

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：PBM1120241

學門專案分類/Division：商業及管理

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

應用流程圖與問題導向式學習以減少 Python 程式設計初學者的困難
度
Python 程式設計

計畫主持人(Principal Investigator)：張丁才

協同主持人(Co-Principal Investigator)：無

執行機構及系所(Institution/Department/Program)：中華大學工業管理系

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 8 月 15 日

應用流程圖與問題導向式學習以減少 Python 程式設計初學者的困難度

一、本文 (Content)

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

(1) 教學實踐研究計畫動機

「Python 程式設計」課程為工業管理系學生的必修課，其課程教學目標係介紹 Python 程式語言之語法撰寫與其應用，使學生具備 Python 程式設計之能力及對進階的應用有所了解。選擇 Python 程式語言做為本系學生學習程式設計的入門語言，比起以 C 或 C++ 做為初學者的第一個程式語言，算是容易許多。然而，學習 Python 程式語言，對初學者而言，依然容易枯燥無味和感到困難，對於低學習成就或運算邏輯較差的學生，更是往往會造成學習挫敗而導致焦慮感上升和學習動機低落。

(2) 教學實踐研究計畫主題及研究目的

本教學實踐研究計畫的主題是「應用流程圖與問題導向式學習以減少 Python 程式設計初學者的困難度」，主要想透過使用 Flowgorithm 來學習應用流程圖來加強流程的概念和減少程式設計入門時的困難。應用問題導向學習，以生活化的問題做為解題的對象，使學生更感興趣於學習。在這門課程中，我們採用部份個人和部份分組的方式進行問題導向式學習，希望能達到以下目標：

(一) 應用流程圖減少程式設計的困難度和增加流程概念;

(二) 訓練學生獨立作業和團隊合作以及使用演算法表達之能力;

(三) 增進學生學習成效與興趣;

(四) 提升學生的自信心和減少挫折感。

2. 研究問題 (Research Question)

觀察本校工管系的學生在「Python 程式設計」這門課程教學現場的問題，發現普遍學生學習程式語言編程的技巧相當地緩慢，不容易理解上課所學的語法內容和運用時機。或許學生能將課堂上的範例練習緩慢地撰寫出程式碼，但卻(1)無法或難以舉一反三，缺乏推論和聯想的能力，學生只能解決相同類型的問題，而問題稍加變化即難以獨立完成；再者，(2)在分組的合作上，他們很少安排時間進行溝通討論，以致無法激盪出團體學習的效果，(3)有些組別的工作分配更是不均，常由學習能力強的學生主導或是負責整個專案的編程與製作，導致弱勢者無法參與其中，最後，更(4)無法在課程上獲得學習的成就感。為減少學生學習的挫折感、增進學習動機、強化運算邏輯概念和確保學習的成效，特以提出本教學實踐研究計畫，導入應用流程圖與問題導向式學習，以促進本課程的教學品質及教學成效之提升，讓學生能知悉程式設計的流程和演算法的概念，進而有學習意願而願意自主和自動學習後續更多的程式設計技能和應用。

3. 文獻探討 (Literature Review)

為了強化學生自主學習能力，過去在不少通識課程曾結合問題導向學

習法(PBL)，以問題引導為重心，激發學生主動學習的動力（楊淳皓，2017）。問題導向學習需以真實情境的案例問題為學習觸媒，並發展學生的問題解決能力（Barrows，1996）。問題導向學習法最早應用於改進傳統醫學教育，及培養學生的問題解決與決策判斷能力，並逐漸擴散至其他學科。

問題導向學習強調思考始於問題情境，從分析問題情境開始，進而產生探究問題，最後進行研究並提出報告（Barrows & Tamblyn, 1980）。

Barrett et al. (2011a, 2011b) 認為好的問題應該引起學習動機，涵蓋真實世界問題，容納多種意見，並引發持續性討論，最終增強批判性思考與合作學習能力。

PBL 已逐漸受到大學其他學科的重視。夏皓清(2022)研究了 PBL 在都市環境概論的應用，何素美(2022)研究了多元化教學策略提升技職校院非資訊科系學生學習程式語言的動機與成效，吳汶涓(2022)探討了合作學習與業師協作對程式學習成效和創造力的影響，金凱儀(2022)則研究了專題導向學習模式與微翻轉教室法於統整式 STEM 教學設計中的應用。

