

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 : PGE1122320

學門專案分類/Division : 通識(含體育)-通識課程

計畫年度 : 112 年度一年期 112 年度多年期

執行期間/Funding Period : 2023.08.01 - 2024.07.31

不同學院學生於各種程式設計能力學習成效之研究

通識初級程式設計-行動應用程式

計畫主持人(Principal Investigator) : 顏士淨

協同主持人(Co-Principal Investigator) : 無

執行機構及系所(Institution/Department/Program) : 國立東華大學資訊工程學系

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 7 月 31 日

# 不同學院學生於各種程式設計能力學習成效之研究

## 一. 本文 Content

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

主持人自 107 年開始在通識課程教授程式設計課程，至今每年約有六百多位學生修課，對象學生來自全校八大學院。我們發現程式設計課程對於非理工學院的學生是一項挑戰，用一般程式語言如 C/python 教學效果不佳，無法做到深入與完整的教學。學生也無法將上課所學到的東西學以致用，只為了拿到學分而修課，形成教育的浪費。因此必須有創新的教學方式才可以解決目前的問題，讓課程內容真正對學生有所幫助。由於通識課程學生來自不同學院，包括人文社會、理工、管理、教育、藝術、原住民等學院，各有不同背景，而如何評估研究不同學院學生在目前教學方法的學習成效，並藉以改進教學模式，就成為重要的課題，也是本計畫試圖解決之問題。

本計畫研究不同學院學生在通識程式設計課程之各種階段的學習成效，為了深入分析，根據申請人多年的教學經驗，將此通識課程之程式設計能力再拆解為四種：基本程式撰寫能力，邏輯迴圈流程控制能力，資料結構函數使用能力，進階物件運用能力。本計畫將研究這四種程式設計能力於各學院學生之學習效果。

### 2. 研究問題 Research Question

為了更深入分析學習成效，我們以一學年開設的四門課的 640 位修課學生，涵蓋全校各學院學生修習通識課程學生為研究對象，將本課程內容再細分為四種程式設計能力：基本程式撰寫能力，邏輯迴圈流程控制能力，資料結構函數使用能力，進階物件運用能力。希望能進一步了解各學院學生對各種程式設計能力的學習成效是否有差異。未來將可根據此研究成果，增加更多教材，並且設計不同評量方式或作業專題等，因材施教，引導最有效率的學習方式。

### 3. 文獻探討 Literature Review

學習領域可以分為認知領域、心理運動領域（技能）和情感領域（態度）。由於我們的通識程式設計課程是行動應用程式，可以在學生熟悉的手機上執行，提升學生學習興趣，因此我們以認知、技能和態度作為設計自我評估問卷的基礎。這分類方式是由 Benjamin Bloom 領導的一組研究人員在 1956 年制定的學習領域分類法 (Taxonomy of Learning Domains) [1]。學習領域最早是在 1956-1972 年間開發和描述的。一些參考文獻將所有域都歸於 Benjamin Bloom，雖然 Bloom 參與了對認知和情感領域的描述，但他作為第一作者出現在認知領域。因此，儘管他的同事大衛·克拉思沃爾 (David Krathwohl) 是 1956 年出版物的合作夥伴，但該書多年來一直以他的名字命名，並且在教育家中普遍稱為 Bloom 的分類法[2]。在本計劃中，由於我們教育目標包括讓學生可以使用程式設計技能來解決問題，因此我們採用 Bloom 分類法，將學習領域分為認知領域、心理運動領域（技能）和情感領域（態度）三個部分，並依此分類設計問卷。

我們相信，對大多數學生來說，尤其是非資訊本科系的學生，參加第一門程式設計課程被認為是困難的。依據 Ali[3] 等人的研究，對於剛接觸程式設計的學生，編寫程式語言的非常可能令人受挫，包括死板的語法、不熟悉的結構、開發程序的時間長短，並獨立工作等。Daly[4] 認為這些問題常常會導致學生容易犯錯誤的環境中學習抽象的程式語言概念，造成程式設計課程讓學生感到沮喪，失去信心，並避免參加程式課程。因此，Kaplan[5] 認為應該選擇程式語言，最大限度地降低程式設計的困難度，否則學生在學習程式時會因此受挫。

