
機器人問題導向程式設計課程對女高中學生 學習程式設計影響之研究

王裕德* 陳元泰 曾鈴惠

國立臺中女子高級中學

摘 要

本研究目的是想瞭解機器人融入高中程式設計課程中，對女高中生在學習程式設計成就及態度之影響。本研究以台灣中區一所女校之高一學生為樣本，隨機選派 2 班學生，一班為控制組(42 人)，另一班為實驗組(42 人)，二班學生先實施六週(12 節)基礎程式設計課程，基礎課程結束後實施程式設計成就及態度前測，隨後控制組進行問題導向程式設計課程六週(12 節)，實驗組進行問題導向機器人程式設計課程六週(12 節)，課程實施中並隨機挑選實驗組學生進行半結構性訪談，以瞭解學生的學習狀況。課程結束後再實施程式設計成就及態度後測，研究資料以 MACOVA、Paired T-Test 等統計方法進行研究考驗，統計考驗後發現不同教學策略分組的學生其程式設計學習成效沒有顯著差異，但實驗組學生在接受完問題導向機器人程式設計課程後程式設計態度有顯著提昇；學生訪談記錄經質性分析也發現藉由機器人來學習程式設計較能瞭解程式的作用且較能引起學生學習動機及興趣。

關鍵詞：機器人、程式設計、問題導向、女高中生

壹、研究背景與動機

Duke, Salzman, Burmeister, Poon & Murray (2000)認為學習程式設計能夠培養學生問題解決以及高層次思考能力，並有助於他們日後的職業規劃。但是程式設計在高中階段卻是一項難教也難學的科目，Linn & Clancy (1992)認為傳統的程式設計教學太著重於語法的教學，容易使得初學者落入嘗試錯誤的方式來拼湊程式，

Costelloe (2004)指出學生剛開始學習程式設計時，在語法及程式概念上常會碰到困難而感到沮喪。由於一般的程式語言語法結構龐大複雜，容易造成程式初學者學習的困擾(Mannila, Peltomäki, & Salakoski, 2006)。初學者學習程式設計的主要困難在於難以跨越一道鴻溝，亦即人們腦海中所想的解題方法與電腦所能接受的解題描述方式距離遙遠(Smith, Cypher & Tesler, 2000)。因此，上述學者們嘗試使用不同的教學方法，希望能增進學生學習程式設計

* 為本文通訊作者

的成效。

樂高(以下簡稱LEGO)公司所出品的Mindstorms 機器人,其前身是MIT 開發出來的可程式化積木,樂高積木通常包括了許多大小、形狀、顏色不同的塑膠積木,且可經由相互的組合而連結在一起,進而建構出各式各樣不同成品,它就是結合了視覺化及微世界概念的實體機器人,幫助各級學校進行機器人教育的推廣,讓學童及青少年可提早接觸機器人的操控方式,為將來的學習做好準備。因此,許多的學者將其運用於程式教學上。Klassner (2002)在大學人工智慧課程的教學上採用Mindstorms 讓學生以小組專題的方式學習;Jorgen & Thomas (2007)使用LEGO Mindstorms在小學訓練學生解決問題的能力。

許多的研究中可以發現學生對科學的態度有性別差異的現象,樣本從國小至高中皆有,研究發現男生態度比女生積極(Shrigley, 1972; Gardner, 1975; Leeves, 1975; Power, 1981; Haladyna, Olsen & Shaughnessy, 1982; Fleming & Malone, 1983; Hartly & Beall, 1984; Olstad & Haury, 1984; Schibeci & Railey, 1986)。另有研究發現,男女生由國小至高中對科學的態度皆呈下降的趨勢,而且有些研究指出女孩下降的程度大過於男孩(Ayers & Price, 1975; Reid & Tracey, 1985; Yager & Yager, 1985; Kelly, 1986; Yager & Penick, 1986)。

在台灣地區,男學生在科學知識、一般科學的興趣顯著高於女學生,男學生對

於科學的態度亦顯著高於女學生;對於學習科學的必要性,男學生有顯著較高的體認;男學生對於應用科學具有明顯較女學生強烈的意圖,且將科學應用於休閒及生活上,男學生的意圖都顯著高於女學生,顯示在應用科學的意圖上具有顯著的性別差異(陳志豪,2004)。國外研究亦顯示,男性對於物質科學(如物理學、化學)的態度顯著高於女性;但是生物科學及醫學較能吸引女性的青睞,使女性對生物科學及醫學的態度顯著較男性為高(Dawson, 2000)。