Kadar et al. (2021) 系統地回顧了程式設計教學與學習的困難，指出學生缺乏邏輯思維、創造性思維和批判性思維是 PBL 實施薄弱的原因之一。Santos et al. (2018) 和 Cheon et al. (2021) 指出 PBL 能提高學生參與度、滿意度和學習成效。Shivacheva & Ruseva (2021) 和 Hu et al. (2021) 提及使用

Flowgorithm 進行可視化編程有助於學生在早期學習調試和測試。Idrees & Aslam (2022) 強調了可視化編程語言 (VPL) 如 Scratch 和 Snap 的實用性及其在簡化編程上的潛力。

4. 教學設計與規劃 (Teaching Planning)

本研究對象為管理學院工業管理系修讀必修課「Python 程式設計」的學生共二班，其中一班主要為本地生 (31 位本地生，3 位越南生)，另一班則為外籍生 27 (全部皆為越南生)，實際參與計劃人數為 48 人。本研究教學設計概分成六個階段，如圖 1 所示：

第一階段：著重於 Flowgorithm 的教學和應用，讓學生熟悉用流程圖解決簡單的生活化問題，以增加學習成就感。

第二階段：著重於學習 Python 基本程式設計，使學生了解變數、選擇、迴圈和函數等基本概念。

第三階段：引入問題導向學習，使學生以個人或分組的方法，思維如何解決一些生活化的實務問題，以激勵學習意願。

第四階段：引入自然語言虛擬碼，使學生學習閱讀和使用虛擬碼及初步了解演算法，練習使用 Python 語言逐步解譯出其程式碼。

第五階段：讓學生發想和設計基於問題導向學習的衍生新問題和解決之，以啟發學生舉一反三的可能性。

第六階段：執行學習成效評量調查，包括：google 表單問卷調查、證照測驗和

取得、PBL 學習成果報告。

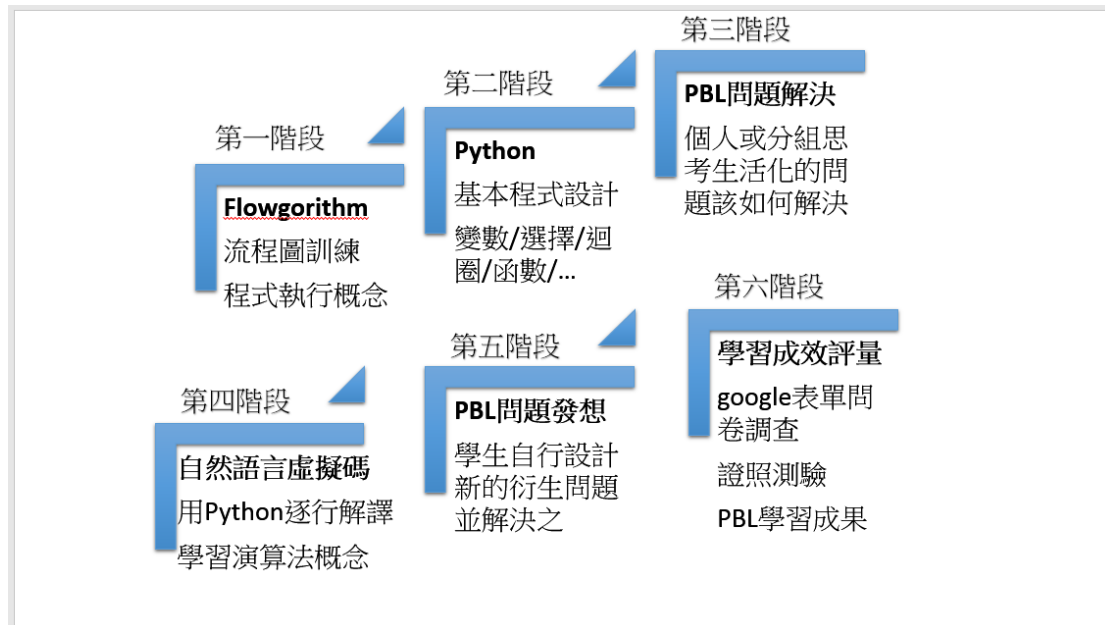


圖 1. 教學設計及研究架構

5. 研究設計與執行方法 (Research Methodology)

