

教育部教學實踐研究計畫成果報告
Project Report for MOE Teaching Practice Research Program

計畫編號/Project Number：PGE1122003

學門專案分類/Division：通識(含體育)

計畫年度：☒112 年度一年期 ☐111 年度多年期

執行期間/Funding Period：2023.08.01 – 2024.07.31

培養非資訊領域學生運用程式能力初探社會不平等議題
基礎程式設計

計畫主持人(Principal Investigator)：鄭年亨

協同主持人(Co-Principal Investigator)：無

執行機構及系所(Institution/Department/Program)：臺北醫學大學通識教育中心

成果報告公開日期：☐立即公開 ☒延後公開

繳交報告日期(Report Submission Date)：2024 年 09 月 20 日

培養非資訊領域學生運用程式能力初探社會不平等議題

一、本文 (Content)

當前運算思維已成為 108 學年度十二年國民基本教育科技領域的核心能力，而高等教育也藉由「推動大學程式設計教學計畫」銜接 108 課綱（教育部，2014），以培育非資訊科技專業學生的基礎程式設計能力。在最新的「教育部提升大學通識教育中程計畫」中，「資訊科技素養培育」被列為重點之一，特別強調「多元運算思維與量化分析素養培育」以及「軟體創作統合與運算表達理解培育」的子項目，充分反映出程式設計能力的重要性。這些項目期望學生能夠透過跨領域的創新結合，運用程式設計來探索議題並解決專業領域中的重要問題，最終對未來社會做出有意義的貢獻。因此，程式設計能力已逐漸成為現代大學教育中的一項通識能力。

隨著我們邁入大數據時代，未來的研究人員或從業者將面臨大量的研究資料。若能運程式技能來處理和分析數據，不僅能有效率地完成任務，還能從中揭露隱藏的資訊。即使學生未來不從事與程式設計直接相關的工作，他們也可能會與資訊專家合作，共同開發新的數位產品。在這種情境下，具備跨領域合作的能力，能夠與專家攜手解決複雜問題或探索現象，將成為未來學生不可或缺的技能。例如，未來的醫學生或許能運用程式技術，從龐大的病歷資料中找出隱性規律，或開發自動偵測病灶的軟體，協助醫生進行更精準的診斷與治療。這些能力顯示出程式設計不僅是一項技術，還是與科技對話的橋樑，未來它將如同聽、說、讀、寫一般，成為數位時代公民的基本資訊素養。

臺北醫學大學根據本校通識教育十年白皮書中的兩大發展主軸——「完整新世代人才培育」及「智慧科技導入學習與管理」，自 106 學年起將「基礎程式設計」納入通識核心必修課程，主要針對大一新生，但也鼓勵高年級學生選修。在教學團隊的努力下，修習「基礎程式設計」課程的學生比例已超過 99%。學生在修畢此課程後，可以繼續選修跨領域學院的程式設計微學程，並於四年級時修習專業系所的智慧生醫特色課程。由此可見，學校對程式設計能力的高度重視，並期望學生能在通識課程中奠定扎實的基礎，激發對程式設計的學習興趣。

1. 研究動機與目的 Research Motive and Purpose

1.1 研究動機

筆者自 109-2 學期加入教學團隊，擔任本校通識必修課「基礎程式設計」主授老師。在過去幾年的教學中，筆者發現，對於本校醫護領域的學生來說，理解程式語法並不是他們學習的主要困難，真正的挑戰在於如何運用這些語法來解決實際問題。整體而言，這些困難主要來自以下兩個方面的原因：學習目標欠缺意義，以及教學方式字典化。

首先，學習目標欠缺意義是指課程期待學生最後的作品對於學生的意義不足。對於醫護領域的學生來說，程式設計並非他們未來職場的核心技能，因此學習興趣普遍較低。儘管課程在第一週已向學生說明未來工作場域的變遷，以及學校對程式素養的期望，但大多數學生仍然對程式設計缺乏興趣。由於領域之間的差異，學生很難在課程初期預見期末能夠達成的成果。以

往學生需經歷漫長的基礎課程，直到期末才開始思考實際應用。然而，如果能在課程開始時設立有意義的學習目標，學生便更有動力投入學習。因此，作為一門通識課程，我們需要為學生設置適切的學習目標，讓他們了解程式設計同樣能夠為社會帶來正面的影響與貢獻。

其次，**教學方式字典化**是指教學形式流於語法講解。典型的程式語法教學通常包括介紹語法、示範程式碼、佈置類似習題、示範解決過程。實際上，在程式教學中，後兩個步驟——程式練習與解題——更為重要。然而，由於程式語法往往包含許多規則和細節，教師經常會花過多時間在一開始的語法介紹上，這種字典化的教學方式可能會讓學生失去耐心，降低學習動機。申請人雖然了解字典化教學的缺點，但作為新進教師，常常不自覺地多講解一些語法，結果反而壓縮了學生實際練習的時間。學習程式語言如同學習外語，不是僅靠增加單字或文法的講解就能進步，更重要的是提供足夠的練習機會，使學生能內化所學。因此，教學方式需要在語法講解與實際練習之間取得平衡，教師應在簡明扼要地介紹基本概念後，留出足夠的時間讓學生實際應用，從而提升學習效果。

1.2 研究目的

為了讓學生更好地理解程式語言的功能與意義，並迎接大數據時代的挑戰，本計畫將重構「基礎程式設計」通識課程，將傳統的教學方式轉型為以「社會不平等」為主題的探索型學習活動。「社會不平等」屬於聯合國永續發展目標（Sustainable Development Goals, SDGs）中的第十項目標「消弭國內與國家間的不平等」。社會中總是存在著資源與階級的不均，而這種現象往往難以從個人角度看清，除非處於弱勢群體。新課程旨在培養學生善用公開資料，透過程式技術呈現不同族群之間的差異，進而理解弱勢群體的困境。選擇「社會不平等」作為資料探索主題有三個好處：首先，社會不平等是一個真實的社會議題，能夠接露生活中不易察覺的現象，進而引發學習動機；其次，社會不平等議題涉及大量數據，提供一個自然的環境讓學生運程式解決問題；第三，學生上完本課後能對該議題有初步認識，並將此經驗運用於其他相關通識課程。

具體而言，本研究旨在重新設計「基礎程式設計」課程，以「資料視覺化」程式設計教育為主軸，藉由引入「社會不平等」的議題，來引導學生學習程式設計，以提升學生的學習動機與未來選修相關課程的意願。課程設計強調讓學生學習如何收集與整理數據，並根據事實進行討論與發想，有助於他們在未來應對其他議題導向的課程。此外，研究將深入探討新版課程對學生學習成效的影響，為後續課程設計和教學改進提供參考依據。

2. 研究問題 (Research Question)

本計畫在基礎程式設計通識必修課程中，推動以「社會不平等」議題為主題的「資料視覺化」程式教育，旨在幫助非資訊領域的學生建立學習程式的意義感，並促進他們與其他通識課程的連結。第一個研究問題是：社會不平等議題的資料視覺化程式教學，是否能夠有效促進非資訊領域學生的基礎程式語法學習？本計畫的假設是，透過這種教學方式，學生能夠在解決真實社會問題的背景自然學習程式語法，從而增加學習的應用性和動機，同時不會影響他們學習程式語法的時間與進度。

本計畫第二研究問題為：社會不平等議題的資料視覺化程式教學，是否有助於提升非資訊領域學生的學習動機？本計畫假設，社會不平等議題能夠為學生提供具體且有意義的學習目標，從而提升他們的學習動機。與單純學習程式語法不同，將程式設計應用於解決真實的社會問題，讓學生能夠看到學習成果的實際應用，這有助於增強他們對學習內容的認同感和參與感。透過資料視覺化，學生不僅能夠呈現和進一步了解社會不平等現象，並在這個過程中發現程式設計的社會價值，這種實際應用的情境可能提升學習的內在驅動力，進而深入探究社會議題。