本門課是程式設計，雖然是通識課程，但是內容必須符合專業程式設計課程的架構。跟同等級的高階語言相比，基本功能均一樣具備，但是，不同的程式語言的認知難度不同，對於學生學習有著相對的影響。Ling[6] 等人的研究，程式語言的結構簡單，語法易解，可以讓學生自我效能感與學習動機更好，進而降低學生的挫折感，讓學生更有意願去學習此種程

式設計的技巧，應用此程式語言於日常生活中。然而，學生學習程式設計的困難是否單純是語言的問題呢？Anabela Gomes[7]等人卻認為解決問題的能力低是導致大量學生在他們的程式語言課程中失敗和沮喪的主要原因，傳統教學方法和學習方法並不是最適合許多學生的，如何增進學生解決問題的能力亦是計算機科學重要的發展方向。Konecki[8]認為學生在理解複雜的程式設計概念存在最大的問題是這些概念本質上是很抽象的，且直觀上並不清晰。學生在理解程式設計任務和設計合適的演算法方面比較困難，未來的課程應降低程式語言的複雜語法和概念，與演算法思維方式的訓練。我們在上一年計劃中，探討使用圖形化的教學方式，降低學習門檻。本年度計畫則是在深入探討不同背景學生在學習程式設計的學習成效。

#### 4. 教學設計與規劃 Teaching Planning

##### (1) 教學設計與規劃說明

###### 一、教學目標：

本課程內容主要是 Android 手機程式撰寫，撰寫程式的工具為 MIT App Inventor 2(AI2)。本課程內容分為四種程式設計能力：基本程式撰寫能力，邏輯迴圈流程控制能力，資料結構函數使用能力，進階物件運用能力。教學目標為使學生了解電腦程式的基本概念與應用，訓練邏輯思考與培養運算思維的能力，能夠運用抽象的空間思維，能夠獨立完成簡易手機應用程式，提高學生對於程式設計的興趣。

###### 二、課程計畫：

本課程為線上課程搭配實體課，請按順序自行到[東華 e 學苑](#)下載與閱讀線上課程。

上課活動包括：

自行閱讀線上課程與演練範例。

[東華 e 學苑](#)線上測驗，每章節有一次。

期中考與期末考：上機考試，每人都必須參加期中考與期末考。

線上講解：提供電腦教室詢問問題與閱讀線上課程，自由參加。

[FB 社群線上提問](#)：“NDHU AI2 通識程式設計”平時有疑問可以到 FB 社群線上提問，TA 將會儘快回覆。

● 小考：考試範圍下表括弧內容，每次範圍包括三周，只考一題。

週次	上課內容	活動	備註
1	課程簡介	課堂講授教學	基本程式撰寫能力
2	AI2 程式環境設置	線上講解	
3	AI2 程式介面設計	線上講解	
4	資料型態與運算	第一次小考	
5	邏輯判斷流程控制(單雙向)	線上講解	邏輯迴圈流程控制能力
6	複雜邏輯判斷流程(多重與巢狀判斷)	線上講解	
7	迴圈流程控制(計數與條件迴圈)	線上講解	
8	巢狀迴圈控制	第二次小考	
9	期中考試第一梯次(25%)	實體上機考試	
10	期中考試第二梯次(25%)	實體上機考試	
11	陣列清單資料結構	線上講解	資料結構函數使用能力
12	程序函式&多重頁面	線上講解	
13	內建函數應用	第三次小考	

14	多媒體	線上講解	進階物件運用能力
15	繪圖動畫	線上講解	
16	資料庫	第四次小考	
17	期末考試第一梯次(25%)	實體上機考試	
18	期末考試第二梯次(25%)	實體上機考試	

### 三、教學方法：

課堂講授、討論，實際上機操作，自行到東華 e 學苑下載與閱讀線上課程。

### 四、評量標準：

- 期中考/期末考：實體上機考試，每次三題，兩次共佔總分50%。每次考試有一題基本題，其餘兩題分別對應一種程式設計能力。
- 小考：每次一題(佔總分7%)，每次內容對應一種程式設計能力，共四次佔總分28%。
- 東華 e 學苑線上測驗(包括前測/後測/各種問卷等)：佔總分12%。
- 專案：佔總分 10%。