本課程學生學習成效評量，分為安置性評量(Placement Assessment)、形成性評量(Formative Assessment)及總結性評量(Summative Assessment)。安置性評量於上課第一週進行測驗，形成性評量則包含：隨堂測驗(成績列入作業成績)、平時作業、學生出席率、程式設計演練、課堂提問和實時反饋等。總結性評量之量化為期中上機測驗、期末管理學院會考上機測驗、校內資訊能力檢定(自由報名)、TQC Python 3 基礎程式設計證照(自由報名)。學生的學期成績評量則採用管理學院統一的標準：出席(Attendance)佔 20%，期中考(Midterm

Exam) 30%，期末會考(Final Exam) 30%，作業 (Homework) 20%。

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

(1) 教學過程與成果

本研究實施程序分為開課前、開課中與開課後。

開課前：完成教材中的教案設計、各階段學習成效評量量表設計。

開課中：第一週說明課程規劃、安排分組合作學習、進行課前安置形評量；最後三週進行教學反應回饋問卷、證照及校內資訊能力考試、期末會考。教學進行方式，先介紹和學習 Flowgorithm 以建立程式設計的基礎概念。隨後，依照課程大綱的進度安排學習 Python 程式設計，以 TQC Python 3 證照的測驗題庫做為實際演練和解題的對象。考量到學生大多數是第一次接觸到文字式的程式設計，故著重基本程式語言的語法和邏輯思考。示範解題時，採用先說明要解決的問題、徒手示範解題的邏輯程序、以 Flowgorithm 示範解題的流程，最後才以 Python 的方式解題和請學生自由練習，而整過程皆錄影供學生課後下載觀看。而接近學期末時，則引進 PBL 學習方式，強化學生的學習思維和增進學習動力。

A. 學習成效評量問卷分析

甲、基本資料分析：參與計劃學生，男性佔 58%、女性佔 42%。

本地生佔 54%，而越南生佔 46%。高中就讀：一般高中佔 67%、綜合高中佔 23%，而高職生佔 10%。入學管道：申請入學佔 67%、甄試入學佔 19%、其他佔 14%。高中職時曾學過程式設計工具：Scratch 佔 38%、Python 佔 29%、C/C++ 佔 27%、未學過則佔 29%(越南生大都未學過)。本地生都有自己的電腦可使用，而越南生則約 10%的學生有筆記型電腦。曾上過使用「問題導向學習」教學方法的課程佔比 29%。學習方式：聽老師講課和示範佔 54%、小組討論佔 25%、實作練

習佔 17%，而自主學習和使用線上學習資源 2%。高中職時學程式設計工具時的感受，對寫程式很有興趣佔 23%，其他對程式設計感到焦慮(38%)、沒學會(38%)，則總佔比 76%。Cronbach's Alpha 值為 0.68，表示問卷的內部一致性中等。刪除題目 B2（選修課程）後 Cronbach's Alpha 值則提高到了 0.729，顯示問卷的內部一致性已經達到了可接受的水平。而平常每天使用電腦或平板電腦的總時間?(小時)(不含上課時間): 平均值為 0.92 小時，標準差為 1.51 小時。平常每天玩電腦遊戲的時間?(小時): 平均值為 2.71 小時，標準差為 2.30 小時。

乙、學習前自我認知(五點量表): 共計 47 份調查回應，問卷的 Cronbach's Alpha 值為 0.919，表明問卷具有高度的內部一致性。敘述性統計摘要為：(A) 大多數題目的平均值接近或高於 4（五點量表中的 "同意" 或 "非常同意"），這表明學生對於大多數問題的自我評價較高。(B) PR03（我會盡力完成教師要求的作業）和 PR07（我覺得將程式設計相關課程內容學好是重要的）的平均值分別為 4.26 和 4.11，顯示了學生對完成作業和學習程式設計重要性的高度認同。(C) PR05（遇到較困難的學習內容時，我通常會放棄或只學習比較簡單的部分）的標準差為 1.16，顯示了對這個問題的回答存在一定的分歧，即存在興趣和期望的個體差異。

