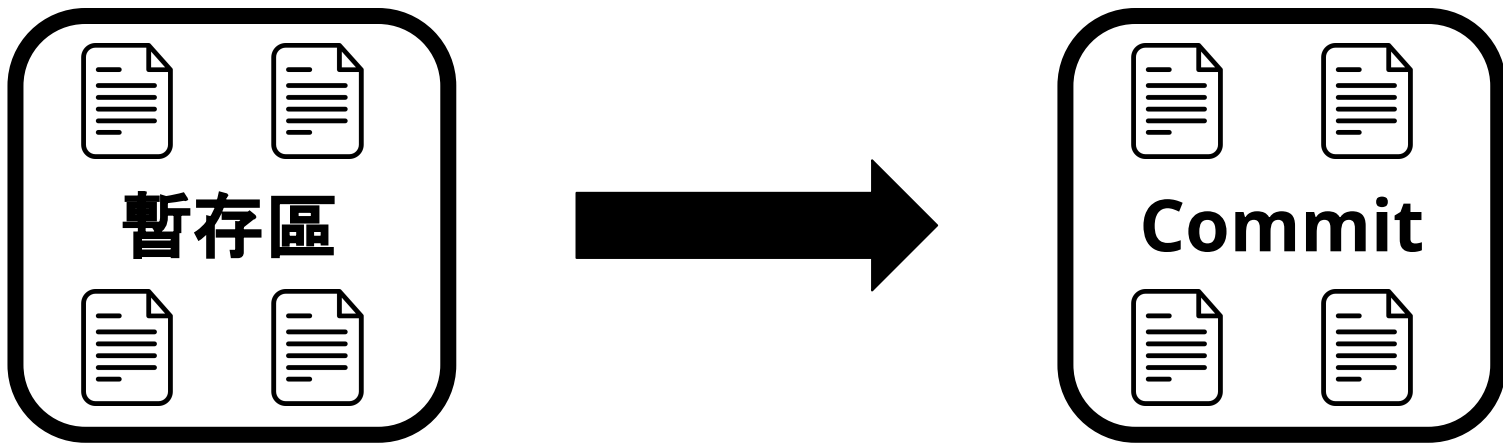
檔案的變更紀錄新增至暫存區, commit 之前必須做的事情



新增所有檔案: git add .