因此,本研究想瞭解LEGO Mindstorms 機器人融入高中程式設計課程中,對女高中生在學習程式設計成就及態度之影響。

貳、文獻探討

一、程式設計教學

程式設計是一項需要邏輯推理能力的認知活動,在學習的過程中能夠培養學生高層次思考以及邏輯推理的能力(Cañas, Bajo & Gonzalov, 1994; Costelloe, 2004)。在撰寫程式的過程中學生必須經歷兩個階段,首先必需了解題目想出解題的方法(演算法),就是問題解決的階段;再來將問題的解法轉換為程式碼,意即所謂的程式撰寫與實踐的階段(Costelloe, 2004)。在問題解決階段對初學程式設計的學生而言是相當困難的,Soloway (1986)認為以往程式設計教學太過著重語法而忽略教導問題解決能力的重要。Costelloe 所提出的程式撰寫與實踐階段,學生必需將各種程

式的概念組織起來才能完成程式，但教師們常因為學生不明白某些特定的程式概念而感到沮喪。Linn & Clancy (1992) 認為學生已具備一般問題解決的能力，但並不能將其運用在程式設計的解題方面，因此在教學上應教導學生如何將一般問題解決能力應用在程式設計方面，而過去的程式教學過於著重程式語法的教學，忽略了規劃程式的技巧。Soloway (1986) 也認為應該教導學生如何規劃程式的技巧，他認為程式設計的專家，在解題時能夠靈活運用其知識與策略。因此他建議可以採取 goal/plan 的教學策略，讓學生以 top-down 的方式學習規劃程式。開始解題時依題目的要求來確立目標 (goal)，然後細分有哪些已學過的策略 (plan) 可以達成目標，再逐步精化、合併每一個策略，就可以順利解題。Valente(1995)指出，程式設計過程中不斷反覆進行之「描述—執行—反思—除錯—描述」循環(the description-execution-reflection-debugging-description cycle)，對學生而言正是最有效的學習訓練。

二、機器人程式設計教學

在程式設計教學亦有整合微世界與視覺化觀念的教學設計，1990 年代美國麻省理工學院電腦教育學者們，設計出一套可程式化的實體積木 (tangible programming bricks)，並可以結合感應器讓可程式化的積木和外界的環境互動，也就是後來的 Lego Mindstorms (McNerney, 2004)。而機器人學是一門整合型的課程與技術，

LEGO Mindstorms 機器人是一套能夠應用圖形化程式設計的科技創新積木玩具。為了增加學習者程式設計學習的趣味性和應用性，同時降低初學者學習程式設計的障礙與恐懼，並增進學生主動探究問題，發揮創意合作學習，與培養學生解決問題的能力。蕭佳明、黃瑛綺(2012)調查發現採用電腦樂高課程進行機器人教育能夠激發學習者創意、給與學習者回饋與成就，並能節省教育成本。Dagdilelis, Sartatzemi & Kagani (2005)在希臘的中學針對九年級和十年級的學生進行 LEGO Mindstorms 教學，研究發現在程式設計中加入實體物件(即「機器人」)，使學生易於追蹤程式的執行過程，這對提升學習動機有明顯助益。陳怡伶(2009)研究發現學生對於應用 LEGO Mindstorms 作為程式設計學習輔具教材的喜好程度等層面，皆獲得正面的回應，尤其提升了初學者對於程式設計的學習興趣，亦有助於學習者於程式設計概念之學習。

Fagin 等人 (2001) 採用 LEGO Mindstorms 於程式設計導論的課程中，教學生循序、變數及常數的概念。首先要運用機器人教學生循序的概念，只要將要機器人執行的程式碼依序寫好，當機器人執行的時候自然就會依照先前寫好的順序執行。採用機器人於程式設計教學，因為其具有實體的特性，學生能藉由撰寫程式操控機器人，除了可以讓學生有動手實作的經驗之外，也富含趣味性。Klassner(2002)讓學生以 LEGO Mindstorms 為基礎來設

計人工智慧的專題，讓學生充份應用各種零件組裝發展出符合老師要求的專題，從評量的結果也顯示採用機器人的教學方式的確增進學生對於人工智慧概念的瞭解。近年來，國內亦有其他研究者使用 LEGO Mindstorms 在國、高中進行程式設計教學實驗，這些研究均大致肯定了 LEGO Mindstorms 對於提升程式設計學習動機與成效之助益。(蔡學偉，2004；許雅慧，2005；曾義智，2006；謝亞錚，2009)

三、問題導向學習

PBL(Problem-based Learning)的理念可追溯於進步主義運動 (progressive movement)，尤其是以Dewey的理念為核心。Dewey認為教師在進行教學應能夠吸引學生運用自然的本能去驗證與創造，學生的生活經驗才是教師統整課程的線索，而如此設計出來的課程才能引發學生的興趣與學習動機(高廣孚，1984)。Barrows(1992)認為，「問題導向學習」是以學生為中心與真實情境為依歸，透過病人的實際症狀所呈現資訊不足、紊亂待釐清的結構模糊問題為學習的內容，以小組討論為手段，促使學生主動參與學習、討論分析，來獲得解決問題的技巧與基礎和臨床科學的知識，並藉由教師及同儕的協助，得到學習上的回饋與更好的成效。因此，「問題導向學習」企圖讓學生能面對問題、解決問題，不像傳統教室的解題，總是在學會所有公式定理後，好整以暇地解出標準答案；相反地，從真實情境所獲