丙、學習後自我認知(五點量表): 共計 45 份調查回應，問卷的 Cronbach's Alpha 值為 0.918，表明問卷具有高度的內部一致性。合併了學習前後自我認知問卷的數據後，得到 34 筆有效資料。經配對樣本 t 檢驗後，得到；(A)大多數題目的 p 值大於 0.05，這表明學習前後的自我認知問卷得分在大多數題

目上沒有顯著差異。(B) PR03 (我會盡力完成教師要求的作業) 和 PR13 (我希望透過此課程學習更多程式相關知識) 的 p 值接近顯著水平 (p 值分別為 0.107 和 0.106), 顯示這些問題上的變化趨勢值得進一步關注。總結:(A)PR03: 學生在學習後仍然保持較高的完成作業的努力程度, 雖然平均值略有下降, 但整體仍然較高。(B) PR13: 學生在學習後對於課程的期望有所提高, 顯示了課程在激發學生學習程式相關知識方面的正面效果。

丁、Flowgorithm 學習成效問卷(五點量表): 有效資料總共為 38 筆, Cronbach's Alpha 值為 0.981, 表明問卷具有高度的內部一致性。大多數問題的平均值在 3.5 到 4.0 之間, 顯示了學生對 Flowgorithm 學習成效的高度評價。例如, F18 (操作 Flowgorithm 過程中能夠幫助我將想法視覺化) 和 F20 (我覺得完成 Flowgorithm 作業很有成就感) 的平均值分別為 3.95 和 3.92。學生對 Flowgorithm 學習成效的評價普遍較高, 顯示了學習工具在課程中的有效性。

戊、PBL 整體滿意度調查表(五點量表): 有效資料總共為 50 筆 (有額外的學生參加調查), Cronbach's Alpha 值為 0.945, 表明問卷具有高度的內部一致性。大多數問題的平均值在 4.0 到 4.4 之間, 顯示了學生對 PBL 教學法的高度滿意度。例如, PBL07 (教師上課態度熱忱、認真、負責) 和 PBL05 (教師於授課時會引導我們思考) 的平均值分別為 4.40 和 4.28。學生對 PBL 教學法的滿意度普遍較高, 顯示了 PBL 教學法在課程中的有效性。

己、Python 程式設計學習成效問卷(五點量表): 有效資料總共為 49 筆(有額外的學生參加調查), Cronbach's Alpha 值為

0.980，表明問卷具有高度的內部一致性。最低平均值為 3.57 (P16A)，最高平均值為 4.12 (P20)，大部分題目的平均值在 3.7 至 4.0 之間，表明學生對 Python 學習成效的整體滿意度較高。使用科技接受模式分類和分析結果，如表 1 所示，皆顯示具高度的有效性。

TAM 分類	分類代表	平均分數
PEOU	感知易用性 (Perceived Ease of Use)	3.73
PU	感知有用性 (Perceived Usefulness)	3.85
BI	行為意向 (Behavioral Intention)	3.76
AU	實際使用 (Actual Use)	3.87

表 1. Python 程式設計學習成效 TAM 分類

B. 教學成果

甲、已完成之教學成果

- i. PBL 學習教案 5 件
- ii. 通過 TQC Python3 基礎程式設計證照 4 位(學生自由參加)：近年來無學生有意願參加證照考取，故算是一項進步。
- iii. 通過校內資訊能力檢定 Python3 程式設計 5 位(學生自由參加)：近年來無學生參加此項能力檢定，故算是一項進步。
- iv. 完成 15 份課程教學影音教材並放置於 ChuMoodle 線上學習平台上，程式設計解題影音教材 40 題，供修課學生隨時復習。

- v. 新增 7 份程式設計新的解題原始碼並放置於 google drive 平台上，原已有程式設計的解題影音教材及原始碼各 50 份，供修課學生隨時復習。
- vi. 各階段學習成效評量量表。

