

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：PHA107008

學門分類：人文藝術及設計

執行期間：2018/8/01~2019/7/31

探索程式設計遊戲於技職大學生之遊戲程式設計課程學習成效/
Exploring the effect of coding games on vocational-technical college students' learning
performance in game programming courses
資料結構/ Data Structure

計畫主持人：黃永銘

共同主持人：

執行機構及系所：南臺科技大學多媒體與電腦娛樂科學系

繳交報告日期：108/08/15

目錄

摘要.....	ii
Abstract.....	iii
一、 報告內文.....	1
1. 研究動機與目的.....	1
2. 文獻探討.....	2
3. 研究方法.....	3
4. 教學暨研究成果.....	5
二、 參考文獻.....	11
三、 附件.....	13

摘要

近年來，程式設計師已經被視為是發展數位遊戲產業的重要人才。然而，程式設計人才的培育是相當困難的事情，因為傳統的教學模式無法引起學生的學習動機。除此之外，教師無法針對不同程度的學生來提供個別化的指導，因而使得學生的學習成效低落。幸運地，數位遊戲式學習已經被證實有助於提升學生的學習動機，並且改善他們的學習成效。因此，本計畫預計採用程式設計遊戲(coding games)於遊戲程式設計課程之中，並搭配行動研究法來輔助學生學習程式設計。研究結果顯示，學生對於這類教學方式是可以接受的，其中大部分的學生認為它有助於提升學習動機。但是，教師在實驗活動中發現學生很少使用這類遊戲進行練習，而教師推測是兩個原因的影響。第一，教師受限於課程進度壓力，所以只能安排學生在課後使用這類遊戲進行練習。第二，這類遊戲的訓練方式與實際撰寫程式碼之間存在落差，所以學生可能不信任這類遊戲的學習效果。

關鍵字：程式設計、程式設計遊戲、技職大學生、學習成效

Abstract

In recent years, the programmer was considered as important talent for the development of digital game industry. However, the cultivation of programmer talents is difficult, because the traditional learning mode can not arouse students' learning motivation. In addition, teacher can not provide individualized guidance to students of different levels, which makes student' learning low. Fortunately, digital game-based learning has been confirmed by researchers that it has a positive impact on students' learning achievement because their learning motivation can be promoted when they engage in such learning environments. Therefore, this project is expected to use coding games in game programming courses and action research methods to assist students in learning programming. The results show that this kind of teaching method is acceptable to students, and most of them think it helps to improve their learning motivation. However, teacher found that students rarely use such game to practice in the experimental activities, the teachers speculated that there were two reasons. First, teachers are limited by the progress of the curriculum pressure, so they can only arrange students to use such games for practice after class. Second, there is a gap between the training methods of such games and the actual writing code, so students may not trust the learning effects of such games.

Keyword: Programming, Coding Games, Vocational-Technical College Students, Learning Achievement

探索程式設計遊戲於技職大學生之遊戲程式設計課程學習成效

一、報告內文

1. 研究動機與目的

隨著數位遊戲產業的蓬勃發展，程式設計人才的需求也隨之大幅增加，而主要原因在於程式設計在遊戲設計中扮演串連每個環節的重要角色。具體來說，數位遊戲是透過許多不同專業人才互相合作所開發出的電腦軟體，其中這些人才包含遊戲企劃人才、遊戲美術人才、程式設計人才、音樂音效人才以及遊戲測試人員等相關人才(葉思義, 2010)，而在這些人才之中，程式設計人才被視為最重要的關鍵角色。因為程式設計人才是負責將遊戲企劃人才的提案實作成遊戲系統，其中遊戲系統不只包含玩法操作與運行規則，同時也涵蓋美術風格、動畫特效以及音樂音效的呈現方式。由此可知，程式設計人才在數位遊戲產業的關鍵性，以及培育程式設計人才對於數位遊戲產業的重要性。

然而，學習程式設計對於學生而言是一件困難的事情，因為傳統的教學模式會使學生面臨許多學習困境，例如教師無法給予學生個別化學習、學生被動學習無法內化知識以及學生在家練習時缺乏適時輔助等問題，而這些學習困境容易影響學生在學習過程中失去動機並放棄學習(Kelleher & Pausch, 2005)。關於教師無法給予學生個別化學習，原因是教師需要面對多位學生進行授課，並考慮到課程進度，因此學生有學習上的問題時，教師無法針對個別學生進行指導。關於學生被動學習無法內化知識，原因是傳統的教學模式是由教師單方向傳遞知識給學生，而學生扮演的角色是被動的學習者，其中他們不一定能將這些知識轉化成自己的能力(邱文傑, 2015)。關於學生在家練習時缺乏適時輔助，原因是學生在家練習時，他們可能會忘記課程學到的相關知識，並且因為沒有輔助練習的方法，所以他們容易採用錯誤的方式嘗試解決問題，而這些錯誤方式無法幫助學生訓練解決問題的能力(Linn & Clancy, 1992)。綜合上述，目前的程式設計課程會讓學生面臨許多學習困境，使得學生的學習動機低落，進而導致他們放棄學習程式設計語法。