git commit -m “message” <your-files>

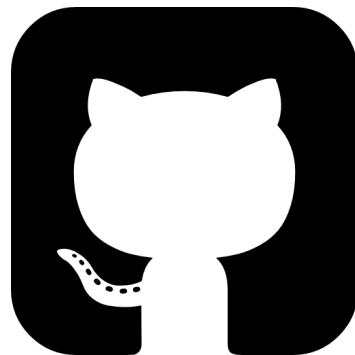
挑選暫存區檔案的變更記錄轉換成 commit



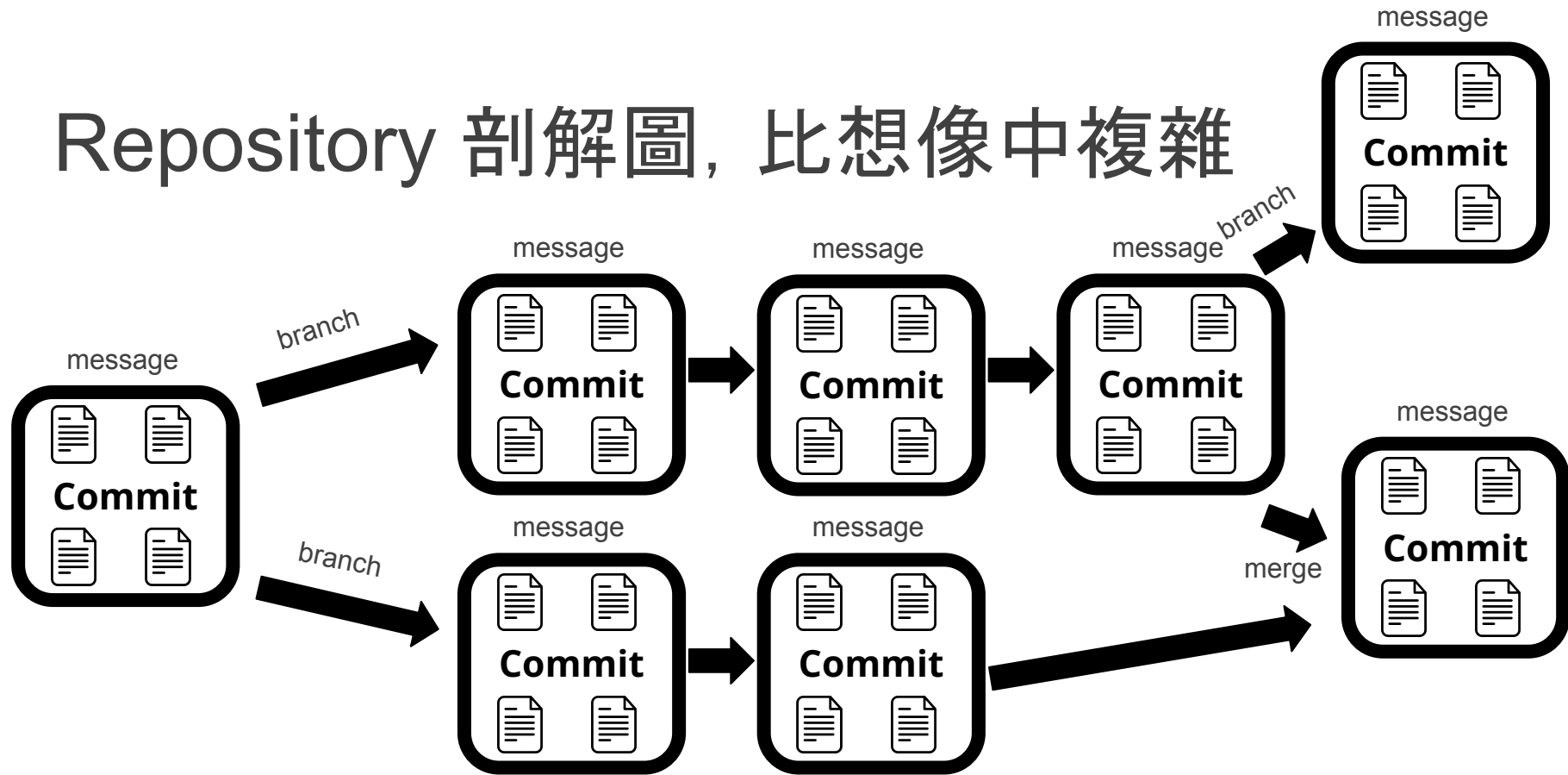
git push

本地 branch 裡的 commit 紀錄更新上傳到 GitHub

message



Repository 剖解圖，比想像中複雜



實作：在 Repository 打打 Git 指令

- 請利用 Git 指令，嘗試成功上傳一個檔案！！！！！！！！！！！！！！！！！！！！
- 你可能還會需要用到：
 - `git config --global user.name "your-name"`
 - `git config --global user.email your@email.com`



Now It's your Turn !!

中場休息

10 分鐘

GitHub: 全球工程師開源平台

GitHub、開源？

- GitHub 讓大家開放公開程式碼與作品給其他人觀看
- GitHub 也可以是社群，俗成工程師交友平台
- 以 GitHub 遠端伺服器為基礎，方便團隊協作和程式碼管理
- 開源：允許自由修改與散布 (授權條款)，社群共同維護與優化專案



在 GitHub, 你可以.....

