

天天玩遊戲就可以學寫程式-以 codecombat 為例

士林高商 翁聖恩

壹、前言

傳統上教授程式語言課程的方式，不管是電腦教室上機實作或傳統教室講解虛擬碼。不外乎教師講、學生聽，學生皆處於被動的狀態。而此法或許可以方便達到授業的目的，但易養成學生不願思考的被動式學習態度，以致在「解惑」上就鮮有成效。而傳統聽講的授課方式，學生僅就課堂上老師所講做好筆記，並於課後加以背誦牢記，要通過考試並不困難，惟因缺乏學習動機，學習難以精進。

就筆者個人的授課經驗，許多學生對程式設計充滿了恐慌，這是他們從未接觸過的領域。在上機實作時，也只在老師所舉範例中，尋找可能的答案，若提出開放式問題，學生則顯現毫無頭緒之貌，更無法期望他們思尋可能解決問題的方法。因為大部份學生已習慣於記憶許多繁瑣的語法，從既有的虛擬碼及四個選項尋求解答，而不願對問題的解決方法多加思考，筆者於是思考如何在程式設計之教學上作些改變，藉由安排角色扮演遊戲活動的方式，讓學生在參與活動的過程中學習思考，逐步加強解決問題的能力、降低不安的心、取得神馳(Flow)經驗，進而提高學習程式設計的興趣。

貳、遊戲式教學對於神馳(Flow)的影響

一、神馳經驗

神馳(Flow)是人類覺得活動在控制中，自己是自己命運的主人，感覺一種深沉愉快的享受。那種狀態，是個人完全投入活動中，其他的事都不重要，經驗本身就是目的，是如此愉悅，因此人們為了完全獲得，會更不計成本去做Csikszentmihalyi (1990)。而神馳有二個特色，第一是「挑戰」和「技巧」的感知，這是影響最優經驗的主觀變項；第二是神馳不會有極限存在，即個人會持續尋求更複雜及更大的享受，若學生在活動的過程中，能達到神馳境界，則不待老師指導，就會主動學習。電腦興起，有研究也採用神馳理論去探討網路瀏覽或電腦遊戲活動的情境(劉旨峰、周倩、林珊如，2001；黃瓊慧，2000；Webster、Trevino & Ryan, 1993)。Webster、Trevino & Ryan 等人(1993)研究人與電腦的互動，認為神馳是一種主觀的人機互動的經驗，具有遊戲性和探索的特質，在人機互動的期間，個人能主觀的感知到愉悅和涉入，而較高的遊戲特質，則可以得到較正面的情緒和滿意，並引發個人進一步的探索。嚴祖弘 (2001)也認為學生會迷上遊戲，頗符合遊戲理論中神馳經驗的描述，學生會陷在其中跳不出來，一方面是求技能的提高，另一方面是尋求更高的挑戰，如果技能超過挑戰，參與者就感到無聊；反之，如果挑戰超過技能太多，參與者就感到焦慮，所以，學生就在自我技能與挑戰之間相互拔河。遊戲式可以促進學習者的內在動機。在神馳時，會花更多的時間參與來達成，獲得讓學習者滿意的效果(蘇佩芬、李蔡彥、廖玟俐，2002)

二、遊戲的構成元素

Prensky (2000)認為構成遊戲的主要元素有下列六個：

- (1) 規則(Rules):規則是讓遊戲者知道遊戲的一些限制，以及允許他行動的準則。
- (2) 目標(Goals and objectives):即學習目標。
- (3) 結局和回饋(Outcomes and feedback):提供外在籌賞(音效、動畫、具特殊功能的物品等)可促進學生的學習動機。
- (4) 競爭和挑戰(Conflict/Competition/Challenge/Opposition):遊戲通常含有某種程度的競爭，且遊戲吸引人的最重要因素就是提供某種形式的挑戰。
- (5) 互動(Interaction):適當的提示教學和回饋，可讓學生更投入在遊戲互動時的心理滿足感。
- (6) 故事情節(Representation or story):教學遊戲主要目的是教學，但利用故事情節的吸引力來引發學習動機和加強學習效果。

三、遊戲教學的機制

遊戲教學需要有一個完整的決策模式機制，來決定學習者的學習機制。而決策模式有三個元素，

1. 障礙 2. 任務 3. 說明

- (1) 障礙：為了引導出要傳遞的訊息，多種答案及選擇是不可或缺的。對於參與者，從中選出一個好且正確的答案是困難的。因此可驗證出參與者的學習程度是否有到達某個水平值。
- (2) 任務：即使有了多重選擇，如果選擇的對錯或選項對遊戲的進行沒有影響，也許就會影響參與者作答的意願，因此該選項必須設計跟勝負有關或遊戲劇情有關，參與者較能引起學習興趣及作答意願。
- (3) 說明：如果學習者不瞭解該選項的意義或者是要怎麼選？就會干擾到學習者的選擇或最後的勝負，因此必須要有一些資訊，提供給學習者幫助他們完成這個遊戲。