另外，同儕能力的差異也會影響技職大學生的學習狀況，而這也是目前技職大學需要解決的關鍵問題。具體來說，目前技職大學的學生來源可分為高中生與高職生，其中高職生更包含不同學科的學生，例如電子科、機械科、資訊科、美工科等學科。換言之，即使同樣是技職大學生，有些學生可能已經接觸過程式設計課程，而有些學生則尚未接觸，其中這種學生程度的差異將會影響到他們在程式設計課程的學習狀態。舉例來說，目前的程式設計課程是由教師統一進行授課，而有程式基礎的學生可能覺得練習太簡單而無法獲得有效學習，沒有程式基礎的學生則可能認為練習太過困難而放棄學習。因此，不論是哪一種情況，學生都無法發展程式設計專長，進而成為資訊科技產業需要的人才。根據教育部的技職教育白皮書中指出，技職大學體系的教育目標是培育專業的技術人才(教育部, 2002)。而在現今社會，資訊科技的發展使得相關產業對於專業技術人才的需求與日俱增，連帶影響程式設計成為技職大學等技職教育體系所重視的領域(賴怡君, 2008)。然而，技職大學面臨的問題將使它們無法有效培育專業技術人才，進而導致我國相關產業的競爭力低落，因此上述問題有儘速解決的必要性。

綜合上述，本計畫的目的是協助技職大學生於程式設計課程使用程式設計遊戲來學習程式設計語法，其中本計畫預計透過程式設計遊戲具備的遊戲式學習與個人化練習來幫助學生避免遭遇上述學習困境，如圖 1 所示。具體來說，培育程式設計人才對於數位遊戲產業的發

展是一件重要的事情，但目前技職大學在培育學生成為程式設計人才時，學生在學習過程中會遭遇許多學習困境的挑戰。因此，本計畫嘗試導入程式設計遊戲於這類課程之中，藉此輔助學生學習程式設計，進而提升數位遊戲產業發展與競爭力。更重要的，這類遊戲除了提供學生使用遊戲的方式來學習程式之外，它更提供一個個人化練習的機會，其中每位學生可以按照自己的程度來挑戰遊戲中不同的關卡，藉此實現每位學生的個人化學習與練習。因此，本計畫的主要教學實踐貢獻是應用數位遊戲於程式設計課程之中，並進一步調查其成效，故本計畫之成果將可提供給相關領域教師參考，藉此激發出數位遊戲式學習之應用與發展。



圖1. 研究目的

2. 文獻探討

數位教育遊戲的特色在於它包含教育的元素，其中學生可以透過遊玩遊戲來學習相關知識。具體來說，學生使用數位教育遊戲進行學習時，遊戲中的聲光效果能夠引起學生的好奇心與愉悅感，藉此讓學生較容易沉浸於學習環境中，進而達到寓教於樂的效果(Prensky, 2007)。另外，陳昇介(2010)曾指出數位教育遊戲導入於教學活動中有四大優點。第一，數位教育遊戲在學生進行學習活動時，它可以有效提升學生的學習動機，因此學生使用這類遊戲進行學習活動的願意更高。第二，數位教育遊戲包含一個多樣化的學習環境，其中學生的學習活動可以透過學習環境的輔助變得較為多元化。第三，數位教育遊戲可以被設計成具有建構式與情境式的學習環境，其中學生透過環境的輔助可以進行建構式學習。第四，數位教育遊戲可以提供一個互動式的學習環境，其中學生在遊戲中不僅可以學習到遊戲內包含的教育知識，同時他們也可以學習到如何與其他玩家合作的技能。

有鑑於數位教育遊戲的特色，許多研究者開始探討其影響學生學習動機的原因，其中研究者發現有四個原因可以引起學生的學習動機，分別有挑戰性、好奇心、幻想性和控制性等四個原因(Prensky, 2007; Rieber, 1996)。關於挑戰性，它是遊戲中的一項基本要素，其中玩家在遊戲中面對不同的挑戰時，當挑戰的程度符合玩家本身程度時，玩家會開始主動思考如何解決挑戰。關於好奇心，玩家在遊戲世界中無法預測會有什麼事情發生，因此這種對於虛擬環境的未知感能夠激發玩家的好奇心，進而驅使玩家主動在遊戲世界中進行探索。關於幻想性，數位遊戲本身即是一個虛擬世界，其中玩家在虛擬世界中可以發揮他們的想像力與好奇心來探索各種未知事物。關於控制性，玩家在遊戲中擁有控制許多事物的控制權，其中包含角色的攻擊、移動和對話內容等事項。綜合上述，在遊戲世界中可以讓玩家體驗到挑戰性、好奇心、幻想性以及控制性等不同感受，進而使玩家持續沉浸在遊戲世界，而在這個過程中他們會不斷學習知識並提升他們的學習成效(Hainey, Westera, Connolly, Boyle, Baxter, Beeby, & Soflano, 2013; Sampayo-Vargas, Cope, He, & Byrne, 2013)。