ryanlinjui / taiko-music-generator

Type / to search

<> Code

Issues 2

Pull requests

Discussions

Actions

Projects

Wiki

Security

Insights

Settings

taiko-music-generator

Public

Pin

Unwatch 1

Fork 0

Star 3

main

1 Branch

0 Tags

Go to file

Add file

<> Code

ryanlinjui chore: ignore gradio cert ✓

20cae53 · 3 weeks ago

29 Commits

| | | |
|-------------------|--------------------------------------|--------------|
| .github/workflows | chore: deploy huggingface space | last month |
| assets/sound | chore: add taiko soundtrack source | last month |
| examples | fix: compress ogg file | last month |
| taiko | fix: github source badge not showing | last month |
| .gitignore | chore: ignore gradio cert | 3 weeks ago |
| LICENSE | Create LICENSE | 7 months ago |
| README.md | docs: add huggingface space link | last month |
| app.py | fix: output error message on gradio | last month |
| pyproject.toml | chore: update gradio version to 5.xx | last month |
| requirements.txt | chore: add requirements.txt | last month |

README

MIT license

Taiko Music Generator

Mix sougs and [太鼓の達人 \(Taiko no Tatsujin\)](#) drum sounds to generate Taiko Music.

Go to website [here](#).

About

Input your .tja file to convert Taiko Music.

[huggingface.co/spaces/ryanlinjui/taik...](#)

music

songs

tja

taiko-no-tatsujin

Readme

MIT license

Activity

3 stars

1 watching

0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Languages

Python 100.0%

上傳你的作品
給大家看

秀出你的個人頁面



RyanLin
ryanlinjui

Edit profile

19 followers · 21 following

@NTNU-CSIE

Taipei, Taiwan

15:16 (UTC +08:00)

ryanlinjui@gmail.com

Achievements



Highlights

PRO

Pinned

Customize your pins



course-learning Public

List of NTNU, NTU, NTUST courses that I learned.

C 10



darkchess-robot Public

A self-learning robotic arm plays Darkchess with you in Real-World.

C++ 3 1



fahsuanlin/BCGUnet Public

BCGUnet: suppressing ballistocardiography (BCG) artifact from EEG-MRI data

Python 3 4



hinagiku-dev/Hinagiku Public

Hinagiku (雛菊), or Daisy in English, is an intelligent system designed to support discussions in educational environments.

Svelte 1

484 contributions in the last year

Contribution settings

2025



Learn how we count contributions

Less More



@hinagiku-dev



@CodeAIdapter



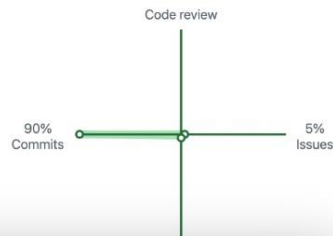
@NTNUHackathon

More

Activity overview



Contributed to [ryanlinjui/course-learning](#), [ryanlinjui/cpGrader](#), [hinagiku-dev/Hinagiku](#) and 23 other repositories



查看最新、朋友動態

Home

Filter 5



skyhong2002 followed

3 days ago



Wama wama-tw

對於技術充滿熱情。期許自己能為很多人使用的網站或功能貢獻。

55 repositories 47 followers

Follow



malteborgmann starred a repository

4 days ago



sonnylazuardi/cursor-talk-to-figma-mcp

Cursor Talk To Figma MCP

TypeScript 1.1k

Star

Trending repositories · [See more](#)



ai-dynamo/dynamo

A Datacenter Scale Distributed Inference Serving Framework

Rust 2.7k

Star



langmanus/langmanus

A community-driven AI automation framework that builds upon the incredible work of the open source community. Our goal is to combine language models with specialized tools for tasks like web search...

Python 2.5k

Star

Popular projects among people you follow



barry-ran/QtScrcpy

Android real-time display control software

C++ 23.3k

Star



JacobLinCool starred a repository

5 days ago



modelcontextprotocol/servers

Model Context Protocol Servers

JavaScript 22.2k

Star



JacobLinCool created a repository

5 days ago



k033-research/firstrade-rs

Unofficial Firstrade Client in Rust.

Rust 1

Star



April-Aries starred a repository

last week



albertoperdomo2/astro-vim

Astro landing page simulating a Vim console, where key bindings work!