## 5. 研究設計與執行方法 Research Methodology

### (一) 研究問題：

研究不同學院學生在通識程式設計課程之各種階段的學習成效，計畫的課程之程式設計能力可分為四種：

1. 基本程式撰寫能力
2. 邏輯迴圈流程控制能力
3. 資料結構函數使用能力
4. 進階物件運用能力

修課的學生來自東華大學八大學院，各有不同背景，本計畫將研究這四種程式設計能力於各學院學生之學習效果。

### (二) 研究範圍：

針對東華大學各學院修習通識課程學生，在計畫設計的教材與教學方式對學習四種程式設計能力的成效為研究的範圍。

### (三) 研究對象與場域：

以每年 600 位的東華大學各學院修習通識課程學生為研究對象，每學年度上下學期各兩班，每班人數 160 人，共計 640 位學生。本計劃報告包括 111 學年度和 112 學年度的資料，共計 1289 位學生，42 個科系，8 個學院，表一為學院名稱、編號和修本課程的學生數量，圖 1 為各院學生分布圖。場域為教室、電腦教室、東華 e 學苑。

No.	College name	中文簡稱	Number of students
1	Humanities and Social Science	人文	436
2	Environments and Oceanography	環境學院	25
3	Arts	藝術	128
4	Huashi Education	花師	261
5	Science and Engineering	理工	59
6	Indigenous Studies	原民院	146
7	Management	管理學院	219
8	Common Education	通識	15

表 1 學院名稱、編號和修本課程的學生數量

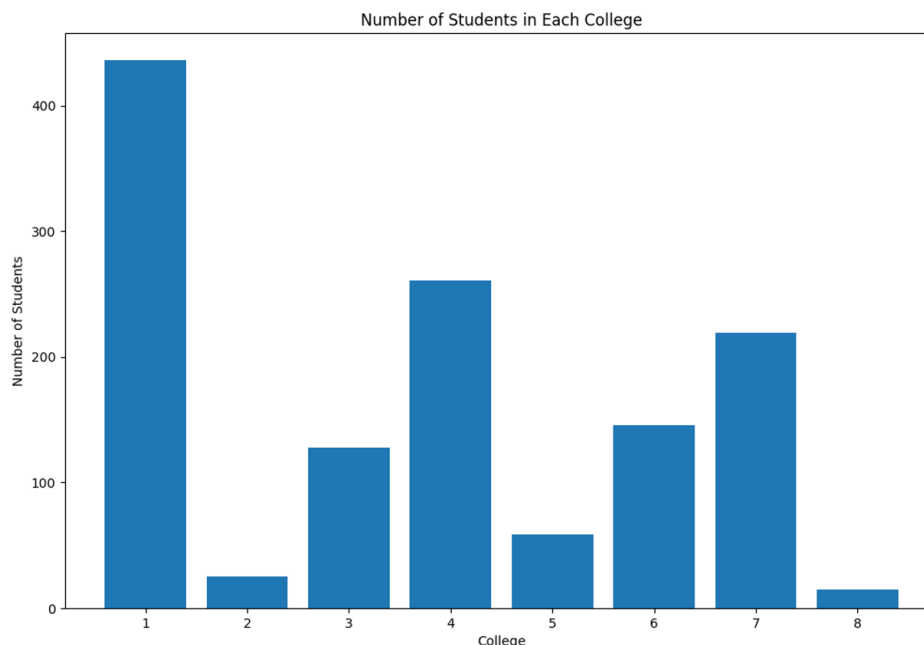


圖 1 各院學生分布圖

#### (四) 研究方法與工具：

評量工具包括考試與作業、前後測、自我評估問卷、學生訪談等等。考試與作業內容對應四種程式設計能力，詳細如下：

- 期中考/期末考：實體上機考試，每次三題，兩次共佔總分 50%。每次考試有一題基本題，其餘兩題分別對應一種程式設計能力。
- 小考：每次一題(佔總分 7%)，每次內容對應一種程式設計能力，共四次佔總分 28%。
- 東華 e 學苑線上測驗(包括前測/後測/各種問卷等)：佔總分 12%，內容對應到各種程式設計能力。
- 專案：佔總分 10%。