由於數位教育遊戲具有引起學生學習動機的特性，目前已有許多研究者嘗試在程式設計課程中導入數位遊戲式學習，其中這類遊戲包含程式設計相關知識，因此學生在遊玩遊戲的

過程中同時可以學習到程式設計的相關知識(Amenyo, 2012; Corral, Balcells, Estévez, Moreno, & Ramos, 2014; Moreno, 2012; Wang & Chen, 2010)。具體來說，Wang & Chen (2010)在數位教育遊戲中融入程式設計的基本概念，例如常數與變數的宣告，其中學生可以透過遊玩遊戲學習這方面的知識。類似地，Moreno (2012)在研究中透過這類遊戲指導學生學習程式設計的相關知識，其中學生可以在遊玩遊戲的過程中學習到排序與疊代等程式概念。類似地，Corral et al. (2014) 則是在程式設計課程中結合電子積木與數位遊戲，其中學生藉由遊戲學習物件導向程式設計的相關知識。類似地，Mathrani, Christian, & Ponder-Sutton (2016) 在程式設計課程中導入一款網頁遊戲，其中學生可以透過遊玩遊戲學習到執行、偵測、迴圈以及條件判斷等程式概念。綜合上述，許多研究者發現數位遊戲式學習的潛力與效果，因此研究者們紛紛在程式設計課程中導入數位教育遊戲進行教學，同時探討數位遊戲式學習對於學生學習成效的影響。

3. 研究方法

(1). 實驗場域

本計畫預計採用資料結構課程作為實驗場域，其中資料結構屬於進階的程式教育課程，同時它也是電腦科學的基礎課程之一。具體來說，資料結構是電腦科學的基礎課程，其他它探討電腦中資料存儲與組織的方式，而學生如果能熟練使用程式語言則可以有效運用資料結構的概念來提升程式的執行效率(Michalewicz & Hartley, 1996)。換言之，當學生更熟悉資料結構的使用方式時，則電腦便可以更快速的提供我們想要得到的資訊(蔡明志，2011；謝樹明，2013)。由此可知，資料結構是程式教育中相當重要的基礎課程，因此本計畫嘗試使用資料結構作課程作為計畫的實驗場域。

(2). 研究對象

本計畫的研究對象是南臺科技大學多媒體與電腦娛樂科學系的學生，其中該科系一年級約 80 位學生分成兩個班級，並且該科系一年級皆有資料結構課程。因此，本計畫預計將以該科系一年級學生作為此次的研究對象。

(3). 研究架構

圖 1 是本計畫的研究架構，其中此架構包含本計畫的研究目的與之後預計實施的行動研究法流程。具體來說，主持人長年深耕於遊戲設計的教學研究中，其中主持人在教學現場發現目前學生在遊戲設計課程中遭遇的學習困境大多屬於程式設計方面的問題，因為學生認為程式設計課程過於艱澀難懂，進而使得學生的學習成效低落。因此，本計畫在確認研究問題後，開始蒐集相關文獻並著手分析處理資料，進而擬定出本計畫的研究目的，即程式設計課程透過融入「程式設計遊戲」於練習中，藉此改善學生學習程式設計的學習成效。之後，本計畫預計實施的行動研究法流程，其中包含計畫、行動、觀察、省思等四個階段。計畫階段主要是擬定未來行動計畫，其中包含搜尋適合用於課程教學之數位程式設計教學遊戲、確認課程內容與大綱以及安排每週上課進度等課程教學準備相關工作。行動階段主要是實施行動計畫，其中包含進行課程教學、協助學生使用程式教育遊戲進行練習等課程教學實施相關工作。觀察階段主要是蒐集與分析學生在教學過程中的回饋，其中包含評估學生課堂表現、學生平時作業水準等學習成效評估相關工作。省思階段主要是依據分析結果來反思課程實施上的問題，其中包含課程教學方式、課程進度安排以及課程教學工具等學習成效反思相關工作。然後，教師會依據觀察與省思階段所得到的成果來修正計畫執行方式，並在後續課程中實

施行行動研究法的第二次循環。總的來說，本計畫預計將此研究架構實施在程式設計課程中，其中本計畫藉由研究目的與行動研究法的搭配來探討學生在程式設計課程中的學習成效是否有所提升。