Astro 31

Star



giantche starred your repository

last week



ryanlinjui/darkchess-robot

A self-learning robotic arm plays Darkchess with you in Real-World.

C++ 3

Star



NaoCoding starred a repository

last week

讚 + 收藏 (Star)

suragnair / alpha-zero-general

Type / to search

<> Code Issues 44 Pull requests 26 Discussions Actions Projects Security Insights

alpha-zero-general Public

Watch 113 Forks 1.1k Stars 4k

master 9 Branches 0 Tags

Go to file Add file <> Code

About

A clean implementation based on AlphaZero for any game in any framework + tutorial + Othello/Gobang/TicTacToe/Connect4 and more

Jason N. White Update LICENSE, fix license year (#327) f1a78e0 · 2 months ago 221 Commits

| | | |
|----------|---|-------------|
| connect4 | removed tf1 and chainer implementations | 3 years ago |
| docker | Installation process was broken (#253) | 4 years ago |

討論區 (Issue, Discussion)

is:issue state:open

Open 46 Closed 137 Author Labels Projects Milestones Assignees Types Newest

- Can not run the code - nvidia-docker issue
#331 - fatmazgucc opened last week
- Question about MCTS tree retention strategy and potential improvement
#330 - Karesis opened 2 weeks ago
- visit count initialization for the leaf node
#326 - eljoac opened on Nov 20, 2024
- Shall we use 2 channels for the board representation?
#323 - zjusbo opened on Nov 10, 2024
- About Arena
#321 - huskyth opened on Aug 8, 2024
- iteration times
#320 - su322 opened on Aug 5, 2024
- Nit: Chess requires more than the state of the board.
#314 - BasedLukas opened on Jan 26, 2024
- mini-othello not trained right?
#312 - elephantpanda opened on Oct 7, 2023
- google colab err
#304 - jinpengchina opened on May 2, 2023
- Using while loop in the MCTS search function instead of recursion
#303 - apple1113 opened on Apr 16, 2023

 Google Developer Student Clubs

Issues 46 Pull requests 26 Discussions Actions Projects Security Insights

ZeroDivisionError: float division by zero #288

Open



visualization opened on Jan 6, 2023 · edited by visualization

Edits · ···

Hey there,

First of all thank you for sharing this repo with the public. Really cool project!

Regarding the opened issue: I experience currently a problem with MCTS in `def getActionProb`, namely the error `ZeroDivisionError: float division by zero`. It originates from `probs = [x / counts_sum for x in counts]` and the problem is that all the elements in `counts` are 0 because the state/action pair has not been discovered and saved in `self.Nsa` yet (`counts = [self.Nsa[(s, a)] if (s, a) in self.Nsa else 0 for a in range(self.game.getActionSize())]`).

Any ideas what might be causing the issue & how to fix it? I see in the other projects (e.g Othello) no such problems, so I am wondering what might be culprit here. I am currently trying to make the game of hex work in the project.

Cheers



goshawk22 on Jan 6, 2023

Contributor · ···

[#191](#) might help



visualization on Jan 6, 2023

Author · ···

Yes, thanks for pointing this out. However I already discovered this one and I couldn't find anything similar in my code. I made sure that I always copy the pieces in the game code. It just seems that mcts is not exploring edges which the players in the arena then want to play. But I guess it should only play discovered moves (s/a) in the arena since it uses greedy search, right?



Assignees
No one assigned


Labels
No labels

Projects
No projects

Milestone
No milestone

Relationships
None yet

Development

 Open in Workspace

No branches or pull requests

Notifications [Customize](#)

 Subscribe

You're not receiving notifications from this thread.