前後測為課程相關能力測驗，於學期初與學期末讓學生填寫。問卷中學生以程式設計課程的認知、技能與情意三方面對自我的評鑑。其資料分別以使用描述性統計、t-test、F-test 和單向 ANCOVA 分析數據，以期了解學生主觀的認知，技術與情意是否相關，再以學生的客觀四個階段學習成績與其主觀的自我評鑑做比較，測試其是否相關，並且評估其學習成效。

問卷共有 9 題，內容如下：

##### 認知

1. 我認為自己對於寫程式基本相關知識已經有初步的了解。
2. 我會想要吸收更多有關寫程式的知識。
3. 我能運用程式設計的相關知識於生活或學業中。

##### 技能

4. 我對如何寫程式感到很有興趣。
5. 我已經會寫個簡單程式。
6. 我會在日常生活或課業中寫個簡單的程式。

##### 情意

7. 我有意願去接觸能更多程式設計的課程。

8. 我希望能有機會參與有關程式設計的活動。
9. 我期盼自己能在畢業專題中多使用一些自己寫的程式。

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

### (1) 教學過程與成果

本課程建立一個完整的圖形化程式設計線上教材，共 12 單元，期能引導非本科系學生學習的興趣。各單元均附有線上測驗題目，能提供學生學習基本的程式設計概念與技能，在課程的前期與後期分別提供學生對於程式設計的認知、技能與情意的自我評估問卷，供教師教學改進之參考，全部的課程由淺入深，學理與應用兼顧，若學生確實學習，認真完成作業，不僅能獲得程式設計的基本概念，更可以應用此技術在其日常生活之中，發揮學生學習的智慧。

本課程內容主要是 Android 手機程式撰寫，撰寫程式的工具為美國麻省理工學院(MIT)所發展的 App Inventor 2(AI2)。課程設計包含電腦程式設計的各種基本需求，包括資料型態與計算，邏輯判斷，迴圈控制，陣列使用，程序函式等概念。本計畫將這些需求分為基本程式撰寫能力，邏輯迴圈流程控制能力，資料結構函數使用能力，進階物件運用能力。

教學大量引進各種圖片解說程式語言概念，如圖 2。搭配積木式的程式撰寫方式，如圖 3，適合非資訊相關領域的學生修讀。教學目標為使學生了解電腦程式的基本概念與應用，訓練邏輯思考與培養運算思維的能力，提高學生對於程式設計的興趣，能夠獨立完成簡易手機應用程式。

研究實施程序如下：

- 學期初先請學生填寫研究參與者知情同意書，如附件
- 對於計畫的課程之程式設計能力分為四種能力，相關教材針對這些能力做配置
- 設計前後測、於學期初與學期末實施測試
- 於四個程式設計能力課程結束時，讓學生填寫自我評估問卷內容
- 四次小考分別針對四個程式設計能力設計基本題目
- 期中期末各有四題針對四個程式設計能力出進階題目
- 分別以使用描述性統計、t-test、F-test 和單向 ANCOVA 分析數據，以期了解學生主觀的認知，技術與情意是否相關
- 再以學生的客觀學習成績與其主觀的自我評鑑做比較，測試其是否相關，並且評估其學習成效