得的結構模糊問題，因為欠缺相關資訊，在條件不足的情況下，學生必須自行確認問題，蒐集解題資訊，隨著資料的更新與評估，促使學生以不同的角度來看待問題，藉此引導學生了解知識建構與共構的重要性，無形中培養出學生的自行探究、自我學習、問題解決、團隊合作等能力。

Edens(2000)亦提出PBL乃是使用一種真實、複雜的問題來當作學習的啟動來源並幫助學生同時獲得知識與問題解決的技能。陳毓凱(2005)於研究中彙整資料指出PBL應具有下列特徵：

- (1) 以學生為學習的中心並促進學生建構學習的過程。
- (2) 以結構模糊的問題來當作學生學習的起點。
- (3) 以小組合作的方式進行學習。
- (4) 教師扮演後設認知教練的角色。
- (5) 養成學生自我導向學習的能力並培養其終身學習的習慣。
- (6) 多元的評量方式。

然而依國情與需求的不同亦可做適當的改變，參考國內實施PBL之實證研究統整出其教學步驟(計惠卿、張杏妃，2001；張瓊穗等，2002；張民杰，2003；劉忠昱，2012)，對於PBL實際在教室中的運作過程我們可以將它分為數個步驟來說明：

- (1) 呈現問題引起學生的學習動機。
- (2) 第一次小組討論對問題進行分析並發展學習議題。
- (3) 學習者針對學習議題進行資料蒐集

並展開自我導向學習。

- (4) 第二次小組討論綜合歸納小組成員的學習結果並形成解題方案。
- (5) 學習成果展現。
- (6) 評估學習成果並進行反省評鑑。

綜合上述，可知 PBL 乃以學生為中心，藉由一些非結構性的問題，學生以小組來解決問題，並藉由解決問題的過程建構知識，並可藉由合作學習的過程訓練學生的團隊合作、解決問題及口語表達等能力，教師在此過程僅扮演催化者和學習促進者的角色，讓學生藉由這個過程養成自我導向學習及終身學習的習慣。

參、研究設計與實施

一、研究樣本

本研究以台灣中區女性高中一年級新生兩班為實驗對象，隨機抽出一班為實驗組(42 人)實施問題導向機器人程式設計課程，另一班控制組(42 人)實施問題導向程式設計課程進行實驗。刪除未填答問卷

及缺課等無效樣本，實驗組 37 人，控制組 35 人。總樣本數為 72 人。

二、研究設計與步驟

本研究以量的研究為主，質的研究為輔，限於原班級建制之教育環境與班級人數之限制，在不影響學校正常教學下採「不等組前後測設計」準實驗研究，為期 12 週，研究設計如圖 1，實驗組(O1)及控制組(O4)學生在前六週(每週二節)進行相同的基礎程式設計課程(E)，基礎程式設計課程結束後隨即進程式設計成就及態度前測(O2,O5)，第七週~第十二週，控制組實施問題導向程試設計課程(X2)，實驗組實施問題導向機器人程試設計課程(X1)，研究步驟如圖 2，實驗結束後立即針對二組進行後測(O3,O6)；並於每週實驗課程結束後針對實驗組學生進行約 30 分鐘的半結構性訪談，包括對程式設計的看法、對程式設計的學習狀況、對課程的想法...等，作為對教學策略質性分析之依據。