Participants



suragnair / alpha-zero-general

Q Type to search

<> Code Issues 44 Pull requests 26 Discussions Actions Projects Security Insights

alpha-zero-general Public

Watch 13

Fork 1.1k

Starred 4k

suragnair / alpha-zero-general

Q Type to search

Code

<> Code Issues 46 Pull requests 26 Discussions Actions Projects Security Insights

Add Othello player that can play with external programs using GTP #316

Merged suragnair merged commit into suragnair:master from plamentotev:gtp on May 8, 2024

Conversation 2 Commits 1 Checks 0 Files changed 4

plamentotev commented on May 7, 2024

GTP (Go Text Protocol) is a text protocol that allows Go (the game, not the programming language) programs to play with each other. Due to the similarity in the board representation the protocol is used by Othello/Reversi programs.

Add Othello player that uses the protocol to play with external programs. That makes it easy to test how well the AI performs against other AI bots.

Add some additional methods to the Player interface - startGame, endGame and notify. Those methods are essential to keep the state synced with external program.

Update Arena to use those methods if present to keep it backward compatible and not to force other implementations to implement class. To this end GTPothelloPlayer is callable so that players can still be a simple function.

Using the player in pit.py is pretty straightforward:

player2 = GTPothelloPlayer(g, ["/path/to/bin/Egaroucid_for_Console", "-gtp"])