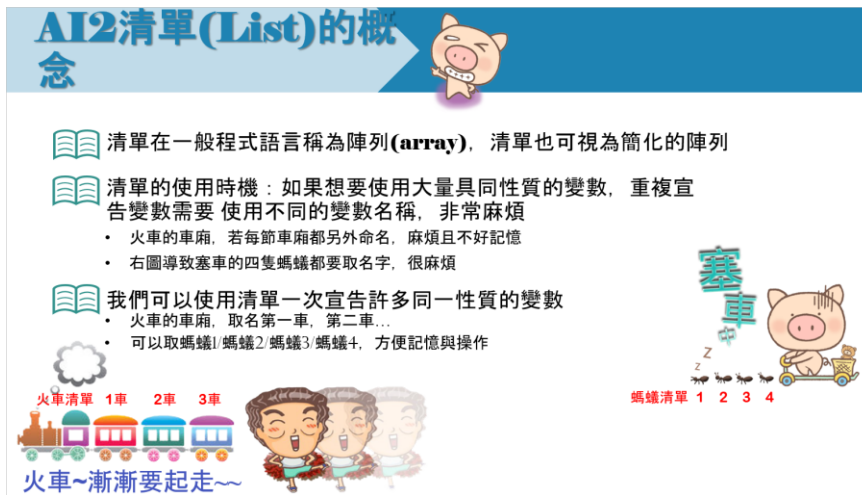


圖 2 使用大量圖片解說程式語言概念



圖 3 積木式的程式撰寫方式

## (2) 教師教學反思

計畫主持人已經有六年以上的通識程式設計課程開設經驗，每年課程教材持續不斷改進。兩年前加入英文教材，本校外籍生也可修讀此課程取得資訊能力證明，預計會繼續每學期開設此課程。本課程採用實體與線上偕同教學，提供大量線上教材，學生可依據自身不同程度閱讀線上教材學習，期望達成因材施教的目的。成績評量方式，主要是以實體上機考試，由教師與助教親自當面對每一位學生批改其撰寫的程式，確保能正確評估每一位學生的學習狀況。

修課學生反應大致良好，停休比例也從一開始的 25% 下降至 5%，並且評量分數也逐步上升。也有很多學生反映修課之後，對資訊科技產生興趣，從而改換跑道，在研究所或是出國深造時，選擇資訊科技領域。也有藝術學院學生以此課程所學為基礎，申請並獲得青年創業獎勵。本課程為通識課程，以大班的方式上課，無法直接照顧到所有學生，必須應用資訊科技的協助，讓學生完成學習程式設計的目的。

本研究分析八個學院的 42 個科系，總共 1289 位學生，結果顯示以學院為單位，學生在四種學習能力上有顯著差異，重視數學的科系，學習能力較佳。而不同年級之前也有差異，一年級生的能力最佳，顯示程式設計能力與高中成績有正相關。因此可以鼓勵學生在大一就開始修通識程式設計課程。

## (3) 學生學習回饋



圖 4 為各學院學生的程式設計學習能力，Part1 是基本程式撰寫能力，Part2 是邏輯迴圈流程控制能力，Part3 是資料結構函數使用能力，Part4 是進階物件運用能力，學院編號對照請參考表一。其中管理學院(7) 學習能力分數最高，該學院最多的修課學生來自觀遊系 (30.13%)與企業管理系(20.09%)。