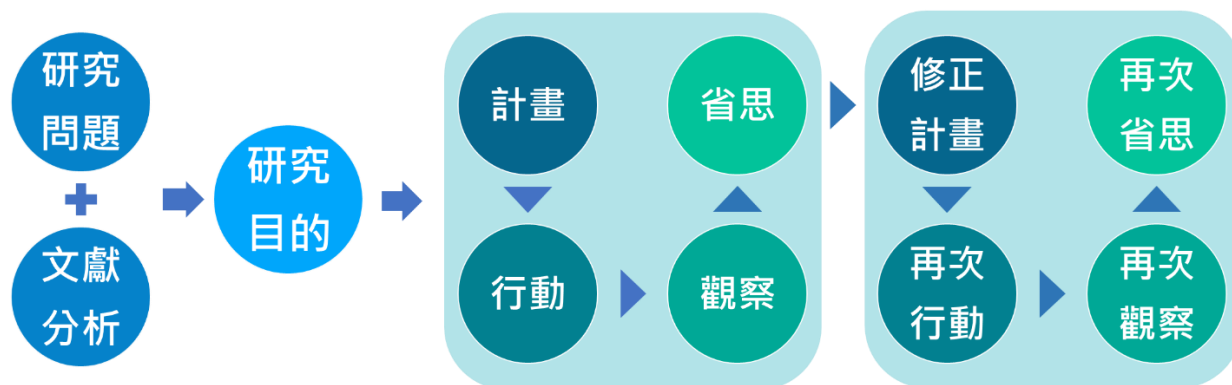


圖1. 研究架構

(4). 課程設計

圖 2 是本計畫的課程設計，其中本計畫預計在第 10 週正式開始實施教學研究。具體來說，本計畫規劃 1 至 8 週採用傳統講述的教學模式，而教學主軸在於培育學生具有基礎的資料結構概念，然後平時作業的繳交是評量學生成績的主要方式。而第 9 週的期中考則是評估學生學習資料結構概念的進展。接著，本計畫從第 10 週開始正式實施教學研究，其中本計畫預計每週實施一次行動研究法，而一共實施 8 週。其中 10 至 12 週主要是進行業師協同授課，在授課過程中業師除了說明資料結構的相關知識外，同時業師也會分享業界案例來幫助學生瞭解課程中的知識與未來就業或創業的相關性。然後，本計畫在第 13 至 17 週主要是導入程式設計教學遊戲來輔助學生進行個人學習，其中本計畫安排學生在課後使用這類遊戲進行練習，讓學生透過遊玩這類遊戲來提升程式設計能力。之後，本計畫在第 17 週安排一個實驗活動來調查學生對於此教學方式的滿意度，其中此實驗活動採用自願報名，並在課後時間進行。此外，在這段時間內，教師會從旁觀察學生的學習情況來調整計畫實施的方式。最後，在第 18 週的期末考，其中教師透過學生期中期末測驗的學習成效變化與過往學生相比較獲得研究成果。



圖2. 課程設計

(5). 研究工具

圖 3 是 CodeCombat 的遊戲畫面，其中它是一款角色扮演的網頁遊戲，而玩家需要撰寫程式語法來控制角色完成關卡目標。具體來說，玩家可以透過撰寫程式語法來控制遊戲角色進行各種動作，當遊戲目標要求玩家攻擊敵人時，他們必須撰寫適合的程式碼來控制角色移動以及攻擊敵人。另外，CodeCombat 同時支援多人對戰模式，玩家可以透過網路與其他玩家合作完成任務或互相對抗，進而感受到競爭的樂趣性。而 CodeCombat 提供多種程式語言給玩家學習，例如 JavaScript、Python、Lua、CoffeeScript 等程式語言，其中玩家在進行遊戲時，

系統會提供相關語法的教學給玩家，讓玩家可以順利過關。因此 CodeCombat 是一款適合初學者使用的程式教學遊戲。

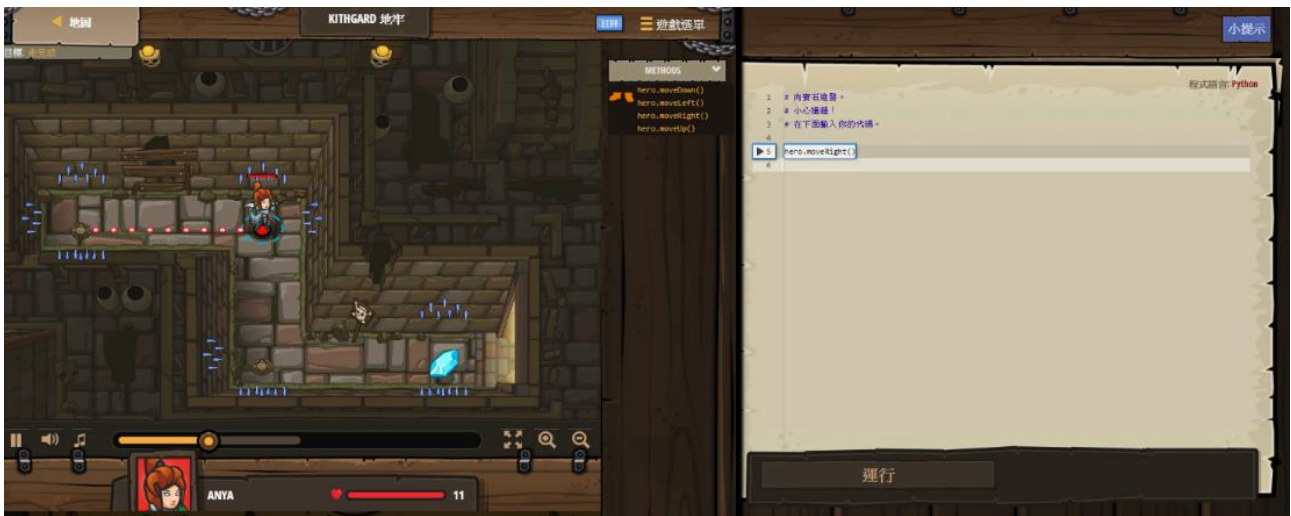


圖3. CodeCombat

本計畫的調查問卷，其中此問卷包含 5 個構面一共 23 個題目來瞭解學生對於這類學習方式的滿意度，參考附錄 1。評分方式以李克特五點量表進行評分，其中非常不同意代表 1 分，不同意、普通、同意、非常同意則依序代表 2 至 5 分。另一方面，問卷最後一題為開放式問題讓學生可以表達個人意見。