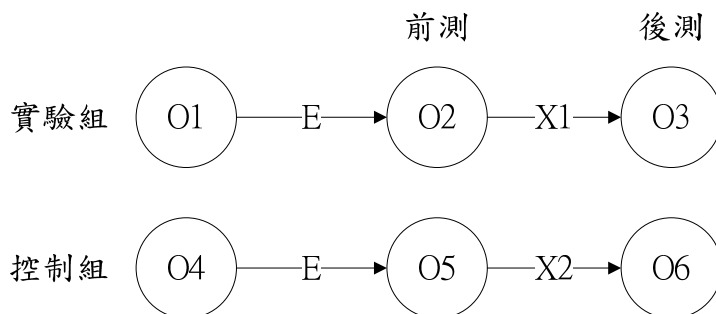


圖 1、研究設計

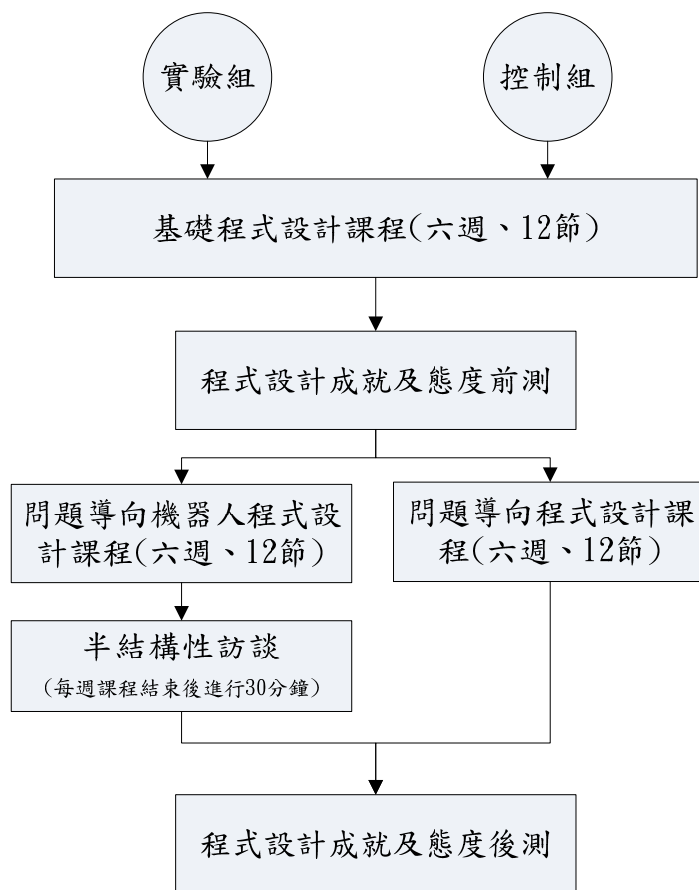


圖 2、研究步驟

三、研究工具

(一) 程式設計成就量表

本研究因分別導入 6 週(12 節)的 PBL 機器人程式設計課程及 PBL 程式設計課程，故僅挑選程式設計較重要的三個單元(基本概念、流程控制及副程式)進行教學實驗。而程式設計成就量表仍依據國立臺中女中國科會高瞻計畫第一年發展之程式設計能力指標與能力項目之權重為基礎，建構本程式設計成就量表，經五位高中程式設計任課教師建議，各指標命題數如表 1 所示。預試成就量表共 53 題，預試生為 147 位高中學生，試題選擇標準依據美國

學者 Ebel 所提出之標準(郭生玉，民 78)，如表 2 所示，至於難易度方面，以接近 0.5 為最理想之試題，但要找到所有題目都接近 0.5，實際上有其困難；Chase(1978)主張選擇題難易度以 0.4~0.8 為選題之標準，因此，本研究將鑑別度定為 0.2 以上，難易度定為 0.4~0.8 為選題之標準。程式設計成就量表之項目分析表如表 3，第 1、22、23、25、37、44、45 題因具代表性，故予以修正保留，刪除不適用之試題，確立正式量表之題數為 30 題，其 KR20 為 0.84。

表 1、程式設計成就量表雙向細目表

能力指標	權重	30 題	能力項目	權重	30 題
一、C++程式語言基本概念	0.3	9	1.C++語言概說	0.1	1
			2.識別字/基本資料型態	0.2	2
			3.常數/變數	0.2	2
			4.運算式	0.2	2
			5.輸入/輸出	0.3	2
二、流程控制	0.5	15	1.邏輯判斷	0.3	4
			2.選擇流程	0.2	3
			3.重覆控制流程	0.4	6
			4.中斷流程與跳離流程指令	0.1	2
三、副程式及其應用	0.2	6	1.系統副程式	0.3	2
			2.自建副程式	0.5	3
			3.遞迴副程式	0.2	1

表 2、鑑別度的評鑑標準

鑑別指數	試題評鑑
.40 以上	非常優良
.30~.39	優良、但可能需要修改
.20~.29	尚可、但通常需要修改
.19 以下	劣、需要淘汰或加以修改

資源來源：郭生玉(民 78)：心理與教育測驗，頁 271。

表 3、程式設計成就量表之項目分析表

題號	P 值	取捨	D 值	取捨	修正或刪除	正式題號
1	0.86	X	0.22		修正	1
2	0.31	X	0.13	X	刪除	
3	0.41		0.57			2
4	0.51		0.57			3
5	0.45		0.3			4

表 3、程式設計成就量表之項目分析表(續上頁)

題號	P 值	取捨	D 值	取捨	修正或刪除	正式題號
6	0.59		0.77			5
7	0.35	X	0.35		刪除	
8	0.47		0.45			6
9	0.26	X	0.18	X	刪除	
10	0.44		0.18	X	刪除	
11	0.4		0.25			7
12	0.46		0.58			8
13	0.29	X	0.43		刪除	
14	0.2	X	0.2		刪除	
15	0.29	X	0.32		刪除	
16	0.39	X	0.27		刪除	
17	0.71		0.48			9
18	0.32	X	0.4		刪除	
19	0.42		0.5			10
20	0.51		0.38			11
21	0.4		0.3			12
22	0.3	X	0.5		修正	13
23	0.34	X	0.28		修正	14
24	0.54		0.57			15
25	0.37	X	0.6		修正	16
26	0.32	X	0.3		刪除	
27	0.41		0.73			17
28	0.52		0.3			18
29	0.39	X	0.13	X	刪除	
30	0.54		0.52			19
31	0.47		0.3			20
32	0.35	X	0.25		刪除	
33	0.39	X	0.32		刪除	
34	0.34	X	0.33		刪除	