(2) 教師教學反思

教學目標的達成情況方面：本教學實踐研究計畫在課程安排了一系列教學活動，採用問題導向式的教學方式，以實務問題和 TQC Python3 證照題庫為學習核心，力求學生對於程式設計有感且能引發學習動力。課程安排首先導入 Flowgorithm 軟體學習使用流程圖設計程式和建立程式設計的基本概念，以訓練學生邏輯思維、強化程式觀念的理解及瞭解電腦的運作原理。通過觀察學生的學習成果、作業、測試和討論的參與，可知學生的反應良好，然而，越南生因語言的障礙，無法跟上預定的學習進度。對於文字式的 Python 程式設計而言，學生對於採用 Mu Editor 做為初學者的 Python IDE 工具感到滿意、易學和好用。然而學生的學習態度大都偏向被動，只少數的學生會閱讀教科書和教學材料，花在課後學習上的時間很少。

教學方法的有效性方面：基於反思的結果，現在學生用手機的頻率很高，主要以玩遊戲和觀看影片為主，玩手機時間長，接受的網路訊息複雜，越不喜歡動腦筋，不想學習。因而，問題導向學習的教案需精簡和如何引發學生的自我學習動機多多著墨。而多媒體教學資源，則需力求每段影片宜短。

學生學習方面：學生在學習程式時，有時沒有真正理解程式的用法，自己練習時，直接參考老師的操作解答，照老師的教學步驟做，上課也沒有記筆記的習慣，課後的練習太少，等到考試時不會應用。因此，若能設計一些大同小異無參考答案的習題，協助學生學習、練習與複習，應可提升學生學習成效。

(3) 學生學習回饋

對流程圖的方面的回饋：學生大都表示流程圖幫助他們更清楚地理解程式設計的結構和邏輯順序。流程圖有助於學生在著手程式設計的規劃和檢查邏輯，從而減少編碼錯誤。

對問題導向式學習方面的回饋：學生感受到通過解決實際問題，學到的技能更有實用價值。當學習活動與真實世界的情境相結合時，學生通常反應說他們的學習動機增強了。PBL 常常需要小組合作，學生反映這種模式提升了他們的溝通與團隊協作能力，然而，部份因小組成員的不配合和不積極，有時變成單兵戰鬥而失去團隊協作的本意。

對 Python 程式設計學習方面的回饋：對於初學者來說，建立程式邏輯和思維方式往往是一個挑戰，學生在理解循環、條件語句和函數等概念時會遇到困難。學生對於找出程式中的錯誤和進行除錯感到困惑和挫敗，特別是在面對複雜的程式碼時。

7. 建議與省思 (Recommendations and Reflections)

(1) 建議

以下是本教學實踐計劃的執行後的建議：

- A. 多元化學習材料：結合更多視覺與互動元素，如視頻教程和交互式程式設計練習，以增強學生的學習動力和參與度。提供程式碼範例和案例研究的多語言版本(如：越南語版本)，以幫助非母語學生更好地理解教學內容。
- B. 技術工具的適當使用：考慮引入更多初學者友好的程式設計工具，如 Thonny 或其他圖形界面程式設計軟體，以降低入門障礙。提高 Flowgorithm 工具的使用指導，例如透過教學視頻或互動教程來展示其應用方法。

- C. 課程內容的靈活調整：根據學生的學習進度和反饋調整課程難度和深度，特別是針對學習成就感較低或有語言障礙的學生。定期進行學習需求評估，如：小考，以便及時更新和調整課程內容與進度。

(2) 省思

以下是本教學實踐計劃的執行後的省思：

- A. 教學方法的適宜性：：反思使用的教學方法是否真正適合目標學生群。例如，問題導向式學習是否有效激發了學生的學習興趣和參與度？如何解決部份學生對於團體參與的怠惰問題？考慮是否有必要結合其他教學方法，如個別化學習支援或增加導師與學生的一對一互動？
- B. 學生反饋的整合：深入分析學生對課程的反饋，尤其是那些關於學習困難和挫折感的反饋，以便更好地調整教學策略。
- C. 教學資源的最大化利用：評估現有教學資源是否充分利用，並探索如何更有效地利用這些資源以支援學習，例如通過共享教學材料。省思如何有效整合新興技術，如 AI 輔助學習工具，來提升教學效果和學生互動。