(6). 分析方法

本計畫預計使用不同統計方法來分析研究成果，如 Cronbach's α 、建構效度、滿意度分析和皮爾森相關分析等統計方法。Cronbach's α 主要是測量問卷的可信度，其中它是透過重複測量來檢驗問卷結果的一致性。建構效度主要是測量問卷是否可以真實反映問題的程度。滿意度分析主要是了解參與者對構面的看法，其中它是透過構面題項的平均數變化來了解參與者的看法。皮爾森相關分析主要是測量兩個連續變數之間的相關，如果兩變數之間的相關係數較大，則表示兩變數互相影響的程度較大。因此，本計畫藉由不同的統計方法來了解程式設計遊戲對於技職大學生在程式設計學習活動中的學習影響。

4. 教學暨研究成果

(1). 教學過程

本計畫在第 10 週正式開始實施教學研究，而第 10 週至 12 週是由業師與教師進行資料結構課程的協同授課，如圖 4 所示。具體來說，本計畫安排業師負責的授課內容是資料結構中的搜尋和樹狀結構等概念，其中業師在授課過程中會搭配業界案例來向學生說明資料結構概念如何被使用於業界專案上，藉此讓學生了解如何活用課堂知識。此外，業師在授課過程中，如果出現同學可能尚未接觸過的知識，則教師會針對這部分進行補充說明，讓學生可以更容易了解業師的說明。總的來說，本計畫安排業師與教師進行協同授課，藉由教學上的互補，讓學生可以更加了解課堂知識以及真正活用知識。

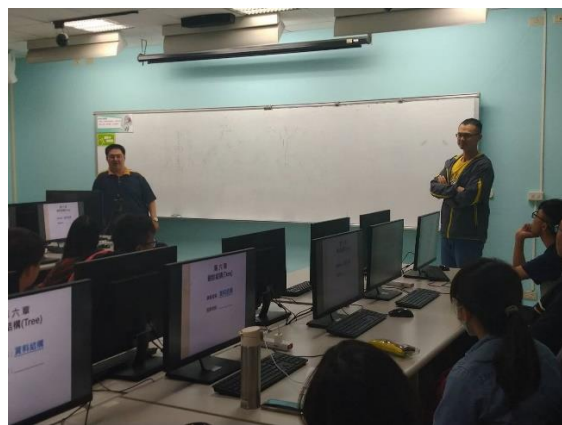


圖4. 業師與教師協同授課

本計畫在第 13 週至 17 週是導入程式設計教學遊戲來輔助學生進行個人學習，如圖 5 所示。首先，教師會在課堂上介紹 CodeCombat 這款遊戲以及它的操作方式，讓學生了解如何使用 CodeCombat，然後教師會請學生在課後使用這款軟體進行自主練習。之後，教師會透過每週在課堂中的簡單調查來了解學生在課後使用 CodeCombat 進行練習的情況，藉此觀察學生對於這類學習方式的看法與接受度。最後，本計畫在第 17 週安排一個實驗活動來測驗學生之前練習的成果，例如在限定時間內到達指定關卡，然後在測驗結束後會請學生進行實驗問卷的填寫，藉此量化學生對於這類學習方式的看法。



圖5. 實驗活動

(2). 研究成果

表 1 是本計畫的參與者人口統計特徵描述，其中本計畫的參與者都是來自南臺科技大學的多媒體與電腦娛樂科學系一年級的學生。本次實驗一共有 28 位參與者，其中包括有男性 20 位與女性 8 位。大部分參與者接觸程式設計的時間都是一年，只有少部分的參與者有接觸較長的時間。此外，過半數的參與者都認為寫程式是一件困難的事情。

表1. 參與者的人口統計特徵

特徵	類別	數量
性別	男性	20
	女性	8
年級	大一	28
接觸程式時間	1 年	18
	2 年	3
	3 年	2

	4 年	4
	6 年	1
寫程式對你來說	非常困難	2
	困難	10
	普通	10
	容易	5
	非常容易	1

表 2 是本計畫的信度分析結果，其中本計畫使用 Cronbach's α 對問卷進行信度的評估。具體來說，五個構面中的 Cronbach's α 值都高於 0.70(PEU = 0.872; PU = 0.854; PP = 0.923; S = 0.940; CI = 0.889)。換言之，這表示此問卷的可信度夠高(Wortzel, 1979)。此外，每個更正後項目總計相關性的最小值都高於 0.5，而這也顯示出問卷具有強大的可信度(Ong et al., 2009)。

表2. 信度分析的結果

構面	項目	信度分析	
		更正後項目總計相關性	Cronbach's α
知覺易用性(PEU)	PEU1	0.784	0.884
	PEU2	0.792	
	PEU3	0.747	
	PEU4	0.671	
	PEU5	0.625	
知覺有用性(PU)	PU1	0.779	0.842
	PU2	0.606	
	PU3	0.713	
	PU4	0.642	
知覺有趣性(PP)	PP1	0.790	0.929
	PP2	0.800	
	PP3	0.787	
	PP4	0.899	
	PP5	0.797	
滿意度(S)	S1	0.920	0.954
	S2	0.928	
	S3	0.831	
	S4	0.894	
持續意圖(CI)	CI1	0.842	0.910
	CI2	0.849	
	CI3	0.649	
	CI4	0.653	
	CI5	0.879	