表 3、程式設計成就量表之項目分析表(續上頁)

題號	P 值	取捨	D 值	取捨	修正或刪除	正式題號
35	0.51		0.38			21
36	0.35	X	0.45		刪除	
37	0.26	X	0.27		修正	22
38	0.16	X	0.07	X	刪除	
39	0.3	X	0.4		刪除	
40	0.39	X	0.38			23
41	0.49		0.43			24
42	0.51		0.23			25
43	0.24	X	0.13	X	刪除	
44	0.31	X	0.22		修正	26
45	0.32	X	0.2		修正	27
46	0.27	X	0.2		刪除	
47	0.4		0.35			28
48	0.4		0.25			29
49	0.34	X	0.22		刪除	
50	0.3	X	0.2		刪除	
51	0.41		0.32			30
52	0.2	X	0.15	X	刪除	
53	0.27	X	0.2		刪除	

(二) 程式設計態度量表

本研究採用戴文雄、游朝煌(1996)所開發之程式設計態度量表,本量表將程式設計態度分為:1.有用性。2.信心。3.喜歡。4.焦慮。5.課程必要性等五個向度,每個向度5題,共25題。其總量表及分量表之Cronbach α 係數如表4。

(三) 課程教材

1. 基礎程式設計課程
基礎程式設計以C++語言為教材內容,由國立臺中女中資訊科技教師設計12小時的教材,內容包括C++程式語言基本概念、流程控制及副程式等三部份。
2. 問題導向程式設計課程
本研究二種問題導向課程(問題導向程式設計教材及問題導向機器

人程式設計教材)皆以程式設計的基本概念來設計，以問題為核向，讓學生分組藉由教材的引導來解決問題。

程式設計學習成就進行共變數分析(MACOVA)，分析步驟如下：

(一) 迴歸線平行之檢定

由表 4-1 所示，不同教學策略分組之共變項與依變項之迴歸線平行檢定，均未達顯著水準，顯示各分組之共變項在依變項上具有均值性，各組迴歸線斜率相同，依變項受各個共變項影響情形沒有顯著不同。

肆、研究結果

一、學生學習成效之差異分析

針對不同教學策略分組的學習者，在

表 3-4 程式設計態度總量表與分量表之信度係數

分量表名稱	Cronbach α
喜歡	0.80
有用性	0.71
信心	0.75
焦慮	0.83
課程必要性	0.82
總量表	0.90

表 4-1 迴歸線平行檢定摘要表

考驗項目	Wilks' Λ	F 值
不同教學策略分組 x 程式設計成就前測	.971	.907
不同教學策略分組 x 程式設計態度前測	.998	.048
不同教學策略分組 x 邏輯思考能力	.932	2.306
不同教學策略分組 x 科技素養	.990	.321
不同教學策略分組 x 科學素養	.915	2.835

(二) 共同迴歸性斜率為零之檢定

由表 4-2 所示可知，邏輯思考能力、科技素養及科學素養在共同迴歸線斜率為零之檢定未達顯著，因此將之排除，僅以程式設計成就前測、程式設計態度前測為共變項進行共變數分析。

(三) 調整平均數

由於上述二項檢定符合共變數分析之假定，故進一步調整平均數後再進行共變數分析，經調整後之平均數如表 4-3：

(四) 單因子多變項共變數分析

由表 4-4 可知，不同教學策略分組的

學生在程式設計成效不具顯著差異(Wilks' $\Lambda=0.978$ ， $P>.05$)。

針對不同教學策略分組的學生在程式設計學習成效上前、後測進行 Paired T-Test 分析，結果顯示接受問題導向程式設計課程的控制組學生在程式設計學習成就及態度上前、後測並無顯著差異，而接受問題導向機器人程式設計課程的實驗組學生在程式設計學習成就前、後測並無顯著差異，但在學習態度前、後測達顯著差異(如表 4-5、4-6)。

表 4-2 共同迴歸線斜率檢定摘要表

考驗項目	Wilks' Λ	F 值
程式設計成就前測	.649	17.750*
程式設計態度前測	.661	17.841*
邏輯思考能力	.967	1.075
科技素養	.972	.887
科學素養	.964	1.137

表 4-3 調整平均數摘要表

教學策略	人數	程式設計成就		程式設計態度	
		平均數	調整平均數	平均數	調整平均數
機器人問題導向程式設計	37	18.41	17.566	86.3784	86.140
問題導向程式設計	35	16.54	17.431	83.6857	83.937