I've tested it with Egaroucid and Edax

Contributor

Reviewers
No reviews

Assignees
No one assigned

Labels
None yet

Projects
None yet

Milestone
No milestone

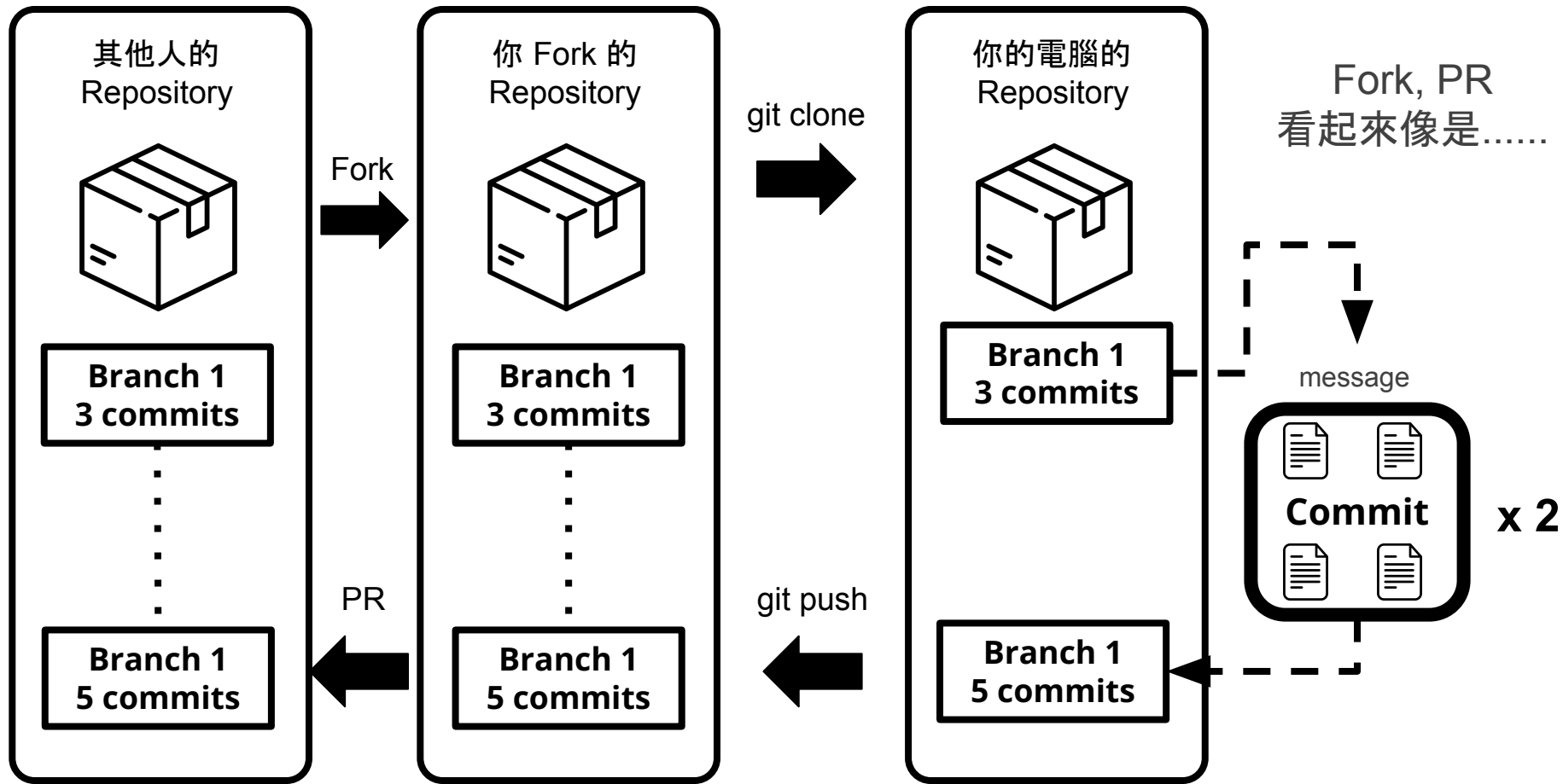
Development
Successfully merged

貢獻你的成果
(Fork, Pull Request)

用 GitHub 貢獻你的成果吧

GitHub 常用: Fork, Pull Request, Issue

- Fork: 在 GitHub 平台上, 將別人的 Repository 複製到自己的帳號底下
- Pull Request (PR): 本質上是一種 branch merge (合併) 的行為請求:
 - 可以在自己的 Repository 的 Branch 之間
 - 可以跨出跟別人 Repository 的 Branch 做合併
- Issue: 貼文討論區, 可討論任何內容, 像是:
 - 回報 Bug、問問題、功能請求、聊天吵架



實作：貢獻你的成果

- Fork 專案到你的電腦內 [[Link](#)], 並且 clone 下來
- 在專案裡的 “./NTNU-GDG/git-tutorial-1/assignments” 資料夾底下新增檔案：
 - 檔案命名為你的學號, 副檔名為 txt
 - 內容隨意 (寫心得、抱怨我)
- 開啟終端機到你的資料夾下 (本機、VSCode 都可以)
- 利用 git 指令上傳你的檔案
- 發 Pull Request 到 ryanlinjui/teaching-archives



Now It's your Turn !!

```
function filterStudies({ studies, filterByOrg = false, filterByTopic = false }) {  
  return studies.filter(study => {  
    if (filterByOrg) {  
      return study.organizat
```


結語

- 今天講了都是最最最基礎的指令部分
- 還有很多要學習的內容, 像是:
 - Branch 管理、Merge Conflict 合併衝突
 - Reset: 回復之前版本
 - 共同開發實務, 邀請別人一起加入你的團隊
- 近請期待下次的 Git 課程系列: Git 指令與觀念(二)

git add

檔案新增至暫存區

Adds files to the staging area.

git commit

暫存區的變更記錄到 repo

Records changes from the staging area in repo.

git status

顯示目錄和暫存區狀態

Displays state of the directory & staging area.

git push

本地 repo 內容上傳到 GitHub

Uploads local repo content to GitHub.

git pull

從 GitHub 提取變更到本地 repo

Fetches changes from GitHub to local repo.

git branch

列出、建立或刪除 branch

Lists, creates, or deletes branches.

git checkout

切換到另一個 branch

Switches to a different branch or restores files.

git init

初始化一個新的 Git repo

Initializes a new Git repo.

git remote

管理與 GitHub 的連線網址

Manages connections to GitHub repo URL.

git config

設定 Git 使用者特定配置

Configures user-specific settings for Git.

git log

查看 repo 的 commit 歷史

Shows the commit history of the repo.