表 3 是本計畫的效度分析結果，其中本計畫使用建構效度對問卷進行效度的評估。具體來說，建構效度是通過收斂效度與區別效度來進行評估(Ong et al., 2009)。收斂效度是透過檢

查平均方差(AVE)來進行評估，其中它必須高於最低標準 0.5 以上(Hair et al., 2006)。區別效度是透過 AVE 的平方根與構面相關矩陣進行評估(Fornell & Larcker, 1981)，其中每個構面的 AVE 的平方根應該超過與其他構面共享的相關性數值。表 3 的結果顯示，大部分的數值都超過標準，進而獲得令人滿意的建構效度。

表3. 收斂效度與區別效度的結果

構面	收斂效度		區別效度			
	AVE	構面相關矩陣				
		CI	PEU	PU	PP	S
CI	0.738	0.859				
PEU	0.681	0.242	0.825			
PU	0.780	0.782	0.465	0.883		
PP	0.687	0.417	0.681	0.456	0.829	
S	0.886	0.834	0.470	0.832	0.490	0.941

表 4 是本計畫的參與者滿意度調查結果，其中本計畫針對五個構面一共 23 項題目進行調查。具體來說，本計畫透過觀察整體問卷的連續數值變化，發現構面題項的平均數值沒有出現低於 3 分的情況，這表示參與者對於這類學習方式是可以接受的。

表4. 滿意度調查

構面	項目	非常不同意	不同意	普通	同意	非常同意	平均
PEU	PEU1	4%(1)	7%(2)	14%(4)	32%(9)	43%(12)	4.03
	PEU2	0%(0)	7%(2)	25%(7)	32%(9)	36%(10)	3.96
	PEU3	4%(1)	7%(2)	21%(6)	36%(10)	32%(9)	3.85
	PEU4	0%(0)	11%(3)	29%(8)	39%(11)	21%(6)	3.71
	PEU5	0%(0)	4%(1)	25%(7)	28%(8)	43%(12)	4.10
PU	PU1	0%(0)	0%(0)	28%(8)	28%(8)	44%(12)	4.14
	PU2	4%(1)	7%(2)	25%(7)	32%(9)	32%(9)	3.82
	PU3	0%(0)	7%(2)	25%(7)	25%(7)	43%(12)	4.03
	PU4	0%(0)	7%(2)	36%(10)	25%(7)	32%(9)	3.82
PP	PP1	0%(0)	4%(1)	25%(7)	35%(10)	36%(10)	4.03
	PP2	0%(0)	4%(1)	28%(8)	29%(8)	39%(11)	4.03
	PP3	0%(0)	7%(2)	39%(11)	18%(5)	36%(10)	3.82
	PP4	0%(0)	11%(3)	28%(8)	29%(8)	32%(9)	3.82
	PP5	0%(0)	7%(2)	50%(14)	25%(7)	18%(5)	3.53
S	S1	0%(0)	0%(0)	28%(8)	36%(10)	36%(10)	4.07
	S2	0%(0)	7%(2)	28%(8)	29%(8)	36%(10)	3.92
	S3	0%(0)	11%(3)	32%(9)	29%(8)	28%(8)	3.75
	S4	0%(0)	4%(1)	32%(9)	32%(9)	32%(9)	3.92
CI	CI1	4%(1)	7%(2)	39%(11)	29%(8)	21%(6)	3.57
	CI2	4%(1)	4%(1)	35%(10)	43%(12)	14%(4)	3.60

CI3	0%(0)	11%(3)	28%(8)	43%(12)	18%(5)	3.67
CI4	0%(0)	14%(4)	44%(12)	28%(8)	14%(4)	3.42
CI5	0%(0)	11%(3)	35%(10)	29%(8)	25%(7)	3.67

圖 6 是本計畫的五個構面平均的雷達圖，其中持續意圖(CI)與其他構面的平均值相比明顯較低。具體來說，持續意圖的平均值低於其他構面，這表示參與者繼續使用 CodeCombat 的意願沒有非常高。換言之，學生們可能不會主動去使用 CodeCombat 進行練習。