表 4-4 單因子多變項共變數分析摘要表

變異數來源	df	SSCP		多變量 Λ 值	單變量 F	
		成就	態度		成就	態度
常數						
組間 (排除共變數)	1	.280	4.573	.978	.40	1.446
		4.573	74.750			
共變數 (成就前測)	1	269.639	-130.058	.640*	38.491*	1.213
		-130.058	62.732			
共變數 (態度前測)	1	.399	30.464	.638*	.057	45.008*
		30.464	2326.966			
組內	68	37.269	276.437			
		276.437	2050.419			
總和	71					

表 4-5 問題導向程式設計課程 Paired T-Test 分析摘要表

		程式設計學習成效					
		成就			態度		
	人數	平均數	標準差	t 值	平均數	標準差	t 值
後測	35	16.53	3.791	1.983	83.6857	10.40297	1.897
前測	35	15.42	3.683		80.5429	13.01208	

表 4-6 問題導向機器人程式設計課程 Paired T-Test 分析摘要表

		程式設計學習成效					
		成就			態度		
	人數	平均數	標準差	t 值	平均數	標準差	t 值
後測	37	18.41	2.910	0.486	86.3784	8.02202	2.881*
前測	37	18.22	2.973		82.8108	8.30608	

二、半結構訪談資料質性分析

本研究於每週課程實施後隨機挑選實驗組學生進行約 30 鐘的半結構性訪談，以瞭解學生的學習狀況，學生半結構訪談資料分析如下：

T: Q1談談你現在對於C語言程式設計學習看法？

有人認為簡單，是基礎、實用的技能，也有人很排斥。大部分同學都覺得，雖然一開始不熟，但經過實際接觸以後，配合機器人的操作，能夠大致了解程式的使用，但在迴圈部份稍為複雜了一些，另外還有一些人不太了解程式的應用。

S: 一開始很不熟，可是越寫之後就越了解這個程式要怎麼用

S: 在上課的東西,未接觸實體，感覺這東西很遙遠，但接觸實作後就比較有感覺

S: 其實寫機器人的程式跟用C++，就有些指令方面會不一樣

S: 蠻有趣的，雖然不知道之後會延伸到什麼樣的東西，可是現在學的蠻有趣的

S: 覺得還沒有學的很深

S: 覺得是基礎

S: 覺得蠻簡單的，在回圈以後部分稍微複雜點

S1: 覺得工科都需要用到寫程式，上次到台大問一些理工科他們到底在做什麼，結論大部分都是在寫程

式，主要認為C語言程式是理工科的基礎，開始有點排斥語言程式，但接觸之後就沒那麼排斥了

S: 只有教你方法沒教你設計，所以會不太瞭解程式的應用

S: 覺得是實用技能

T: Q2從這2週的課程中，我們開始用C語言來控制機器人的作動，這對於你而言，有什麼看法與感想？

大部分人認為：程式輸入進去以後會跟想像的不太一樣，要慢慢修改才能達到想要的動作；也有人覺得實作比起電腦模擬來的有趣，感覺也比較有趣，也有學到東西，但也需要考量到機器人重量和行進方向因素。

S: 一開始的時候靠老師給的大綱跟設想的完全不一樣，最後要慢慢去改，在去找什麼地方要加什麼，過程中可以學到很多思考的問題

S: 我剛開始寫機器人的語言程式，就把C++那一套帶去運用，結果丟進去的時候，程式必須要改掉

S: 發現機器人可以不用很難的程式去寫

S: 就是C語言程式的運用，跟機器程式的運用，不太一樣，機器人的控制會多一些名詞，覺得讓機器人動就覺得蠻有趣

S2: 跟想像的不太一樣，因為程式是對的，動作就不是跟想像中一樣，需

作多次調整，才能達到跟自己預期的效果一樣

S: 主要比起模擬，實作是比較有趣

S: 跟之前電腦模擬的感覺不太一樣

S: 變的比較困難，需考慮到機器人重量，還有行進方向

T: Q3當老師給你機器人控制問題時，當下的思考是什麼？

找前面有做過的來參考，拼拼湊湊把他弄出來，或者是參考別人、請教別人，雖然程式很簡單，但卻很難控制，明明是照著想法寫的、大家討論的，也沒有寫錯，但卻沒有辦法照著我們所想的執行，而且落差很大。想像自己是一個機器人，程式要怎麼輸入才會照著我們的意思走。