本計畫的第三個研究問題為：在實體與遠距兩種教學形式下，非資訊領域的學生在前述程式教學模式中，哪一種形式的學習表現較好？根據過去的經驗與相關文獻，遠距學習形式通常被認為會減弱學習效果，這主要是由於遠距學習缺乏足夠的社會存在感（social presence）與社群感（sense of community）（Kruger-Ross & Waters, 2013）。這些因素可能會使學生在遠距環境中感到孤立，進而影響他們的學習動機和參與度。因此，本研究將探討這兩種教學形式對非資訊領域學生在學習程式語法和社會議題探索上的具體影響，並比較其學習成效，以期為未來的教學設計提供實證支持。

3. 文獻探討 (Literature Review)

近年的文獻綜述（Medeiros, et al., 2018; Popat, & Starkey, 2019）整理出學習程式的益處，除了培養運算思維本身，程式學習還能夠培養數學問題解決、團隊合作、自主學習與批判思維。學生程式學習的挑戰，主要包括問題解決、學習動機與投入，前者屬於認知層次、後者屬於情意方面的問題，而這也與本課程所遭遇到的挑戰相同。面對非資訊領域學習者學習程式語言，鷹架教學法能夠減低認知負荷。但課程不應只減輕難度，忽略學習目標應該保有的意義性。

鷹架教學法原本是指老師為協助新手學生學習新知識或技能，而給予的適性支援，直到學生學會（van Oers, 2014）。老師常用的鷹架支援策略包括提問、提示、解釋、示範等（van de Pol, 2012）。鷹架應用於問題解決時也時常涉及步驟化教學，藉由將問題解決的步驟細緻化，再鷹架大部分的步驟，讓學生練習特定步驟。在程式教學上，鷹架教學法主要用於運算思維，包括問題分解、規律識別、抽象化、演算法思維與除錯等步驟（Angeli, 2022; Zhang, et al., 2021）。一方面而言，新手學生在尚未熟悉全部步驟時，就可以先藉由鷹架體驗全部流程，能夠提高滿足感；另一方面，鷹架教學法也能減少學生的認知負荷，讓學生將注意力集中在老師希望學生了解的地方。因此，鷹架教學法廣泛受老師使用。

鷹架教學法也需要注意鷹架褪除的時機，防止過度鷹架（over-scaffolding）的問題。由於鷹架的目的是幫助學生過渡到獨立解題，因此理想的鷹架最終不應存在。不過在實務上，在許多教學現場，鷹架常常沒有褪除或褪除過慢，導致過度鷹架的現象，造成學生覺得步驟太繁瑣，無法自由發揮自己的能力（Bierer, 2018）。而在程式教育的其中一個教學挑戰就是缺少適當的教學方法與工具，另外也由於學生人數多，老師實際上很難進行個人化教學（Popat, & Starkey, 2019）。雖然學者們正在研究適性化的電腦自動鷹架方法（如 Rau, et al., 2021），但實務上仍有許多挑戰正在克服中。除此之外，也有學者提倡反向鷹架，意即一開始不給鷹架，而讓學生在遇到問題時自己選擇鷹架的方法（Cheng, et al., 2015）。這種方法解放了學生的解題歷程，讓學

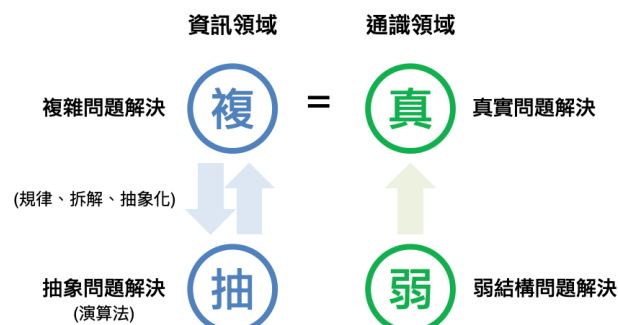
生有更多自由度發揮自己的初始想法，不過可能會有衍伸的濫用鷹架問題。

本計畫受到「推動大學程式教學計畫」國立臺灣大學資訊管理學系孔令傑教授示範課程的啟發(<https://www.openedu.tw/course?id=1274>)。該課程為了讓學生運用迴圈與條件敘述的概念，創造了寄信給大量顧客的情境；為此，該課程在範例中增加了自動寄信的功能，這個功能不是程式初學者該學的內容，因此老師將此技術準備好，直接讓學生使用。從鷹架的角度來說，該課程協助學生完成更困難的事情。然而，這跟典型的鷹架意義不太一樣，原本的鷹架是指學生無法完成的步驟，先由鷹架協助完成，讓學生經歷整個過程，等到學生有能力後，再拆除鷹架讓學生獨自完成；而這裡希望藉由鷹架讓學生能夠做到原本的課程無法教到的技術，讓學生擁有成就感，並理解實際的應用場景。此類鷹架屬於鷹架教學法(van de Pol, Volman, & Beishuizen, 2010)中的「降低自由度」(reducing the degree of freedom)與「引發參與」(recruitment)。前者是認知鷹架，幫助學生解決進階的技術問題，讓學生完成少數步驟就可能完成有意義的任務；後者屬於情意鷹架，讓學生對學習任務產生興趣，進而主動參與任務。本計畫採用此概念來重新設計每個單元，讓學生能夠在基礎課程中完成諸如讀取資料集或繪製象形圖等有意義的任務。

4. 教學設計與規劃 (Teaching Planning)

4.1 教學目標

資訊領域的程式教育著重運算思維能力的培養，期許學生面對複雜問題時，能夠運用規律、拆解、抽象化等技巧，將問題簡化為抽象問題來攻克。在 108 課綱的資訊素養培育方針指導下，高中生的運算思維能力逐年增加，大學端的程式教育勢必轉型。本計畫提出適合通識教育的程式教學方式(如圖一所示)，即更強調「真實問題解決」，讓學生嘗試運用程式探索或解決實際問題，例如永續發展目標所涉及的重要問題。而在培養學生的過程中，也應更強調「弱結構問題」，減少標準答案的程式問題，讓學生學習自訂問題與解決問題。



圖一、通識領域的程式教育目標

本課程為基礎程式設計課程，以 Python 作為程式語言，並於期末專題介紹 pandas 與 matplotlib 兩個資料處理與資料視覺化的重要套件。學生將通過議題導向的小組專題，運用程式探索公開資料集，從中理解和探討社會現象。具體而言，本課程的三大教學目標為：

- 一、培養醫學領域學生掌握 Python 程式語言的基本語法、邏輯及應用能力；
- 二、提升醫學領域學生的運算思維能力，了解程式運作原理及資料處理方法；
- 三、引導醫學學生透過創新與合作方式完成專題，運用程式語言探究社會議題。

4.2 教學設計

本計畫的教學模式如圖二所示，學期中的各項程式作業以弱結構問題解決來設計，而期末專題以真實問題解決為最終目標。在語法教學後，期中作業將提供作業模板，老師在模板中會預先完成較困難的進階語法，讓學生只需要專注於當前語法的運用。除此之外，課程的每項作業皆無唯一解或標準答案，作業題目皆具有資訊不完整的特質，讓學生自行定義內容，發揮個人創意與自行追求想要達成的目標。在期末專題中，學生將分組定義感興趣的題目，自行尋找公開資料集，並以程式整理資料，最終呈現視覺化的程式圖表。老師在期末專題的介入更少，僅提供 pandas 與 matplotlib 兩個常見套件教學。



圖二、教學設計

4.3 教學案例

本學年課程規劃如表一所示。期中考前教授 Python 語法，從基本語法、算數運算子開始，逐步引入邏輯運算、迴圈、字典、模組等進階概念。課堂中的學習任務與作業皆以兩週為單位，第一週布置問題，第二週給予提示。布置作業問題時皆輔以模板，幫助同學體驗程式的功能。期末專題則將最後四週合併為一日工作坊，請同學週末分組完成期末專題。由於過去曾有學生反映遠距分組的效率不彰，因此本學年無論是實體或遠距班，工作坊皆採用實體形式進行。