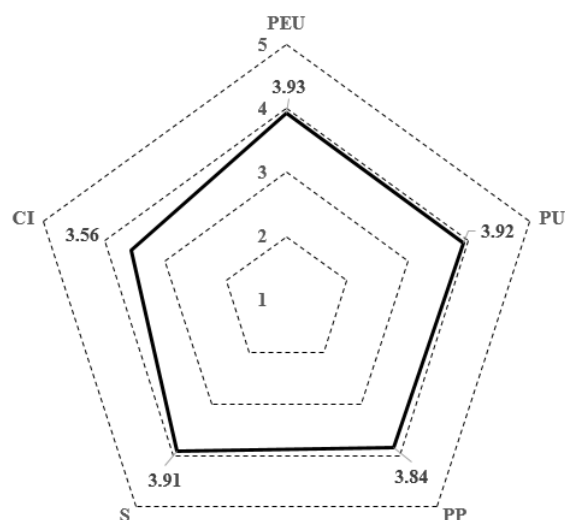


圖6. 五個構面平均的雷達圖

表 5 是本計畫的相關分析結果，其中學習成績與知覺易用性(PEU)、知覺有趣性(PP)呈現正相關。具體來說，學習成績與知覺易用性(PEU)、知覺有趣性(PP)顯示正相關，這表示學期成績與這兩個構面會互相影響。換言之，如果教師要使用這類遊戲進行教學，其中這類遊戲必須具備容易操作與遊玩有趣等兩個特性，才能引起學生的學習動機，進而影響學生的學期成績。

表5. 相關分析的結果

	PEU	PU	PP	S	CI
學期成績	0.508**	0.251	0.388*	0.293	0.252

** $P < 0.01$, * $P < 0.05$.

另一方面，本計畫整理學生的回饋意見，並根據意見性質進行分類。其中大部分的學生都認為程式教育遊戲是有幫助學習的，尤其對於程式初學者而言，這類遊戲可以更容易幫助他們學習程式邏輯。**關於正面意見：**

“很好玩，值得想學程式又害怕程式的人可以先接觸。”、“如果大一一開始就玩這種遊戲，不知道會不會讓他們更有動機學程式呢？”、“我覺得利用程式遊玩遊戲挺不錯的。”

但是，有部分學生對於程式教育遊戲是否可以學習程式抱持懷疑態度，而這部分可以解釋為什麼學期成績與知覺有用性(PU)沒有顯示正相關的原因，因為這類遊戲的訓練跟真正撰寫程式碼之間存在落差，所以學生可能對於其學習成效抱持疑問。**關於質疑意見：**

“可以練習邏輯，但是能否真正利用這類程式碼來撰寫程式，我抱持疑惑。”、“我覺得很棒，但跟真的寫程式有差一些不過會讓人想學看看程式。”、“拿來增加自信心的遊戲，但在幫助程式來說還好。”

另外，有些學生認為 CodeCombat 的關卡提示不明顯，因為他們需要花費時間去了解關卡提示的用意。**關於負面意見：**

“有些關卡的目標標示不清楚，要理解一段時間才會知道要我們做什麼事。”、“中文版的提示與教學有時候會讓人誤會，可以優化一下。”、“挺好玩的，但是提示的中文字需要有一定的理解能力。”

(3). 教師教學反思

本計畫將業師協同授課遭遇到的教學問題歸納成兩個部分，分別是學生未接觸的知識和業師與學生之間的互動等兩個問題。首先，教師在第一週觀察到業師在授課過程中存在一個盲點，便是業師在授課過程中所提到的相關知識，因為業師可能認為學生應該都了解這些知識，所以在授課過程中並不會特別說明，但是其實學生不一定了解業師說明的內容，因而產生學生在學習上的困擾。然後，教師還發現另一個問題，就是業師在授課過程中會與學生進行互動，但學生可能基於陌生或害羞等情緒，所以沒有回答業師拋出的問題，進而產生課堂學習氣氛低落的情況。因此，教師在發現這兩個問題後，開始針對目前的教學方式進行反思，同時與業師討論解決辦法。最後，教師與業師在第二週實施修改後的教學方式，就是由業師擔任主要授課者，而教師則擔任從旁協助者，其中當業師在授課過程中說明學生尚未接觸過的知識時，教師便會針對內容進行補充說明。另外，教師也會適時協助業師與學生之間的互動情形，藉此創造優良的學習氣氛。

本計畫將實施程式教育遊戲協助學生進行個人練習遭遇到的教學問題進行反思，其中本計畫發現學生不會主動使用 CodeCombat 進行個人練習。具體來說，教師鼓勵學生在課後使用 CodeCombat 進行練習，但經過每週課堂調查發現，學生們很少使用 CodeCombat 進行練習。另一方面，本計畫從整理後的問卷數據觀察發現，學生們普遍認為 CodeCombat 是一款很有趣的程式教育遊戲，但是他們並不會主動去使用 CodeCombat 練習程式語法。因此，教師開始進行教學反思並推測可能影響的原因，其中教師推測可能是這兩個原因。第一個原因，教師推測可能是課程安排造成的影響，因為學生在課後使用 CodeCombat 進行練習並不是一個強制性的活動，而原因是教師需要先上完原本預定的課程進度，所以只能利用課後時間進行教學實驗活動，因此這可能是影響學生沒有非常要想使用 CodeCombat 練習程式語法的原因之一。第二個原因，教師推測可能是學生不信任 CodeCombat 的學習成效，因為 CodeCombat 的訓練方式主要是讓學生去學習程式邏輯，而不是透過撰寫程式碼來完成某個功能，所以學生可能認為 CodeCombat 對於練習撰寫程式碼方面沒有太大的幫助，因此這可能也是影響的原因之一。然而，本計畫並沒有解決上述問題，而且本計畫的研究樣本較少，因此本計畫在未來會探討後續研究的問題。