git restore

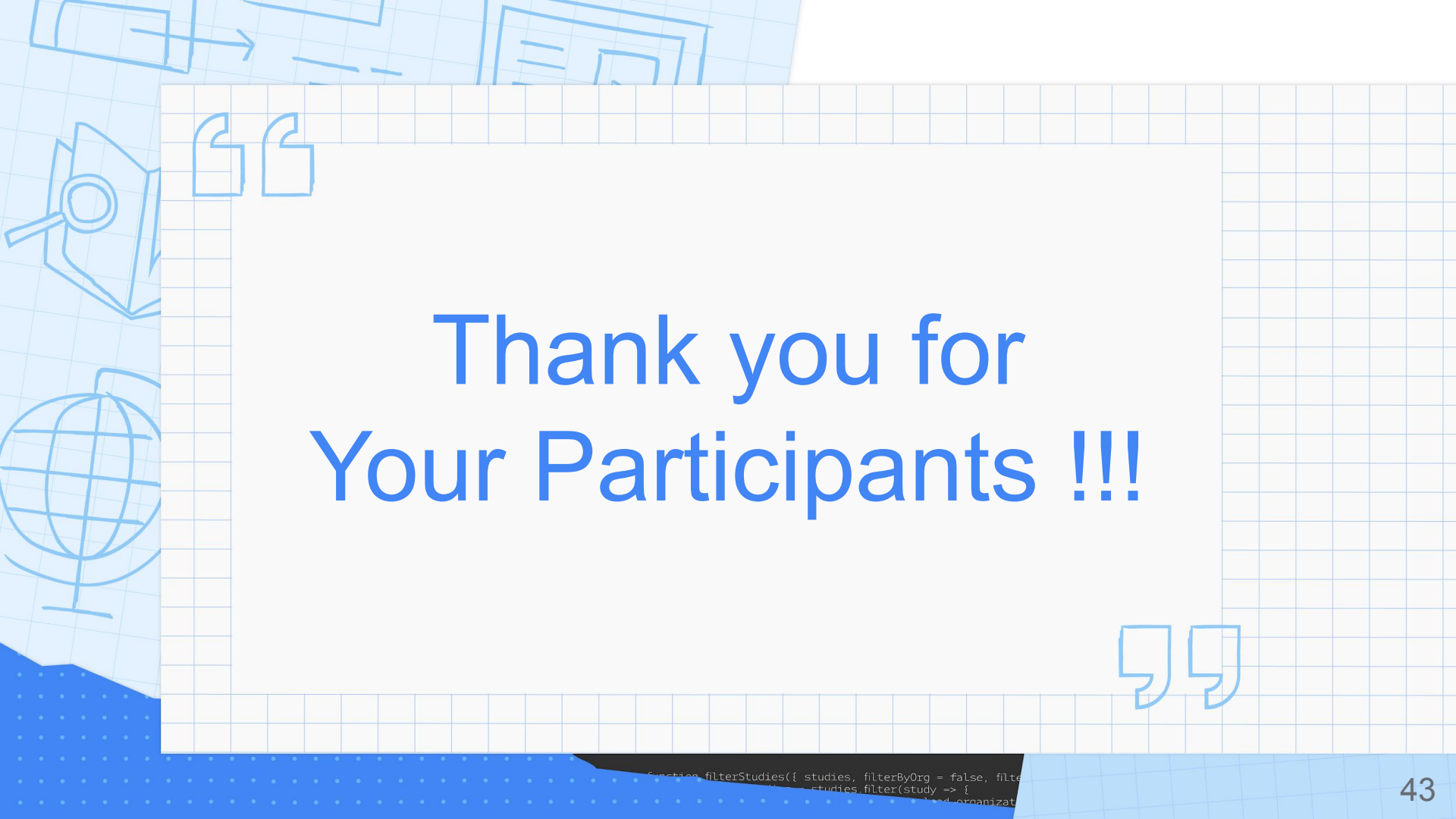
還原目錄檔案到最近 commit

Restores working tree files to their last committed state.

git reset

將當前分支移到某 commit

Moves the current branch to a different commit.



“

Thank you for
Your Participants !!!

”

```
function filterStudies({ studies, filterByOrg = false, filterByStudy => {  
  studies.filter(study => {  
    if (filterByOrg) {  
      return study.organizat
```