表一、課程規劃

週次	週數	課程進度	學習任務與作業
1-2	2	基本語法、算數運算子	BMI 計算
3-4	2	邏輯與比較運算子、條件敘述	作業一：多結局互動故事
5-6	2	串列、元組、for 迴圈	作業二：多圖示象形圖
7-8	2	字典、while 迴圈	作業三：真實資料查詢
9	1	綜合應用	ChatGPT 輔助解題
10	1	期中考	n/a
11	1	模組與套件、資料集	作業四：繪製統計圖
12-15	4	期末專題	一日工作坊

以下針對其中三項學習任務與作業進行說明，分別用於闡釋教材設計、作業與專題之設計。

(1) 教材樣板案例：以「BMI 計算」學習任務為例

此學習任務於第 2 週之算術運算子單元課堂上實施。此時同學僅學過算術運算子，能夠實現的功能有限。但老師提供半成品範例作為教學鷹架，讓學生能完成較複雜的任務，並學習如何進一步修改和完善程式，使學習有方向性和實用性。除此之外，這個課堂活動也運用 Kaggle 競賽平台的 BMI 資料集，此資料集為國外真實資料集，收集了約 500 筆民眾的性別、身高、體重、體態的欄位資料。本活動使用此資料集，讓學生練習 BMI 的語法，其他諸如讀取資料集、迴圈等進階語法，皆由老師協助完成（圖三(a)），讓學提前感受到用程式統一處理資料集的好處。除此之外，此資料集中的體態欄位的是以美國國家衛生研究院所制定的標準，與台灣標準不一致。因此在下一個單元條件敘述，也可以繼續使用這個資料集，讓學生練習使用 if...elif...else 條件敘述語法，以台灣標準來分類此資料集中的體態。

```
ClassifyBMI.ipynb
檔案 編輯 檢視畫面 插入 執行階段 工具 說明 最近於 3月4日 編輯

+ 程式碼 + 文字

1 import pandas as pd #資料操作套件 (期末專題)
2
3 # 讀取csv資料集
4 data = pd.read_csv('bmi.csv')
5
6 # 使用迴圈讀取每一行
7 for index, row in data.iterrows():
8     g = row['Gender'] #g為性別
9     h = row['Height'] #h為身高
10    w = row['Weight'] #w為體重
11    i = row['Index'] #i為體態 (0=極度虛弱、1=虛弱、2=正常、3=過重、4=肥胖、5=極度肥胖)
12    #####
13    #練習1: 請修改本行程式碼，幫每位病患計算其bmi
14    bmi = 0
15
16    #####
17    if i == 0:
18        index = '極度虛弱'
19    elif i == 1:
20        index = '虛弱'
21    elif i == 2:
22        index = '正常'
23    elif i == 3:
24        index = '過重'
25    elif i == 4:
26        index = '肥胖'
27    elif i == 5:
28        index = '極度肥胖'
29    else:
30        index = '資料錯誤'
31    #####
32    #練習2: 請修改以下程式碼，使用bmi，幫每位病患分類其體態(bodysize)
33    bodysize = "請判斷"
34    #####
35    #顯示各筆資料
36    #round(浮點數,小數位數)用於小數的四捨五入
37    print(g, h, w, round(bmi,2), index, bodysize)
38    #####
```

(a) Kaggle BMI 資料集教材案例

二、重複多張圖片

畫布尺寸與圖示座標計算方式說明：

https://jamboard.google.com/d/16ok_Vs5Z0qnKat7SR16O0MWHsQAuh9-niKy62bs

```
[ ] 1 from PIL import Image, ImageOps
2 from IPython.display import display
3
4 number = 10 #圖示個數
5 margin = 5 #圖示間距
6 width = 50 #圖示寬度
7 height = 50 #圖示高度
8
9 im = Image.open("man.png")
10 im = im.resize((width, height))
11
12 #建立新畫布
13 #第一參數: 字串"RGBA"為「三原色+透明」模式
14 #第二參數: 元組(畫布寬度, 畫布高度)
15 canvas = Image.new("RGBA", ((width+margin)*number, height))
16 for i in range(number):
17     #在畫布中貼入圖示
18     #第一參數: 圖示變數
19     #第二參數: 圖示左上角位置, 元組(x, y)
20     canvas.paste(im, ((width+margin)*i, 0))
21 #顯示畫布
22 canvas
```



(b) 多圖示象形圖作業樣板案例

圖三、教材案例

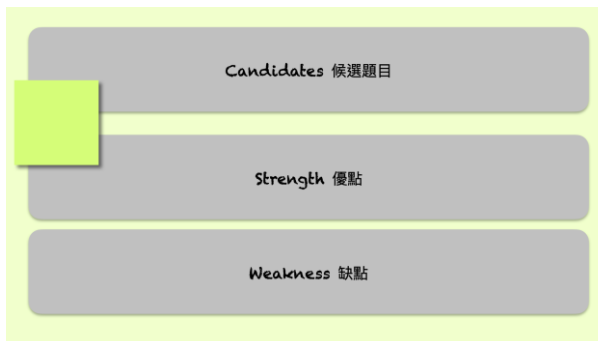
(2) 作業樣板案例：以作業二「多圖示象形圖」為例

此作業於第 5 週之 for 迴圈的單元發布，為期兩週，並於發佈隔週提供作業提示。本作業請學生自行蒐集與社會不平等議題或其他 SDGs 相關的事實數據（圖三(b)），並自行下載適合的 png 圖示，用象形圖的方式呈現資料視覺化的結果。在作業樣板設計上，老師會事先提供如何運用 pillow 套件匯入 png 圖示，並以 for 迴圈讓圖示重複數次；此外，也提供按比例在圖示中著色的函式，讓學生能夠用顏色來表達小數的資料。此作業體現了「弱結構問題解決」的特點，老師教授語法並提供樣板，讓學生自行尋找題目，包括事實數據與圖示，表達自己對於 SDGs 的關心與理解。每位學生所交的作業都不會相同（請參考 6.1.3 作業成果），能夠表現學生的創造力。但也由於這個原因，評分標準也必須有所設計，讓標準程式作業訂為 80 分，剩餘 20 分可以讓學生發揮個人想法，讓學生願意自我追求更好的程式內容。

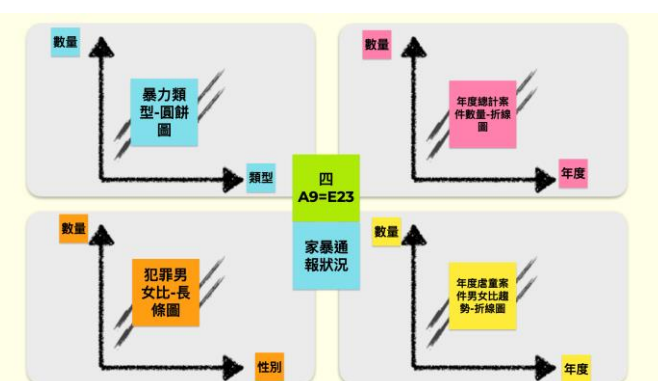
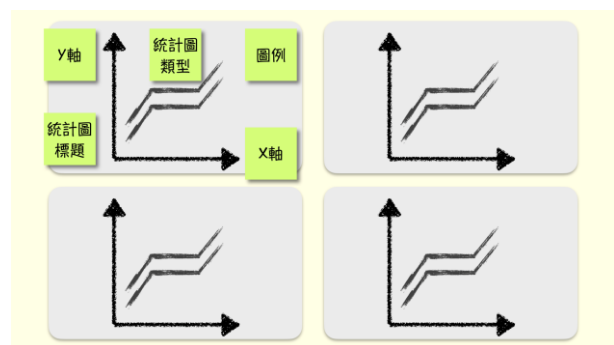
- 符合要求（60 分）：基本要求包括成果必須運用 for 迴圈來表達多圖示象形圖。
- 正確執行（80 分）：學生必須保留執行畫面，以示無錯誤訊息。除此之外，同學所繪製的象形圖，也必須與學生所提供的事實數據一致。
- 加值創意（100 分）：學生能夠發揮個人創意，並能夠清楚表示事實數據。此外，同學的註解也必須完整。
- 抄襲（0 分）、未繳交（20 分）：抄襲的定義是使用他人程式碼但未以註解說明來源，以致作業內容或程式結構過於雷同。這裡特別將未繳交的分數訂得較高，是希望向學生表達抄襲的嚴重性。

