

專題名稱：[簡易五子棋]

作者：[李子濤] - [11303013A]、[陳柏均] - [11303008A]、[黃士偉] - [11303091A]、
[許有維] - [11303103A]

專題簡介

這次的專題利用 Python 的 socket 模組，設計並開發了一個基於 TCP 協定的五子棋。系統採用了 Server-Client 架構，支持兩名玩家通過區域網路進行即時對局。為了提升遊戲體驗，我們使用 tkinter 建立一個簡易的棋盤，並透過多執行緒技術確保通訊與介面操作不互相互干擾。

功能特色

- ✔ 支援即時連線對戰：兩名玩家可透過 TCP 協定實現即時的同步下棋操作。
- ✔ 顏色選擇機制：玩家可自由選擇黑棋（先手）或白棋，系統會自動把剩餘的另一個顏色給另一位玩家。
- ✔ 輸贏判定：伺服器端設有完善的五子連線邏輯（橫、豎、斜方向），能精確判定勝負並廣播結果。
- ✔ 斷線與自動重置：若玩家中途斷線，伺服器會偵測到異常並自動重置遊戲狀態，保障系統持續運行。

系統架構

系統由兩個主要程式組成：

- server：作為中央處理節點，負責監聽連線、管理玩家顏色分配、驗證下棋位子有沒有重複或出界、判定勝負及狀態廣播。
- client：玩家拿的程式，負責顯示棋盤、接收玩家滑鼠輸入並將指令傳送至伺服器，同時同步顯示對手的下棋狀態。

協定設計

訊息格式

本系統使用 純文字字串 搭配 逗號分隔 的格式進行通訊，每條訊息以換行符號 \n 作為結尾。

訊息開頭	參數	說明
COLOR	black/white	玩家選擇棋子顏色。
START	black/white	伺服器通知遊戲開始，並確認該玩家的棋子顏色。

訊息開頭	參數	說明
MOVE	x, y	下棋位置的座標（棋盤範圍 0-14）。
WIN	1/2	通知勝負結果（1 代表黑棋勝，2 代表白棋勝）。
RESET	無	請求或通知遊戲狀態重置（清空棋盤）。

範例

- 客戶端傳送：MOVE, 7, 7\n（玩家在棋盤中心座標（7, 7）落子）。
- 伺服器廣播：MOVE, 7, 7\n（通知另一位玩家在相同位置顯示棋子）。

安裝與執行

需求

- Python 版本：3.7+
- 依賴標準庫：socket, threading, tkinter

執行步驟

1. 啟動伺服器：在伺服器端執行 python server.py，預設監聽 Port 8000。
2. 配置客戶端 IP：開啟 client.py，將 self.sock.connect(("拿 server 的人的 IP", 8000)) 修改為伺服器實際的區域網路 IP 地址。
3. 啟動客戶端：兩名玩家分別執行 python client.py
4. 開始對弈：玩家透過介面選擇黑子白子，待雙方準備就緒，系統會彈出訊息並開始遊戲。

未來改進方向

- [] 增加計時器功能：增加倒數計時，超過時間可以直接獲勝 或 超時方會隨機下在棋盤上。
- [] 實作悔棋功能：在雙方同意下可以悔棋，一局一次:D。
- [] 內建聊天室：舉例：噴垃圾話、場外指導，協調顏色，避免先連上先選色的問題。
- [] 介面美化：使用線上圖片（如木紋棋盤、玉石棋子）取代現有的 Canvas 繪圖
- [] 可連線數量額滿：server 可以拒絕連線
- [] 作弊預防

參考資料

1. *Foundations of Python Network Programming*
2. Python 官方 Socket 模組文件 (Official Python Socket Documentation)
3. 課堂作業
4. AI 輔助

分工

李子濤	黃士瑋	陳柏均	許有維
上台報告、程式	程式	Readme 文案	程式架構設計

<https://github.com/DE02024/TCP-IP-Project-Presentation-Group-2>