一個遊戲式學習活動，就是由以上三個元素結合而成。而在學習遊戲中的核心知識，就是評價整個遊戲式學習是否成功的關鍵，核心知識就是藉由選項的選擇來傳遞，參與者為了選到對的答案，就會運用所學的核心知識來解決問題，所以可以說，寓教於樂。而Prensky (2000)指出，遊戲式學習活動可達到的教學效果有下列幾項1. 主動學習2. 提高學習興趣3. 個別化的學習與體驗知識4. 減輕學習壓力5. 創造性的思考與學習6. 補救教學。

參、程式設計教學遊戲

一、程式設計教學遊戲簡介與比較

遊戲名稱	簡介	優點	缺點
Google Blockly	一款 Web-based，以圖形拖拉組合、類似堆積木的方法，來寫程式的一個圖形化程式語言	<ol style="list-style-type: none"> 積木式建構語法 可轉換為 JavaScript、Python、XML 等進階語法 多國語言介面 	<ol style="list-style-type: none"> 關卡數目前偏少 以堆積木方式，學生缺乏動手寫程式的感覺 無故事性及循序漸進
Light-bot	以簡易的直走、右轉、左轉、跳躍、點燈、函式 1、函式 2 七個指令，控制小機器人，解決各項難題	<ol style="list-style-type: none"> 積木式建構語法 指令簡易 	<ol style="list-style-type: none"> 因指令過於簡易，若學生自學時遭遇難題，沒有老師指導，容易因卡關而放棄 關卡數偏少 無中文介面
Scratch	通過拖曳預先設定好的樂高積木式圖形化程式模件，表達程式指令，設定和控制每一個角色或背景的行動和變化，從而完成程式設計	<ol style="list-style-type: none"> 積木式建構語法 內建豐富圖片和音效素材 可自行製作並匯入素材 多國語言介面 所開發出的程式原始碼可上傳至網路與眾人分享 	<ol style="list-style-type: none"> 須以指定寫法撰寫，彈性較差。 支援其他程式語言支援性較差 使用的圖形為點陣圖，跨平台的彈性較低 無故事性及循序漸進
Code.org	以「一小時的獨立小課程」來讓老師、家長、學生自由組合出不同的學習流程，偏向「程式運算的邏輯」，也就是以電腦科學的基礎架構為主，讓學生先學會程式的運作觀念。	<ol style="list-style-type: none"> 課程是用知名遊戲或電影的角色來組成 獨立小課程可在短時間內傳達完整的概念 有完整配套的整體任務，學習大型專案開發 	<ol style="list-style-type: none"> 無法自行開發課程協作 解題過程屬於個人解題，缺乏社群的互動 無故事性及循序漸進
codecombat	以大型角色扮演遊戲方式，將程式語言包裝成 RPG 英雄打怪、尋寶與解謎遊戲。在有趣的奇幻世界與戰鬥關卡中學會各種程式的基本技巧。	<ol style="list-style-type: none"> 已有完整繁體中文介面 以線上遊戲方式呈現 有 110 多個免費的關卡 能夠學習 Python、JavaScript、Lua、CoffeeScript、Clojure 等不同程式語言 導入 RPG 元素，吸引學生參與 以鍵入語法方式自由度高 	<ol style="list-style-type: none"> 免費的關卡難度較低，若要更深更廣須另行付費

二、codecombat 教學遊戲介紹

1. 友善的登入畫面

CodeCombat 上手簡單，不需要額外指導，任何玩過遊戲的朋友一定能立刻開始遊戲，也能從中學習到各種程式語言的基礎。



只需用 Google 或 Facebook 就能登入「CodeCombat」

教師還可建立教師身分，開創團體任務，讓學生加入。可以觀察學生每一個人的學習進度，並且互相激勵且合作過關。



挑選一位你喜愛的英雄，並選擇自己想要學習的程式語言。而隨著過關的任務越多，能選擇的英雄越多。

隨著任務的演進，英雄可以添購新的裝備跟技能，也就是學到新的語法。把遊戲裡的命令操作、武器道具都變成程式語言的指令，可以使用語法去攻擊敵人，或是當獲得技能寶典時可以學會讓程式自動迴圈的新指令。能在打怪、尋寶的過程中，慢慢理解每個程式語法的作用。





可以選擇要闖關的地區，不同地區可學到不同的語法。循序漸進，依序可學到參數、字串、迴圈、變數、狀況判斷、關係運算符號、物件屬性、輸入處理、數學運算、計數器、迴圈、迴圈中斷、陣列、變數比較、尋找最大最小值、物件執行、FOR 迴圈、繪畫等語法。

新學到的語法會有詳細的用法說明及其變化。



每一關都會有其過關條件及額外特殊條件，如將程式碼減量或執行時間減少。所有的程式語法皆以鍵入方式輸入，關卡裡都有詳盡的提示，提示要如何輸入程式語言來命令角色行動，當輸入錯誤時，也有清楚的提示為什麼這樣的語法會出錯。