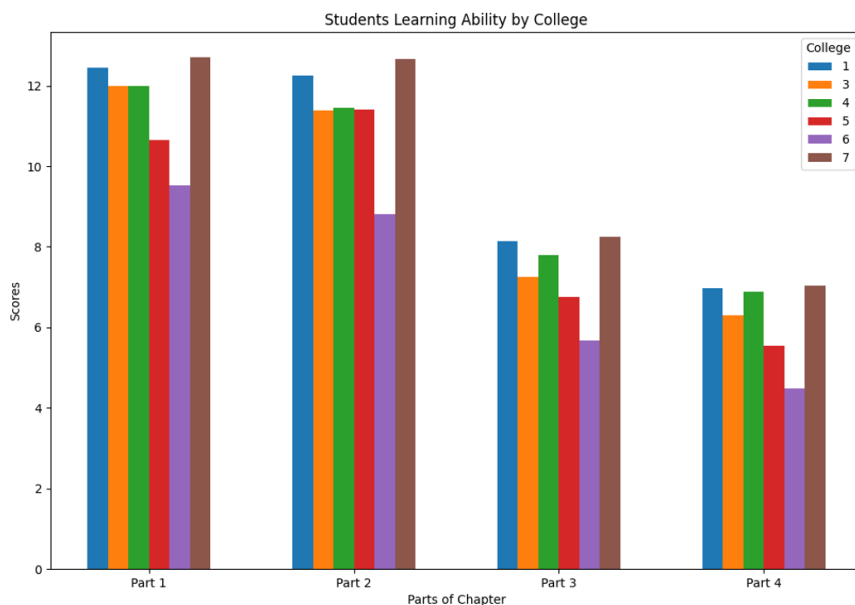


圖 4 各學院學生的程式設計學習能力

由於各學院科系可能有差異，以學院來區分不見得好用，以科系區分則是太細。我們嘗試將各科系的所有學生過去五年修課成績互相比較，使用 Kmeans 演算法來進行分群，演算法的輸入為這些學生的 GPA 的機率分布，輸出為分群結果。分群個數嘗試值為 3, 5, 7，值為 7 時會有最好表現，可以將每群數量控制在三個到五個科系之間。使用的距離函數為歐幾里得(Euclidean distance)距離，最後的誤差值 SSE 為 4487.82。

計算出新的分類(群)如下：

- 第 0 群：旅遊、藝術與設計、原住民、歷史。
- 第 1 群：管理、財務金融、諮商心理、化學、材料科學。
- 第 2 群：幼教、法律、國貿。
- 第 3 群：藝術與文學、音樂、體育。
- 第 4 群：中文、社會學、台文。
- 第 5 群：英語語言文學、經濟、教育。
- 第 6 群：理工學院。

圖 5 為新的分類(群)學生的程式設計學習能力，其中第 1 群的學習能力最為突出，除了第一種能力**基本程式撰寫能力**之外，其餘三種能力都是最高。而第 3 群則是在各項能力中落後，觀察原始數據後發現，主要差別在於過去五年，第 1 群獲得 A 等級分的比例大幅超越第 3 群。而第 5 群在第 4 種程式設計學習能力(進階物件運用能力)較為突出，可能跟科系屬性有關。

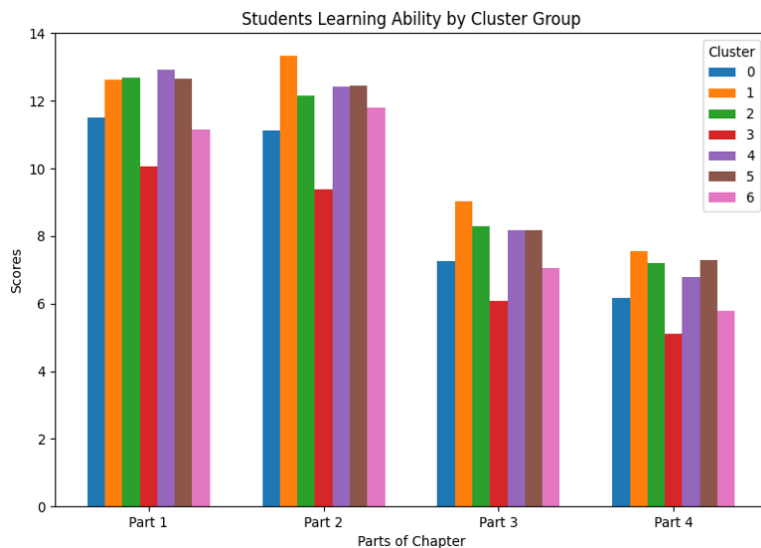


圖 5 新的分類(群) 學生的程式設計學習能力

圖 6 顯示大一生與其他年級的程式設計學習能力，可以看出大一生在各項能力要比其他年級要高，由於本門課是通識課程，而且有開程式設計的科系之學生不會修本課程，因此程式設計並不是修課學生的專業科目，越早修讀本課程對學生越有利。