(3) 期末專題：資料視覺化

期末專題為團體報告，學生以 6 人為一組。專題要求學生運用 Python 程式來探索與「社會不平等」或其他 SDGs 相關議題，尋找公開資料集，並以程式繪製成統計圖；專題也允許學生自行延伸應用或開發服務，進一步展示程式的實際應用與創新能力。如前所述，實體班與遠距班皆舉辦實體專題工作坊，在工作坊中，老師會先進行 pandas 與 matplotlib 套件語法教學，讓每組學生都具有能夠用程式讀取資料集與繪製統計圖的基本能力。除此之外，老師也會運用 Google Jamboard 引導每組學生分析題目的優缺點(圖四(a))，以及規劃統計圖的安排(圖四(b))。只可惜 Jamboard 將於 2024 年底終止服務，未來老師可以使用其他類似功能的平台(例如 Miro)來協助學生進行討論。



(a) 題目分析 Jamboard



(b) 統計圖規劃 Jamboard

圖四、專題引導教材

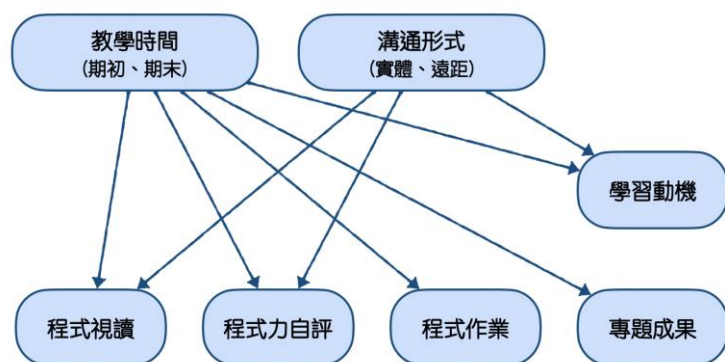
在專題評分上，分為個人作業、專題成果、組內互評與組間互評。以下說明各項評分標準。

- 個人作業（總成績 5 分）：此作業於工作坊前繳交，讓學生可以事前探索資料集並熟悉簡單的程式語法。讓學生在工作坊當天有基本的能力。
- 專題成果（總成績 15 分）：授課老師以以下三個指標評分。
 - 重要性：專題探討具有社會價值的議題，符合當前的需求或趨勢，或者融入了以人性為核心的設計理念。
 - 完整性：專題的程式功能能夠正確執行，沒有錯誤訊息，程式碼與註解文件完整，並能夠整合了多方面的資料集以支持論點。
 - 創意性：專題展現獨特的想法，融入了趣味設計，或者具有進一步應用或擴展的潛力。
- 組間互評（總成績 5 分）：各組依據前述老師指標，即重要性、完整性、創意性給分。
- 組內互評（總成績 5 分）：組員以專題貢獻度的角度互相評分。若貢獻度平均 3 以上，此項目給予滿分；反之，將依據貢獻比例給分，情節嚴重者專題 0 分計算。

5. 研究設計與執行方法 (Research Methodology)

5.1 研究架構

本研究研究架構如圖五所示。本研究採用準實驗研究法，自變項包括教學時間（期初、期末）與溝通形式（實體、遠距班），依變項包括程式視讀能力、程式力自評與學習動機。除此之外，本成果報告也將摘錄學生的程式作業作為學習歷程展現，並呈現學生專題成果作為質性總結性資料。



圖五、研究架構

表二、資料收集

人數	實體	遠距
修課人數	45	110
期初測驗人數	43	98
期中考人數	38	99
期初問卷	24	101
期末問卷	44	86

5.2 研究對象與場域

本研究的對象為大一新生，主要為 112 學期入學的學生。學生還可以根據自己的需求選擇參加實體班或遠距班。實體課程在電腦教室中進行，而遠距課程將透過 Cisco Webex 線上會議平台進行同步教學和分組討論。雖然授課形式有異，但兩班的期中測驗與期末專題，皆採用實體形式。在資料分析階段，我們將排除高年級學生以及跨校選修的學生，資料收集請見表二。

5.3 研究工具

(1) 程式視讀測驗

本研究於第 1 週實施程式視讀測驗前測，並於第 10 週實施後測（即期中考），作為程式視讀能力的研究工具。期中考與前測為自編測驗，由校內所有主授教師共同審題，20 道選擇題，分別考核範圍內的基礎語法，並可分為三種層級的題型：10 題定義題、6 題預測題與 4 題除錯題。前測則包括 5 題定義題、5 題預測題，題目難度與期中考相同。以下為三種題型之定義，題型範例可參考附件。

- 定義題：此部分測驗 Python 基本語法規則。題目可能包括變數、資料型別、運算子、容器變數、條件敘述與迴圈的基本使用。認知能力主要為記憶層次。
- 預測題：此部分要求學生預測給定程式碼的執行結果，或是根據已知的程式碼結果來判斷程式片段中的語法缺失。此類題目著重於檢驗學生對程式邏輯和執行流程的判斷力，屬於理解層次的認知能力。
- 除錯題：這部分讓學生判斷程式碼中是否存在錯誤，並要求學生指出錯誤原因或提出修正方法。此題型主要評估學生的程式除錯能力與問題解決技能，屬於後設認知能力。

(2) 程式力自評問卷

本研究根據馬尚彬老師在演講中所提出的大學程式教育的資訊素養—讀（程式視讀）、寫（程式編寫）、合（軟體統合）與測（軟體測試）四個維度，邀請校內外專家制定一份程式力的自評問卷。本問卷擬於校內程式相關課程實施，其中包括 10 題，每個題項為敘述句，其回答皆採李克特氏五等量表，選項由「完全不同意」到「完全同意」。其中通識涵蓋其中 4 題，分別對應到四個維度中最基礎的指標，分別描述如下。

- 程式視讀：我能夠理解程式語言並預測執行結果。
- 程式編寫：我能夠編寫小型程式解決基本問題。
- 軟體統合（通識指標改稱為程式表達）：我能夠運用註解與流程圖表達我的程式邏輯。
- 軟體測試（通識指標改稱為程式除錯）：我能夠除錯自己編寫的程式碼。

本自評問卷於第 1 週與期末專題工作坊實施，作為程式力自評的前後問卷。

(3) 學習動機問卷

本研究問卷採用 Pintrich 等人(1993)的動機學習策略問卷(Motivated Strategies for Learning Questionnaire, 簡稱 MSLQ) 其中的動機分量表。此量表包含 31 題，並分為 6 個維度，包括內在目標導向（4 題）、外在目標導向（4 題）、任務價值（6 題）、學習控制信念（4 題）、自我效能（8 題）與測驗焦慮（5 題）。每個題項為敘述句，其回答皆採李克特氏五等量表，選項由「完全不同意」到「完全同意」。除此之外，問卷最後附加一個敘述「我願意選修進階程式設計課程」，選項一樣由「完全不同意」到「完全同意」，用來了解學生是否有意願繼續修習進階程式設計課程。本問卷作為學習動機的前後問卷，於第 1 週與期末專題工作坊與程式力自評量表合併實施。