按下運作，可觀看程式執行過程與結果。



若符合過關條件或特殊條件，則發放經驗值與寶石，讓你的英雄可以添加新的裝備。

過關時，可觀賞其他過關者所撰寫的程式，或將自己撰寫的程式上傳至社群中分享



四、結語

利用 codecombat 教學有下列幾項好處：

一、建立學生邏輯思考能力：

雖然 codecombat 像一個線上遊戲，但它是一個簡單卻能培養學生邏輯思考能力。在商管群的學生學習中，除了資處科之外，其餘科能學習程式語言的機會只有少少一個學期，而若能善用 codecombat，雖不能完全取代程式語言的教學，但學生可以在一個月內的課程內完成初學的過程，讓學生先有先備知識，減少對程式設計的懼怕感。

二、解決問題與實作的能力：

codecombat 遊戲中，每一個關卡，其解決方式並不只有一種，並不侷限在函式的功能有多強大，而在於人們的創意有多寬廣。條條大路通羅馬，只要能解決問題的方法，都是好方法。因此學生在學習的過程中，可以無限制的展現解決問題的創意，老師不再是提供解答者，老師只是工具的提供者和問題諮詢者，是由學生自由天馬行空解決問題。而 codecombat 所提供的排名功能，更能刺激學生如何將程式精簡及如何能讓程式執行的更快更有效率，老師藉此帶入演算法的想法，發揮學生精益求精的精神。

三、合作學習的體驗：

線上遊戲迷人之處，在於跟其餘玩家組隊互動，一個隊伍中，有劍士當盾、弓箭手遠程攻擊、魔法師施展群體攻擊、牧師幫忙醫療補血，每個人各司其職小組合作，而 codecombat 中除了單人關卡之外，也提供多人解題成就。不管是雙人、三四人以上，皆有配合的關卡供學生解任務。而合作學習是一種運用很廣的教學方法，許多研究證實合作學習有助於提升學生的學習成就、學習動機及班級氣氛，是一種適合運用在各年級階段的有效教學策略。當學生在學習過程中遇到困難及認知衝突時，透過彼此討論、溝通與互助合作，不僅可以解決問題，更可以提升認知能力、批判思考能力、問題解決能力，並造成學習遷移。擁有共同目標，並共同解決問題，也可增進同儕之間的感情。

四、高度的學習動機：

由於 codecombat 是以線上角色扮演遊戲為基底設計的特色，簡易好上手，完成度高，即時有反應，貼近學生平常所玩的遊戲，更是提高學習動機的好時機，學生常在練習解題的過程中，下課也在群聚討論的反應，是教學上鼓勵學生思考的，體驗認真思考的好課程。

但利用 codecombat 來教授程式設計也非百利而無一害：

一、程式之間的轉換：

由於課綱目前所教授的是 VB，而 codecombat 中尚無 VB 的程式語言，若學生已熟練 codecombat 中的程式語言，要轉換至 VB 上是否能銜接的好，需要教師勞心勞力。

二、配備需求：

codecombat 全程執行皆需網路運作，以 HTML5 方式運行，目前尚無單機版或行動裝置版，若校園內網速不足或配備不佳，有可能會造成延遲的情形，因此設備的需求是必須事前注意的。

三、遊戲沉迷：

根據研究，神馳(Flow)與沉迷(Addictive)具有顯著的正向關係，當玩家過於追求遊戲所帶來的快樂，將導致自己沉迷於遊戲當中。神馳經驗對於沉迷玩家的影響，大多為玩家本身行為及意志上的衝擊，例如：無法控制的強迫性上網症狀，無法離開線上遊戲太久的戒斷症狀，以及為了追求遊戲所帶來的愉悅感受，而投入大量時間在遊戲中所形成的耐受性症狀(莊雅茹,2005)。因此，教師在引入 codecombat 時，須明確跟學生建立良好的學習習慣，避免產生沉迷心態。

參考文獻

- 黃瓊慧 (2000)。從沉浸(flow)理論探討台灣大專學生之網路使用行為。國立交通大學傳播研究所碩士論文。
- 劉旨峰、周倩、林珊如 (2001)。以心流理論為基礎探討台灣大學生網路沉迷相關因素。國科會研究計畫，計畫編號：NSC87-2511-S-009-013-N。
- 嚴祖宏 (2001)。從休閒輔導建立青少年人生視野。輔導通訊，66，1-4。
- 蘇佩芬、李蔡彥、廖玟俐 (2002)。網路教學之推動策略—以政治大學為例。網路教學與圖書資訊學應用研討會論文集，台灣。
- Prensky, M. (2000). *Digital Game-based Learning*. New York:McGraw-Hill Book.
- Webster, J., Trevino, K. L. & Ryan, L. (1993). The emergence of a new clinical disorder. *CyberPsychology & Behavior*, 1(3), 237-244.