二、參考文獻 (References)

1. 夏皓清(2022). 結合網路社群經營之問題導向學習在都市環境概論上的應用成效與困境之研究都市環境概論. 教育部教學實踐研究計畫成果報告, 計劃編號: PSL1090160.
2. 何素美(2022). 運用多元化教學策略提升技職校院非資訊科系學生學習程式語言動機與成效智慧物聯網程式設計. 教育部教學實踐研究計畫成果報告, 計劃編號: PBM1100980.
3. 吳汶涓(2022). 合作學習與業師協作對程式學習成效和創造力之

- 影響。育部教學實踐研究計畫成果報告，計畫編號: PEE1100819.
4. 金凱儀(2022). 應用專題導向學習模式與微翻轉教室法於統整式 STEM 教學設計：以程式設計課程為例. 育部教學實踐研究計畫成果報告，計畫編號: PEE1100768.
 5. Barrows, Howard S., & Tamblyn, Robyn M. (1980). Problem-based learning : an approach to medical education. New York: Springer Pub. Co.
 6. Barrett, Terry, Cashman, Diane, & Moore, Sarah. (2011a). Designing problems and triggers in different media. In T. Barrett & S. Moore (Eds.), New approaches to problem-based learning : revitalising your practice in higher education (pp. 18-35). London: Routledge.
 7. Barrett, Terry, & Moore, Sarah. (2011b). New approaches to problem-based learning : revitalising your practice in higher education. London: Routledge.
 8. Hu, M., Assadi, T. and Mahroeian, H. (2021), "Teaching Visualization-first for Novices to Understand Programming," 2021 IEEE International Conference on Engineering, Technology & Education (TALE), 2021, pp. 654-660, doi: 10.1109/TALE52509.2021.9678922.
 9. Idrees, M. & Aslam, F. (2022). A Comprehensive Survey and Analysis of Diverse Visual Programming Languages. VFAST Transactions on Software Engineering, 10(2), pp. 47-60.
 10. Minjong Cheon, Ook Lee, Changbae Mun, and Hyodong Ha (2022). A Study on the Factors Affecting Intention of Learning Python Programming: For Non-majors in University. International Journal of Information and Education Technology, 12(5), pp. 414-422.
 11. Rozita Kadar, Naemah Abdul Wahab, Jamal Othman, Maisurah Shamsuddin & Siti Balqis Mahlan (2021). A Study of Difficulties in Teaching and Learning Programming: A Systematic Literature Review.

International Journal of academic research in progressive education and development. 10(3), 591-605.

12. S. C. d. Santos et al. (2018). Applying PBL in Teaching Programming: an Experience Report. 2018 IEEE Frontiers in Education Conference (FIE), 2018, pp. 1-8, doi: 10.1109/FIE.2018.8658978.

13. Shivacheva, G. I. & Ruseva, N. R. (2021). Training in Programming using Innovative Means. IOP Conf. Ser.: Mater. Sci. Eng., 1031 012124, doi: 10.1088/1757-899X/1031/1/012124.

三、附件 (Appendix)

附件一、學生基本資料問卷題目

編號	題目
B01	學號
B02	本地生?
B03	選修課程
B04	性別
B05	入學管道
B06	高中職就讀
B07	有自己的桌上型個人電腦設備
B08	有自己的筆記型電腦
B09	您是否已經閱讀過任何程式設計相關的書籍或資料
B10	平常每天使用電腦或平板電腦的總時間?(小時)(不含上課時間)
B11	平常每天玩電腦遊戲的時間?(小時)
B12	平常每天上網的時間?(小時)
B13	高中職時曾上過電腦程式設計相關的課程
B14	高中職時曾學過程式設計工具
B15	高中職時曾學過程式設計工具 Scratch
B16	高中職時曾學過程式設計工具 Python
B17	高中職時曾學過程式設計工具 C/C++
B18	高中職時曾學過程式設計工具的總合時間(小時)
B19	高中職時學程式設計工具時的感受
B20	高中職的程式設計相關課程在幾年級
B21	您對所選擇的程式語言而言，你能理解:(1) 變數和資料型別
B22	您對所選擇的程式語言而言，你能理解:(2) 使用條件語句 (if-else)