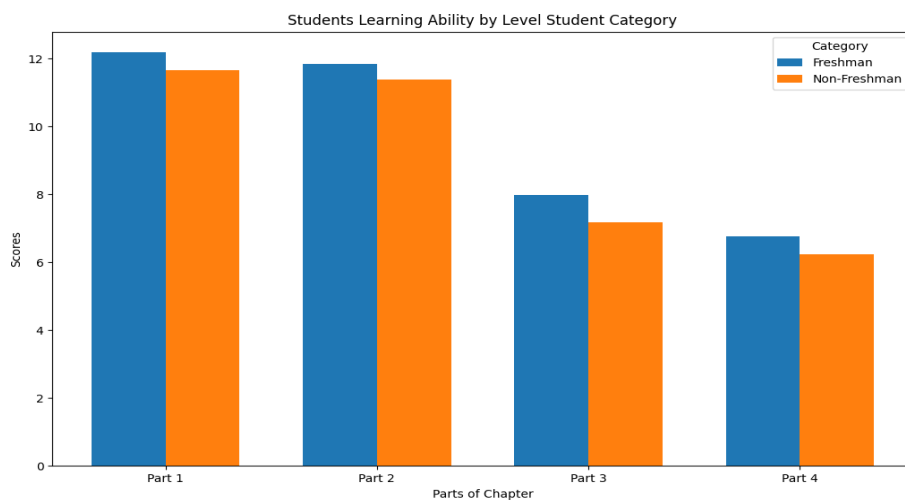


圖 6 大一生與其他年級的程式設計學習能力

## 7. 建議與省思 Recommendations and Reflections

本研究分析了 1289 位學生，來自東華大學 8 個學院 42 個科系，其中剔除兩個修課人數較少的學院：海洋學院與洄瀾學院。分析的程式設計學習能力包括基本程式撰寫能力，邏輯迴圈流程控制能力，資料結構函數使用能力，進階物件運用能力。分析的結果顯示管理學院的學生具有較佳的程式設計學習能力，其中包括觀遊系(30%)與企管系(20%)的學生。

由於同一學院科系屬性可能差距過大，我們也以 GPA 機率分布為基礎來做新的科系分群，可以得到新的分群結果，此結果有助於我們了解學生的學習能力，將來可以給予適當的協助，也可以更正確預測學生的學習曲線，做到更正確的期中預警。

另外本課程每學期開設兩門，一門是給大一優先選課，另一門則是高年級優先選課，而本研究也分析了大一生與其他年級的程式設計學習能力，並且建議最好的程式設計學習階段是大一。

本計畫內容已經投稿 2024 TAAI 人工智慧與應用會議論文，本課程的簡化版已經在線上教學平台 ewant 育網平台上架，如下頁圖 3。並且廣泛開課，作為大學先修課程，每年有數以千計的學生上網修讀此課程，我們也將參考此教育實踐計畫成果，繼續改進此課程。





圖5. 2024年本課程有44門開設給各高中



## 二. 參考文獻 References

1. Bloom, B. S., Engelhart, M. D., Furst, E. J., Hill, W. H., & Krathwohl, D. R. (1956). Taxonomy of educational objectives: The classification of educational goals. Handbook 1: Cognitive domain. New York: David McKay.
2. Hoque1, M. Enamul. (2016). Three Domains of Learning: Cognitive, Affective and Psychomotor, The Journal of EFL Education and Research, Vol. 2, No. 2, September 2016: ISSN-2520-5897.
3. Ali, A., and Smith, D. (2014). Teaching an introductory programming language in a general education course. Journal of Information Technology Education: Innovations in Practice, 13, pp. 57-67.
4. Daly, T. (2011). Minimizing to maximize: An initial attempt at teaching introductory programming using Alice. Journal of Computer Science in Colleges, 26(5), pp. 23-30.
5. Randy M. Kaplan. (2010) Choosing a first programming language; SIGITE '10: Proceedings of the 2010 ACM conference on Information technology education October 2010 Pages pp. 163 - 164.
6. Ling, H.-C., Hsiao, K.-L., and Hsu, W.-C.. (2021). Can Students' Computer Programming Learning Motivation and Effectiveness Be Enhanced by Learning Python Language? A Multi-Group Analysis; Front Psychol. Jan 21, 2021.
7. Gomes, A., Mendes, A. J. (2007). An environment to improve programming education; Proceedings of the 2007 International Conference on Computer Systems and Technologies, CompSysTech 2007, Rousse, Bulgaria, June 14-15, 2007.
8. Konecki, M. (2014). Problems in programming education and means of their improvement. DAAAM International Scientific Book. 2014, pp. 459-470.