二、參考文獻

英文

- Amenyo, J. T. (2012). Playable serious games for studying and programming computational stem and informatics applications of distributed and parallel computer architectures. *Journal of Educational Computing Research*, 47(4), 351-370.
- Corral, J. M. R., Balcells, A. C., Estévez, A. M., Moreno, G. J., & Ramos, M. J. F. (2014). A game-based approach to the teaching of object-oriented programming languages. *Computers & Education*, 73, 83-92.
- Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming. *ACM Computing Surveys*, 37(2), 83-137.
- Hainey, T., Westera, W., Connolly, T. M., Boyle, L., Baxter, G., Beeby, R. B., & Soflano, M. (2013). Students' attitudes toward playing games and using games in education: Comparing Scotland and the Netherlands. *Computers & Education*, 69, 474-484.
- Moreno, J. (2012). Digital competition game to improve programming skills. *Educational Technology & Society*, 15(3), 288-297.
- Mathrani, A., Christian, S., & Ponder-Sutton, A. (2016). PlayIT: Game based learning approach for teaching programming concepts. *Educational Technology & Society*, 19(2), 5-17.
- Michalewicz, Z., & Hartley, S. J. (1996). Genetic algorithms+ data structures= evolution programs. *Mathematical Intelligencer*, 18(3), 71.
- Linn, M. C., & Clancy, M. J. (1992). The case for case studies of programming problems. *Communications of the ACM*, 35(3), 121-132.
- Prensky, M. (2007). *Digital game-based Learning*. New York, NY: McGraw-Hill.
- Rieber, L. P. (1996). Seriously considering play: Designing interactive learning environments based on blending of microworlds, simulations, and games. *Educational Technology Research & Development*, 44(2), 43-58.
- Sampayo-Vargas, S., Cope, C. J., He, Z., & Byrne, G. J. (2013). The effectiveness of adaptive difficulty adjustments on students' motivation and learning in an educational computer game. *Computers & Education*, 69, 452-462.
- Wang, L. C., & Chen, M. P. (2010). The effects of game strategy and preference-matching on flow experience and programming performance in game-based learning. *Innovations in Education and Teaching International*, 47(1), 39-52.

中文

- 邱文傑。(2015)。**翻轉教室教學對於大學生程式設計學習成效之研究**(碩士論文)。取自臺灣博碩士論文系統。(系統編號 103PU000676007)
- 教育部。(2002)。**技職教育白皮書**。載於 <https://lms.cit.cyut.edu.tw/1997041/doc/86465>。
- 陳羿介。(2010)。**應用於圖書館導覽的行動遊戲式學習系統之介面設計與評鑑**(碩士論文)。取自臺灣博碩士論文系統。(系統編號 098NCTU5331004)
- 葉思義。(2010)。**數位遊戲設計達人講座**。臺北：基峯。

蔡明志。(2011)。**資料結構：使用 Java**。碁峯資訊股份有限公司，第三版。

賴怡君。(2008)。**結合概念圖模式於線上知識診斷與補救教學系統之研究-以 Java 程式設計課程為例**(碩士論文)。取自臺灣博碩士論文系統。(系統編號 097CYUT5396024)

謝樹明。(2013)。**細談資料結構**。旗標出版股份有限公司，第六版。

三、 附件

表6. 研究問卷題目

構面	項目	題目
知覺易用性(PEU)	PEU1	我認為「這類遊戲」是容易玩的。
	PEU2	我認為「這類遊戲」的操作是容易理解的。
	PEU3	我認為「這類遊戲」的玩法是人性化的。
	PEU4	我認為要熟悉「這類遊戲」是容易的。
	PEU5	我認為學習如何玩「這類遊戲」是容易的。
知覺有用性(PU)	PU1	我認為使用「這類遊戲」學程式，其效果更佳。
	PU2	我認為使用「這類遊戲」學程式，我會學得更好。
	PU3	我認為使用「這類遊戲」學程式，我花更少的時間學習。
	PU4	我認為「這類遊戲」提供的學習方式比其它的方式更好。
知覺有趣性(PP)	PP1	我認為「這類遊戲」是好玩的。
	PP2	我認為「這類遊戲」是有趣的。
	PP3	我認為「這類遊戲」是令人感到興奮的。
	PP4	我認為「這類遊戲」是令人感到愉悅的。
	PP5	我認為「這類遊戲」是令人感到刺激的。
滿意度(S)	S1	對於使用「這類遊戲」學程式，我是肯定的。
	S2	對於使用「這類遊戲」學程式，我感到滿意。
	S3	對於使用「這類遊戲」學程式，我感到滿足。
	S4	對於使用「這類遊戲」學程式，我感到高興。
持續意圖(CI)	CI1	我想要增加我使用「這類遊戲」的次數。
	CI2	如果可以，我打算繼續使用「這類遊戲」學程式。
	CI3	我想要繼續使用「這類遊戲」學程式，而不是停止使用它。
	CI4	我使用「這類遊戲」學程式的意圖，勝過使用其它任何傳統的方式。
	CI5	即使有其它學程式的方式，我仍然想要繼續使用「這類遊戲」學程式。