B23	您對所選擇的程式語言而言，你能理解:(3) 迴圈（for、while）的運作
B24	您對所選擇的程式語言而言，你能理解:(4) 函數(function)或方法(method)的使用
B25	您更傾向於哪種學習方式
B26	我有註冊 ChatGPT 免費的帳號和使用過 ChatGPT free (如：GPT 3.5) 的經驗
B27	我有使用過 ChatGPT 付費 (如：GPT-4) 的功能和使用的經驗
B28	我有個人常用的 Google 帳號
B29	我曾上過 使用「問題導向學習」教學方法(PBL)的課程

附件二、學習前自我認知問卷調查

編號	題目
PR01	相較於班上的其他同學，我認為自己是個好學生。
PR02	相較於班上的其他同學，我預期我會有好的表現。
PR03	我會盡力完成教師要求的作業。
PR04	我喜歡很有挑戰性的事物。
PR05	遇到較困難的學習內容時，我通常會放棄或只學習比較簡單的部分。
PR06	上課時我很容易分心看其他科目的書或玩手機。
PR07	我覺得資訊科技對生活有很大的影響，例如：智慧型手機、物聯網、人工智慧、機器人、大數據…等。
PR08	我覺得在未來有很多目前的工作被機器人或人工智慧取代。
PR09	我覺得將程式設計相關課程內容學好是重要的。
PR10	我覺得程式設計很好玩。
PR11	我對寫程式很有興趣。
PR12	我希望透過此課程學習更多程式相關知識。
PR13	尚未修此課程前，我很期待本課程的開設。
PR14	我很期待能參加相關的證照考試

附件三、學習後自我認知問卷調查

編號	題目
AF01	相較於班上的其他同學，我認為自己是個好學生。
AF02	相較於班上的其他同學，我有好的表現。
AF03	我會盡力完成教師要求的作業。
AF04	我喜歡很有挑戰性的事物。
AF05	遇到較困難的學習內容時，我通常會放棄或只學習比較簡單的部分。
AF06	上課時我很容易看其他科目的書或玩手機。
AF07	我覺得資訊科技對生活有很大的影響，例如：智慧型手機、物聯網、人工智慧、機器人、大數據…等。
AF08	我覺得在未來有很多目前的工作被機器人或人工智慧取代。
AF09	我覺得將程式設計相關課程內容學好是重要的。
AF10	我覺得程式設計很好玩。
AF11	我對寫程式很有興趣。
AF12	我希望透過此課程學習更多程式相關知識。
AF13	本課程的開設符合我的期待。
AF14	我很期待能參加相關的證照考試

附件四、Flowgorithm 學習成效問卷

編號	題目
F01	我覺得 Flowgorithm 是一個很好玩的軟體。
F02	我覺得學習 Flowgorithm 令我覺得很自在。
F03	我覺得可以容易適應學習 Flowgorithm。
F04	我覺得學習 Flowgorithm 時，總覺得時間過得很快。
F05	我覺得學習 Flowgorithm 不會讓我感到緊張
F06	我覺得 Flowgorithm 操作過程很簡單。
F07	我覺得 Flowgorithm 程式設計並不會很抽象。
F08	我覺得 Flowgorithm 程式設計很容易理解。
F09	我覺得學習 Flowgorithm 很輕鬆。
F10	我覺得 Flowgorithm 課程很有趣。
F11	我覺得 Flowgorithm 軟體的使用者介面環境很方便使用。
F12	我在 Flowgorithm 課程有認真學習。
F13	我覺得學習 Flowgorithm 可加強我對程式觀念的理解。
F14	學習 Flowgorithm 讓我覺得比較理解電腦的運作。
F15	學習 Flowgorithm 程式設計讓我覺得學習程式設計很有挑戰性。
F16	給我題目，我就可以自己想出解題步驟(不需要老師提示)。
F17	我覺得學習 Flowgorithm 可引發我學習程式設計的動機。
F18	Flowgorithm 的流程圖有助於理解要表達的是什麼。
F19	操作 Flowgorithm 過程中能夠幫助我將想法視覺化。
F20	我覺得完成 Flowgorithm 作業很有成就感。
F21	未來遇到比較難解的問題，我會繼續使用 Flowgorithm 來釐清問題。
F22	我喜歡將我的 Flowgorithm 作品分享出來，讓別人來觀摩。
F23	我希望以後有機會可學習更多的程式設計課程。
F24	整體而言，我同意 Flowgorithm 程式設計課程的學習