S: 找看看符合前面有做過的東西，比較簡單的東西，然後就是把它組合起來，來完成這個東西

S: 參考別人或是請教別人

S: 如何組裝它

S1: 認為老師給的提示太多，因為我們自己想到別人沒想到的東西，老師都會說出來

T: 比較難的是什麼？

S1: 程式很簡單，但很難控制它，明明程式是照想法寫的，也照大家討論的，都沒有寫錯，但是不會照我們想像作動，且落差很大

S: 想像一個機器人，前進後退左轉右轉，想像程式輸入進去會怎麼動

T: Q4談談你對於解決問題的過程與完成問題後的感覺？

解決問題的過程中遇到很多問題，要不斷的試，當最後寫出來的程式跟完成的動作和想像的一樣，就會很有成就感、很開心。

S: 中間遇到很多問題，就試很多次，有可能試很多次，也還沒達到你要的結果，可是當最後做出來的時候真的很開心

S: 很開心，很有成就

S: 過程中檢查不足

S1: 最後寫的程式跟完成的動作跟想像的一樣，就很開心

T: 一開始應該有遇到困難的問題吧？

S: 一開始偏了之後要校正扭回來給的力量太大，每次過頭，然後又會亂掉，然後就開始亂跑，後來經由調整，就有跑的比較順

T: 你經由一再修改，會覺得算了嗎？

S: 一定一把它做好

S: 主要問題都在於組裝

T: 問題的思考方式是如何下手？

S: 先畫成表格，問題發生後且不斷的做試驗

S: 先想機器人前進後退，把整個流程想出來，然後再做細部調整

T: Q5在使用C語言寫控制程式的過程中，你最在意(重要)的是什麼？

有人認為最重要的是要先找出大方向、程式的邏輯要對、程式要設計完善、了解程式代表的意義、精簡化程式的編寫、車子能不能達成我們要的目標。

S: 需要了解每一個動作

S: 程式的精簡化

S: 簡單明了解決，做成精簡化

S: 外在的故障排除，及實驗問題

S: 主要邏輯要對，程式要寫的清，結論為程式需設計完善，車子所做出的動作能不能達成目標？

S: 先找出大方向

T: Q6到現在為止，你認為程式設計對與你來講，是[非常有興趣、有興趣、還好、沒興趣、極沒興趣]這五個選項請選一個，並說明

決大多數人都覺得有興趣，因為覺得它有趣，可以照著你寫出來的指令做動作就覺得很開心、也有人覺得能成功把程式寫出來，順利執行讓自己覺得有成就感。也有人認為說程式設計是基礎、對自己以後有幫助，所以盡量讓自己喜歡，或者是不知道程式設計是要做什麼。還有人覺得沒什麼興趣，因為不是自己擅長的部份，也有人喜歡組裝，因為認為組裝很簡單。

S: 有興趣；因為寫出來他就可以照你的指令跑，看到它動就很開心

S: 還好；因為不是很擅長的東西，就覺得還好

S: 有興趣；因為是基礎，即時沒興趣也會強迫有興趣

S: 有興趣；因為有趣

S: 還好；單純寫程式，不知道他要做什麼

S: 有興趣；還好中間，第一堂課在工廠裡，感覺是不屬於我的世界，但學了之後，在自己下去寫，後來還有寫出來，所以就較有成就感

S: 非常有興趣

S: 還好

S: 沒什麼興趣；主要以組裝為主，覺得組裝很簡單

S: 本來就有興趣

S: 有興趣；能夠成功將程式寫出來，機器人也順利執行，就會很有成就感

S: 有興趣；程式設計是假想邏輯，覺得對以後有幫助

伍、結論與建議

一、結論

資訊科技的進展日新月異，現今各種行業的發展皆需應用到資訊技術，程式設計課程已成為各科系所需修習的科目，也是學生必備的基本能力之一，目前部分中小學與高中職學校都有基礎程式設計課程。

Papert 在他的經典之作 *Mindstorms: Children, Computers & Powerful Ideas*(1993)一書中指出，程式設計不過是人們透過人類和電腦雙方都「懂」的一種語言和電腦溝通而已，而語言的學習正是孩子們最擅長的一件事。他表示：「既然每個正常的孩子都學習說話，為什麼不讓他們學習和電腦說話？」(Papert, 1993)，Ellinger(2003)力主程式設計應成為每個孩子必備的基本能力，他列舉了程式設計所能帶來的各種認知與技能上的好處，包括提升創造力、設計能力、組織能力、溝通能力、預判與分析錯誤的能力、精確性等。然而程式設計在中小學階段卻是一項難教也難學的科目，初學者學習程式設計的主要困難在於無法瞭解程式執行的過程及無法瞭解電腦所允許的描述方法，如能在程式設計教學中整合微世界與視覺化觀念，將有助於學生學習程式設計，LEGO Mindstorms 就是一套能夠應用圖形化程式設計的科技創新積木玩具，應用於教學中可增加學習者程式設計學習的趣味性和應用性，同時降低初學者學習程式設計的障礙與恐懼。本研究即將 LEGO Mindstorms 融入高中程式設計教學中以瞭解機器人融入高中程式設計課程中，對女高中生在學習程式設計成就及態度之影響，經過 12 週的教學實驗後本研究獲得下列結論：