6. 教學暨研究成果 (Teaching and Research Outcomes)

6.1 教學過程與成果

(1) 程式視讀能力測驗

研究對象的程式視讀能力測驗總分與各題型正確率如下表所示。從表 2 中可以發現，定義題、預測題與整體的前後測皆有顯著差異，顯示教學基本的學習成效。無論是實體班還是遠距班，這些題型的學習成效都得到了顯著提升，說明這兩種教學模式對學生在基礎概念和程式預測能力的發展均有幫助。除此之外，在遠距班與實體班後測成績比較顯示，定義題、預測題與整體的兩班間皆無顯著差異，說明無論是實體還是遠距教學，對這些能力層級的學習效果一致。

然而，在除錯能力上，遠距班的後測成績顯著優於實體班，這或許反映出遠距學習模式在此類題型上提供了某些優勢。可能的原因包括，遠距班學生可能由於依賴自我導向的學習方式，因而更擅長使用除錯工具或是發揮其後設認知能力，進而在除錯題型中表現得更為突出。但確切原因仍有待未來進一步研究。

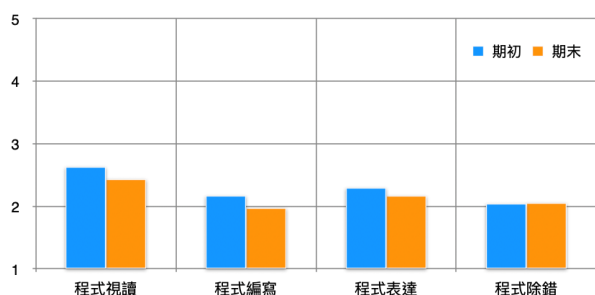
表三、程式視讀能力測驗各題型正確率

112 學年	實體班 (N=38)			遠距班 (N=99)			$t_{\text{遠距-實體}}$
	$M_{\text{pre}} (SD)$	$M_{\text{post}} (SD)$	$t_{\text{post-pre}}$	$M_{\text{pre}} (SD)$	$M_{\text{post}} (SD)$	$t_{\text{post-pre}}$	
定義	33.49 (23.39)	77.12 (17.69)	9.17***	26.12 (22.13)	75.66 (19.38)	16.76***	-0.41
預測	26.05 (18.79)	78.07 (15.55)	13.15***	18.98 (19.34)	75.93 (18.78)	21.02***	-0.62
除錯	n/a	49.34 (28.76)	n/a	n/a	61.36 (31.80)	n/a	2.03*
整體	29.77 (19.58)	71.84 (14.26)	10.71***	22.55 (17.48)	72.88 (17.84)	20.05***	0.32

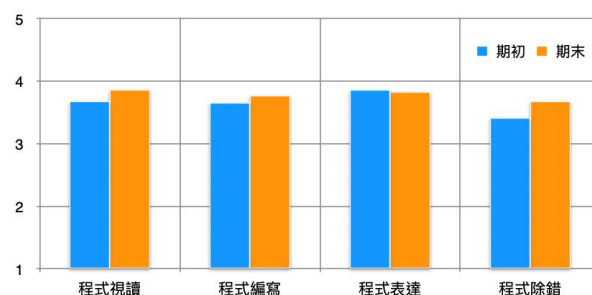
Note: * <0.05 , ** <0.01 , *** <0.001

(2) 程式力自評

在程式力的自評問卷結果中，本研究主要探究在程式的四項基本能力：程式視讀、程式編寫、程式表達、程式除錯。由下圖可知，這四項能力在期初與期末比較上皆有顯著進步。有趣的是在前三項能力，即程式視讀、程式編寫、程式表達等能力上，兩班無顯著差異；而在程式除錯能力上，遠距班顯著優於實體班，此結果與測驗結果一致。



(a) 期初



(b) 期末

圖六、程式力自評

(3) 作業成果

在本研究中增加一項新的作業，請學生運用迴圈程式繪製多圖示象形圖，用於表示社會不平等議題的真實數據。本作業除了讓同學練習迴圈語法，也藉此讓同學熟悉資料視覺化的初步應用，加深對社會問題的意識。以下是本屆學生的一些作品：

- 年齡人口比例：學生通過程式繪製象形圖，呈現台灣不同年齡段的人口比例，特別是老年人口的比例，用於凸顯台灣社會的高齡化問題。這不僅讓資訊接受者容易理解高齡化問題的嚴重性，也能夠強調隨著高齡人口增加，社會對長期照護和醫療資源的需求將逐漸增大。
- 加班比例：學生透過象形圖展示台、中、日、韓的加班比例，形象化地揭露台灣與鄰近國家的過勞問題。這種視覺化方式讓數據變得更加生動，能夠幫助資訊接受者迅速理解過勞在台灣的普遍性，並提醒社會重視工作與生活平衡的議題。
- 西醫師的性別比例：此象形圖展示了台灣西醫師中男女比例的差異。學生運程式將這一性別不平衡的現象視覺化，強調了醫療領域中的性別分布問題。這樣的展示不僅能夠揭示性別差異，還引發對醫療行業中性別平等問題的思考。
- 全球飢餓人口比例：學生利用象形圖來呈現全球不同地區的飢餓人口比例，直觀顯示出全球飢餓問題的分布狀況。透過這種視覺化，資訊接受者能夠迅速理解飢餓問題在世界範圍內的嚴重性，從而引發對全球糧食分配不均與人道援助的關注。



(a) 年齡人口比例



(b) 加班比例



(c) 西醫師性別比



(d) 全球飢餓人口比

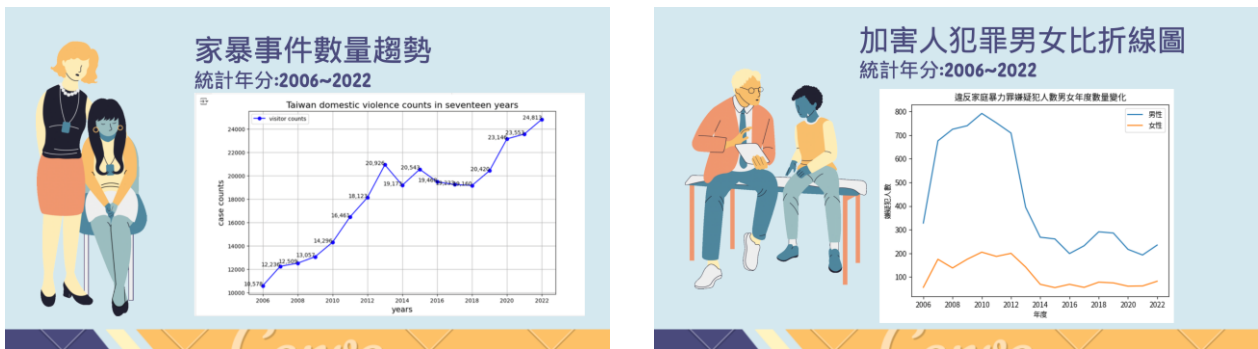
圖七、多圖示象形圖作業案例

整體來說，學生的作業選題符合當前 SDGs 的教學目標貼近社會現實，也涵蓋了全球性與地區性的多樣化議題。這些作業展示了學生能夠有效運用資料視覺化的程式語法來強化數據的傳達效果，也具備良好的社會觀察力與批判性思維。不僅如此，這些作業也能夠引發資訊接受者對重要社會問題的思考。