附件五、PBL 整體滿意度調查表

編號	題目
PBL01	修讀這門課有助於提升我在本課程專業領域之知識。
PBL02	教師設計之討論議題及引導方式能啟發我的學習興趣。
PBL03	教師對本課程所安排的作業、報告或考試，能確實反映我們的學習內容
PBL04	教師於課程中運用多元且創新的教材內容。
PBL05	教師於授課時會引導我們思考，增加對議題或問題的理解，進而組織出解決方案。
PBL06	教師實施問題導向式教學之上課模式能增強我的學習效果。
PBL07	教師上課態度熱忱、認真、負責。
PBL08	我認為問題導向式教學型態的課程，相較傳統課程更具有學習效果。

附件六、Python 程式設計 學習成效問卷

編號	TAM	題目
P01	PEOU	我覺得 Python IDLE 是一個很好玩的軟體。
P02	PEOU	我覺得學習 Python 令我覺得很自在。
P03	PEOU	我覺得可以容易適應學習 Python 。
P04	PEOU	我覺得學習 Python 時，總覺得時間過得很快。
P05	PEOU	我覺得學習 Python 不會讓我感到緊張
P06	PEOU	我覺得 Python IDLE 操作過程很簡單。
P07	PEOU	我覺得 Python 程式設計並不會很抽象。
P08	PEOU	我覺得 Python 程式設計很容易理解。
P09	PEOU	我覺得學習 Python 很輕鬆。
P10	PEOU	我覺得 Python 課程很有趣。
P11	PEOU	我覺得 Python IDLE 軟體的使用者介面環境很方便使用。
P12	AU	我在 Python 課程有認真學習。
P13	PU	我覺得學習 Python 可加強我對程式觀念的理解。
P14	PU	學習 Python 讓我覺得比較理解程式設計的運作流程。
P15	PEOU	學習 Python 程式設計讓我覺得學習程式設計很有挑戰性。
P16A	PEOU	給我題目，我就可以自己想出解題步驟(不需要老師提示)。
P16B	PEOU	給我題目，我就可以自己想出解題步驟(需要提示)。
P17	BI	我覺得學習 Python 可引發我學習程式設計的動機。
P18	PU	Python 的流程語法有助於理解要表達的是什麼。
P19	PU	操作 Python 過程中能夠幫助我將想法具體化。
P20	PU	我覺得完成 Python 作業很有成就感。
P21	BI	未來遇到需要設計程式時，我會繼續使用 Python 來解決問題。
P22	BI	我喜歡將我的 Python 作品分享出來，讓別人來觀摩。
P23	BI	我希望以後有機會可學習更多的程式設計課程。
P24	BI	整體而言，我同意 Python 程式設計課程的學習
P25	PEOU	學習 Python 程式設計後讓我覺得學習程式設計並沒有想像中的難
P26	PU	我覺得 Python 課程之學習可以培養我發現問題與解決問題的能力
P27	PU	我覺得 Python 課程之學習可以訓練我邏輯推演的能力
P28A	PEOU	我覺得 Python 證照考試的題庫很簡單，讓我不害怕學習程式設計

P28B	PU	我覺得 Python 證照考試的題庫有助於學習程式設計
P28C	PEOU	我覺得 Python 證照考試很簡單，讓我不害怕學習程式設計
P29	PU	我覺得 Python 課程範例的學習，有利於我學習程式設計
P30	PU	我覺得 Python 課程安排順序，可以幫助我的學習