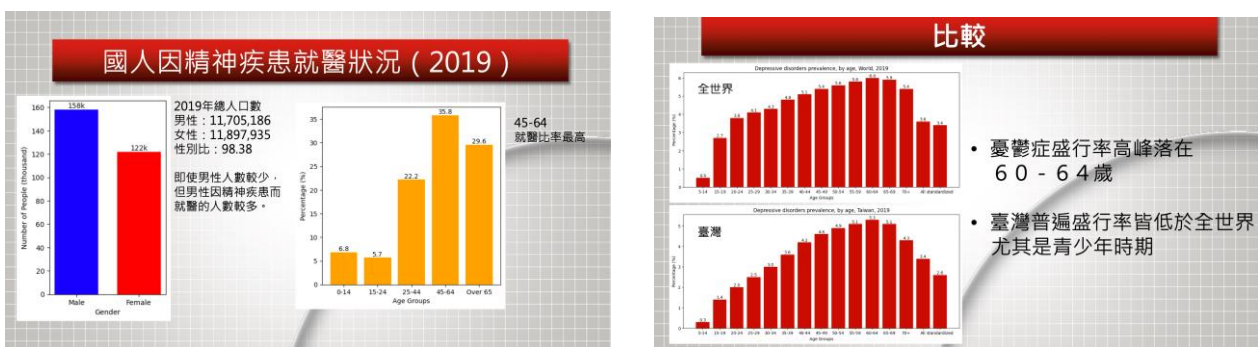
(4) 期末專題成果

在期末專題報告中，各組同學運用了資料視覺化的程式語法進行資料分析與統計圖展示，探討各自所關心的「社會不平等議題」。他們透過系統化的數據整理與統計圖表，呈現了這些議題的趨勢與挑戰，並提出了相關建議。這些專題不僅展現了同學們對技術工具的掌握能力，更反映了他們對社會問題的關注與責任感。透過這樣的專題報告，他們有效將所學應用於實際

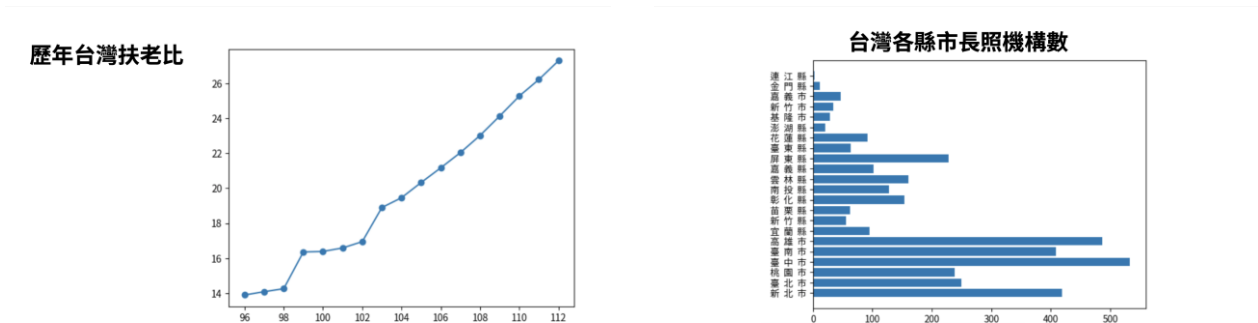
問題分析，並將這樣的問題意識帶到未來課程進一步探索或了解原因，或實際設計解決方案。



(a) 家暴通報案件



(b) 憂鬱症年輕化



(c) 人口老齡化問題與長照機構

圖八、期末專題成果案例

- 家暴案件通報：這組同學在報告中特別提到，台灣《家庭暴力防治法》是亞洲首部家暴相關法律，其立法契機來自 1993 年的鄧如雯事件。為深入探討家暴議題，該組同學運用資料視覺化的程式語法，整理並分析過去超過 15 年的公開資料。他們製作了多張統計圖，詳細展示了暴力類型、家暴事件數量的趨勢變化、加害人性別比例（男女比）、以及致死虐童案件的男女比趨勢等數據。值得一提的是，為了更清晰地呈現數據，這組同學在製作圖表時，特別自學了資料標籤與格線的程式語法，使他們能夠更精確地標記重要數據點，並使圖表的閱讀和理解更加直觀與清晰。這不僅展示了他們對資料視覺化技術的掌握，也反映出他們對家暴議題的深入理解及專注研究的態度。

- 憂鬱症年輕化：這組同學的專題探討「憂鬱症年輕化」的議題，報告中特別提到台灣近年來頂尖大學生接連自殺事件，引發他們對本議題的重視。他們運用了資料視覺化的程式語法，整理了全球和台灣的精神疾病數據，包括各年齡層的憂鬱症盛行率、精神疾患就醫狀況等統計圖。他們發現全球與台灣的精神疾病趨勢大致相同，並指出雖然台灣憂鬱症的盛行率普遍低於全球，但這並不代表問題不嚴重。透過這些視覺化圖表，他們有效比較了憂鬱症在台灣和全球的盛行趨勢，進一步加強了觀眾對憂鬱症年輕化現象的關注。
- 人口老齡化問題與長照機構：這組同學的專題探討「人口老齡化問題與長照機構」，報告中詳盡地探討了台灣面臨的高齡化挑戰及其對長期照護機構的需求和設置情況。他們運用了資料視覺化技術，呈現了台灣各地區近年來的扶老比趨勢、各縣市長照機構的設立情況。首先，他們展示了歷年來台灣扶老比的逐年上升趨勢，強調隨著人口老化，長期照護需求不斷增加。然而，他們也進一步指出，目前老年服務資源的分配有不平衡的現象。某些地區的長照機構數量雖然充足，但因老年人口較少，導致許多資源閒置，無法充分利用；而在老年人口密集的地區，照護資源則遠遠不足，無法滿足需求，這種情況導致老年人難以獲得及時的照顧，嚴重影響他們的生活品質。藉此，該組同學也提出資源分配的建議。

這些期末專題展現了學生能夠透過程式技術進行資料視覺化，並且能夠將技術應用於探討真實的社會議題。學生嘗試將複雜的社會數據轉化為清晰易懂的圖表，讓資訊接受者能夠更直覺地理解這些重要議題的趨勢與挑戰，符合本研究的初衷。

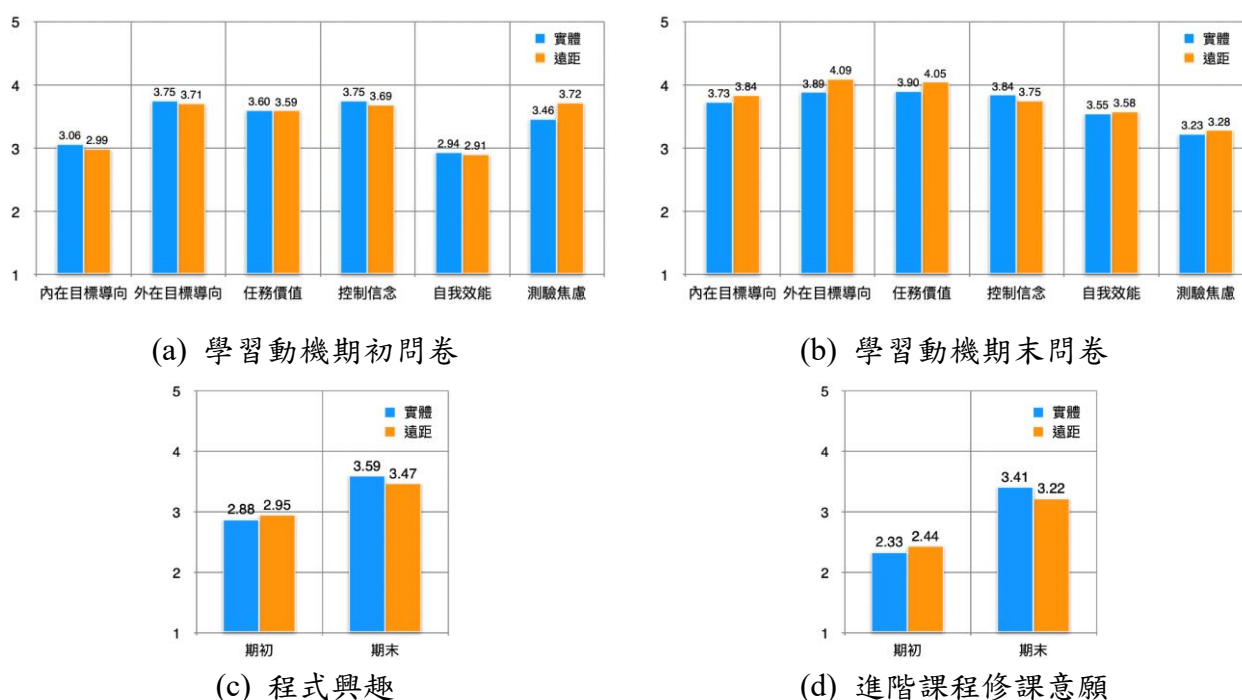
(5) 學習動機

學生期初與期末的學習動機問卷如下圖所示。在期初時，實體班與遠距班皆無顯著差異，但可以觀察到學生期初的外在目標導向較高，代表學生主要是因為必修課，為了取得學分而來。相對的，內在目標導向較為中立，代表學生不會為了自我成長而學習程式。除此之外，非資訊領域學生學習陌生領域的自信心不足，並且有較高的測驗焦慮。對於程式設計這類技術性強的課程，學生可能擔心自己的學習能力是否足以應對其挑戰，尤其是那些沒有相關基礎的學生。所幸學生的任務價值傾向正向，顯示學生大多覺得程式設計重要，這與前幾屆的學生略有不同，可能代表 108 課綱的學生較重視資訊素養的相關能力，讓他們對程式的應用價值有初步了解。

期末兩班之間亦無顯著差異，而在前後差異上，學生的內在目標導向（實體班 $t=3.59$, $p<0.001$ ；遠距班 $t=6.48$, $p<0.001$ ）與自我效能（實體班 $t=3.23$, $p<0.01$ ；遠距班 $t=5.19$, $p<0.001$ ）皆有顯著提升。此結果顯示學生在期末開始為自我成長而學習程式，也對自己學習程式有信心。而在任務價值與測驗焦慮上，實體班無顯著差異，而遠距班任務價值顯著提升（ $t=3.85$, $p<0.001$ ），而測驗焦慮顯著下降（ $t=2.72$, $p<0.01$ ）。此結果顯示遠距班學生在學習過程中對程式的價值和實用性有了更高的認同，這可能是因為遠距學習讓他們有更多機會接觸到程式教學的線上學習資源與公開資料集，並且看見程式對於議題探索與各自專業的助益。而遠距班的自主學習能力較強，在經過一學期的學習後，也比較不害怕學習程式。

除此之外，本研究也特別在期初與期末詢問學生對程式的學習興趣，以及未來是否願意繼續修進階課程。從圖九(c)中可以注意到，實體班和遠距班期初與期末的樣態相似，期初對程式的學習興趣較為中立，但期末傾向正面。而圖九(d)則顯示學生對於進階課程的修課意願，兩班

在期初時都傾向負面，代表學生起初不太願意修進階程式課程，但期末則轉變為中立的態度。雖然仍有努力空間，但這也表示有部分學生開始不排斥學習進階程式課程。



圖九、學習動機

6.2 教師教學反思

第一、弱結構問題程式作業有助於自我追求：弱結構問題要求學生自行選擇題目與資料，鼓勵學生投入自己的興趣與理解，運用所學發揮創造力。但每位學生提交的作業都是不同的，設計統一的評分標準變得更加困難。為此本研究將作業評分標準提高。在資訊領域課程中，正確執行的程式可取得滿分，但此類作業通常都有標準答案，或涉及數學，不一定適合非資訊領域的學生。建議可以將作業轉型為弱結構問題時，保留部分分數讓學生追求正確以外成果，例如更友善的輸入與輸出，或是更完整的程式碼註解等。這樣的作業對學生有益處，但也可能增加老師批改作業的工作量。教學方便性的優化則可在未來進行更深入的思考與設計。

第二、真實問題探究有助於提升學習目標的意義性：相較於虛擬問題，非資訊領域學生傾向關心真實問題。真實問題通常涉及現實生活中的挑戰，能夠激發學生的學習動機。此外，真實問題往往需要跨領域知識來解決，這不僅拓展了學生的視野，也讓他們有機會體驗到自己能夠解決複雜問題的成就感。更重要的是，本課程結合 SDGs 議題，讓學生運用程式技術探索開放資料集，而真實資料通常具有巨量與複雜的特性，這些特徵非常適合通過程式設計進行處理和分析。因此，將真實問題引入課程不僅提高了學習的實用性，還讓學生在面對現實挑戰時具備更強的解決問題能力。另一方面，學生也可以透過本課程從宏觀與量化的角度認識 SDGs 議題，未來在其他通識課程中深入探討進階議題，或在進階程式課程中設計解決方案。

第三、生成式 AI 的挑戰與共存：當前生成式 AI 已經能夠產生程式碼，特別是基礎程式設計此類基礎課程，AI 所產生的程式碼幾乎都是正確的。如果課程仍然採用有標準答案的程式

任務，學生可能不需要思考就可以獲得正確解答。本計畫將課程調整為「弱結構問題解決」與「真實問題探究」，某種程度上已經避免生成式 AI 所帶來的危機。即使學生運用生成式 AI，也僅能獲得 80 分，而必須花費心力做出有別於 AI 的成果才有可能取得高分。因此，這樣的課程設計反而可以鼓勵學生多運用生成式 AI，完成初步語法，協助除錯，鼓勵學生創新思維，運用程式技術探究與解決真實問題。

6.3 學生學習回饋

本節末摘錄了學生的期末學習回饋，根據回饋內容，本課程的優點主要有兩個方面。首先是成就感，這可能得益於教學中的鷹架策略，幫助學生循序漸進地掌握知識，或者是解決真實問題帶來的滿足感。其次是基礎與進階兼具，讓基礎學生能夠學習正確的語法，而進階學生則有機會追求更高水準的軟體開發挑戰。

- 已本來近乎零基礎來說，自己做完作業的時候還算有成就感。
- 上次寫程式已經是高一了，這次學已經能夠自己排除錯誤了~覺得有收穫!
- 老師的課程安排很詳細，內容也超級豐富，能夠學習到很多東西，難易度也會分開供比較進階的同學學習更難的程式，覺得收穫滿滿。

然而，學生也提到本課程兩個主要的缺點。首先是課程仍然過難，部分學生反映課程對於學習速度較慢的學生難度較大。其次是作業與工作坊的勞累感，一些學生感到作業量較多，專題工作坊的安排也讓他們感到疲憊。這些回饋提供了未來課程改進的方向。

- 內容可以再淺一點，我覺得有點難，但也可能是我太爛。
- 不太理解學校把程式設計當必修的目的，這堂課作業超多，超級累人。
- 工作訪好累。

7. 建議與省思 (Recommendations and Reflections)

本計畫重新設計「基礎程式設計」課程，結合社會不平等議題，將「資料視覺化」作為真實問題探究的教學目標，引導學生學習程式基礎，並讓學生感受到程式的功能。研究結果顯示，此教學方式不僅能夠提升程式能力，也能提升內在目標導向與自我效能等學習動機。在執行本計畫後，筆者有以下省思與未來規劃。

第一、強化視覺化圖表的詮釋能力：期末專題有時會發現學生解釋統計圖的能力不足，有時可能誤解或過度解釋圖表的意義。未來課程應著重提醒學生掌握資料視覺化的基本原則，並強化圖表的正確詮釋與解讀，避免因誤解數據而產生錯誤結論。

第二、培養程式除錯技能：由於部分學生反映課程難度較高，可能在撰寫程式作業時頻繁遇到錯誤，進而產生學習挫折感。未來課程可以教授學生如何善用生成式 AI 輔助除錯，幫助他們更有效地解決程式問題，從而提升自主學習能力和問題解決能力。

第三、引導進階學生學習合作開發方法：對於已經熟練掌握基礎語法的學生，應引導他們進一步學習軟體工程的開發方法，如版本控制、模組化開發、測試驅動開發等進階技能。這些能力將有助於小組專題的合作與程式開發任務的統合，提升專案管理和團隊合作的效率。

二、參考文獻 (References)

- Angeli, C. (2022). The effects of scaffolded programming scripts on pre-service teachers' computational thinking: Developing algorithmic thinking through programming robots. *International Journal of Child-Computer Interaction*, 31, 100329.
- Bierer, B. (2018). When more is less: the Risks of Overscaffolding Learning. In R. J. Harnish, K. R. Bridges, D. N. Sattler, M. L. Signorella, & M. Munson (Eds.), *The Use of Technology in Teaching and Learning* (pp. 46-53), Society for the Teaching of Psychology.
- Cheng, H. N., Yang, E. F., Liao, C. C., Chang, B., Huang, Y. C., & Chan, T. W. (2015). Scaffold seeking: a reverse design of scaffolding in computer-supported word problem solving. *Journal of Educational Computing Research*, 53(3), 409-435.
- Medeiros, R. P., Ramalho, G. L., & Falcão, T. P. (2018). A systematic literature review on teaching and learning introductory programming in higher education. *IEEE Transactions on Education*, 62(2), 77-90.
- Pintrich, P. R., Smith, D. A., Garcia, T., & McKeachie, W. J. (1993). Reliability and predictive validity of the Motivated Strategies for Learning Questionnaire (MSLQ). *Educational and psychological measurement*, 53(3), 801-813.
- Popat, S., & Starkey, L. (2019). Learning to code or coding to learn? A systematic review. *Computers & Education*, 128, 365-376.
- Rau, M. A., Zahn, M., Misback, E., Herder, T., & Burstyn, J. (2021). Adaptive support for representational competencies during technology-based problem solving in chemistry. *Journal of the Learning Sciences*, 30(2), 163-203.
- van de Pol, J. (2012). *Scaffolding in teacher-student interaction: Exploring, measuring, promoting and evaluating scaffolding* (Doctoral thesis). University of Amsterdam, Amsterdam, Netherlands.
- van de Pol, J., Volman, M., & Beishuizen, J. (2010). Scaffolding in teacher-student interaction: A decade of research. *Educational Psychology Review*, 22(3), 271-296.
- Zhang, J. H., Meng, B., Zou, L. C., Zhu, Y., & Hwang, G. J. (2021). Progressive flowchart development scaffolding to improve university students' computational thinking and programming self-efficacy. *Interactive Learning Environments*, 1-18.

三、附件 (Appendix)

附件 1. 程式視讀能力測驗

題型	例題	知識點
定義	#1. 請問 <code>num /= 100</code> 與以下哪個程式碼的執行結果相同？ (A) <code>num = 100</code> (B) <code>num = num % 100</code> (C) <code>num = num / 100</code> (D) <code>num = num // 100</code>	第三週運算式 (複合指定運算子)
預測	#3. 請問若要輸出右側星號圖案，請問以下程式碼第 1 行與第 2 行中兩個 <code>range()</code> 函式裡的參數分別應該為多少？ <pre> 1 for i in range(____): 2 for j in range(____): 3 print('*', end='') 4 print() </pre> 輸出 <pre> **** **** **** </pre> (A) 3 4 (B) 4 3 (C) 3 i (D) 4 i	第八週 for 迴圈 (巢狀迴圈) 第八週 for 迴圈 (range 函式)
預測	#4. 以下程式為一個猜謎遊戲，希望使用者猜測含有 3 個字母的英文單字，請問第 7 行與第 10 行空白處的兩處程式碼分別應該如何表示，才能實現以下功能？ ● 輸入正確答案，顯示 CORRECT!，並結束程式 ● 輸入 3 個字母的錯誤答案，顯示 WRONG!，並重新輸入 ● 輸入非 3 個字母的錯誤答案，顯示 Hint: 3 letters! 的提示，並重新輸入 <pre> 1 answer = 'fox' 2 guess = '' 3 print('An animal with 3 letter!') 4 while True: 5 guess = input('Guess a word:') 6 if guess == answer: 7 _____ 8 elif len(guess) == 3: 9 print('WRONG!', end=' ') 10 _____ 11 print('Hint: 3 letters!', end=' ') </pre> (A) break、continue (B) continue、break (C) continue、continue (D) break、break	第六週條件敘述 (if...elif...) 第七週 while 迴圈 (break) 第七週 while 迴圈 (continue)
除錯	#5. 在以下程式碼中，變數 <code>stats</code> 為一個包含身高、體重兩個數值的元組，請問執行該程式後，輸出內容為何？ <pre> 1 stats = (170, 60) 2 stats[1] += 10 3 print(stats) </pre> (A) (180, 60) (B) (170, 70) (C) (180, 70) (D) 程式錯誤	第四週元組 (元組無法變更)
除錯	#6. 以下程式有著無限迴圈的問題，若要輸出「1~100 之間所有奇數」，請問應如何修改？ <pre> 1 num = 1 2 while num != 100: 3 print(num) 4 num = num + 2 </pre> (A) 第 1 行改為： <code>num = 0</code> (B) 第 1 行改為： <code>num = 100</code> (C) 第 2 行改為： <code>while num < 100:</code> (D) 第 2 行改為： <code>while num > 100:</code>	第七週 while 迴圈 (無限迴圈)

附件 2. 學習動機問卷

內在目標導向 Intrinsic Goal Orientation

1. 在本課程中，我喜歡有挑戰性的教材，才能學到新東西。
2. 在本課程中，我喜歡那些能引起我好奇心的教材，就算困難也沒關係。
3. 在本課程中，對我來說最令人滿足的事情是盡可能深入地理解內容。
4. 在本課程中，如果我有機會，我會選擇那些我可以學到東西的課堂作業，就算不保證可以獲得好成績。

外在目標導向 Extrinsic Goal Orientation

1. 現在對我來說，在本課程中取得優良的成績是最令我滿意的事情。
2. 目前對我最重要的事情是提高我的平均分數，所以我在本課程主要關心的是獲得好成績。
3. 如果可以，我想比班上其他大部分同學都拿到更高的成績。
4. 我想要在本課程中表現出色，因為向我的家人、朋友或其他人展示我的能力很重要。

任務價值 Task Value

1. 我會將本課程所學內容應用到其它課堂上。
2. 學會本課程的內容對我來說是重要的。
3. 我對於本課程的內容範圍非常感興趣。
4. 我認為本課程內容對我學習很有用。
5. 我喜歡本課程的主題。
6. 理解本課程的主題對我來說非常重要。

學習信念控制 Control of Learning Beliefs

1. 如果用合適的方法，我就有能力學會本課程的教材內容。
2. 如果我不學習本課程教材，那是我的錯。
3. 如果我夠努力，一定可以了解課程的教材內容。
4. 如果我不理解此課程的教材內容，可能是我不夠用功。

學習與表現的自我效能 Self-Efficacy for Learning and Performance

1. 我相信我會在本課程中取得優秀的成績。
2. 我確信我可以理解本課程最複雜的教材。
3. 我有信心可以理解本課程的基本概念。
4. 我有信心可以理解老師所教的最複雜的教材。
5. 我有信心我可以在本課程的作業和考試表現優秀。
6. 我期望在本課程中表現得很好。
7. 我確信我能掌握本課程所教的技巧。
8. 綜合考量到這門課的難易度、老師與自身能力，我認為我會表現良好。

考試焦慮 Test Anxiety

1. 每當考試時，我會想到自己的表現是否會比其他同學遜色。
2. 每當考試時，我會想到試卷中我不會回答的那些題目。
3. 每當考試時，我會擔心不及格的後果。
4. 每當考試時，我會感到不安與心煩意亂
5. 每當考試時，我會感到心跳加速