1. 不同教學策略分組的學生其程式設計學習成效並沒有顯著差異，此結果與文獻許多研究結果不同，探究其原因可能為二種教學策略分組學生皆以程式設計問題導向為教材之內

容，學生進行問題導向教材時於程式設計成效上並沒有太明顯的差異。

2. 學生在接受完問題導向機器人程式設計課程後對於程式設計的學習態度有顯著提昇，此結果與文獻中許多研究者使用 LEGO Mindstorms 在國、高中進行程式設計教學實驗，均大致肯定了 LEGO Mindstorms 對於提升程式設計學習動機與成效之助益相同。(Klassner, 2002;蔡學偉, 2004;許雅慧, 2005;曾義智, 2006;謝亞錚, 2009)
3. 學生覺得對程式設計雖然一開始不熟，但經過實際接觸以後，配合機器人的操作，能夠大致了解程式的使用，並覺得利用 LEGO Mindstorms 實作比單純使用電腦模擬來學習程式設計有趣。

二、建議

本研究僅於女高中學校進行實驗教學，由研究發現機器人程式設計課程較能引起女高中學生學習程式設計的興趣並提昇學習的態度，但不瞭解男生是否有相同的效益，因此、建議可將性別列為依變項，研究機器人程式設計課程對高中學生學習程式設計之影響；另外基於時間考量，本研究之問題導向程式設計課程僅實施六週(12 節)，可能較無法讓問題導向教材發揮功能，建議可以讓問題導向程式設計課程實施時間增加，可瞭解問題導向教材對程式設計教學之影響。

參考文獻

- 計惠卿、張杏妃(2001)。全方位的學習策略—問題導向學習的教學設計模式。教學科技與媒體, 55, 58-71。
- 高廣孚(1984)。杜威教育思想。台北：水牛。
- 張民杰(2003)。超學科統整模式之一—問題導向學習在國中九年一貫課程的設計與實施。新竹師院學報, 17, 389-424。
- 張瓊穗、賴亦璇、王尹伶、甘珮禎(2002)。問題導向學習模式在國小環境議題教學之應用。教育資料與圖書館學, 40(2), 198-209。
- 許雅慧(2005)。應用 LEGO Mindstorms 視覺化環境輔助程式設計觀念學習。國立臺灣師範大學碩士論文, 未出版, 臺北市。
- 陳志豪(2004)。高二學生對科學的態度與科學相關行為意向之研究。私立東海大學教育研究所未出版之碩士論文。
- 陳怡伶(2009)。程式設計概念之學習基於 LEGO MINDSTORMS 機器人。國立高雄應用科技大學碩士論文, 未出版：高雄市。
- 陳毓凱(2005)。「問題本位學習」教學模式對國中二年級學生自我導向科學學習傾向與科學學習動機之影響。國立中原大學碩士論文。未出版。桃園。
- 曾義智(2006)。應用機器人於程式設計教學—實體機器人與模擬軟體使用成效比較。國立臺灣師範大學碩士論文, 未出版, 臺北市。
- 劉忠昱(2012)。高雄地區問題導向學習對國中生數學學習成效之研究。國立高雄師範大學碩士論文, 未出版, 高雄市。
- 蔡學偉(2004)。問題導向學習於網路輔助電腦樂高課程之研究。國立臺灣師範大學碩士論文, 未出版, 臺北市。
- 蕭佳明、黃瑛綺(2012)。樂高機器人應用於科學與創意教育市場創業之研究, 遠東學報, 29(3), 375-386。
- 謝亞錚(2009)。機器人輔助程式設計學習之學習成效與學生心智模型探討。國立臺灣師範大學碩士論文, 未出版：台北市。
- Ayers, J. B., & Price, C. O. (1975). Children attitudes toward Science. *School Science and Mathematics*, 75(4), 311-318.
- Barrows, H. S. (1992). *The tutorial process*. Springfield, IL:Southern Illinois University School of Medicine.
- Cañas J. J., Bajo M. T., Gonzalov P. (1994). Mental models and computer programming. *International Journal of Human-Computer Studies*, 40(5), 795-811.
- Chase, C. I. (1978). *Measurement for educational evaluation*. (2nd ed.). Reading, Massachusetts: Addison-Wesley.
- Costelloe, E. (2004). *Teaching Programming The State of the Art*. CRITE Technical Report.
- Dagdilelis, V., Sartatzemi, M., & Kagani, K. (2005). Teaching (with) robots in secondary schools: some new and not-so-new pedagogical problems. *Proceedings of Fifth IEEE International Conference on Advanced Learning Technologies*. 757-761.
- Dawson, C. (2000). Upper primary boys' and girls' interests in science: have they changed since 1980? *International Journal of Science Education*, 22(6), 557-570.
- Duke, R., Salzman, E., Burmeister, J., Poon, J., & Murray, L. (2000) Teaching programming to beginners - choosing the language is just the first step. *Proceedings of the Australasian conference on Computing education*, 79-86. Melbourne, Australia.
- Edens, K. M. (2000). Preparing problem solvers for the 21st century through problem based learning. *College Teaching*, 48(2), 55-60.
- Ellinger, H. (2003). A natural sense of algorithm: Children should learn computer programming as a basic skill. Retrieved December 18, 2007,

- from <http://hunter.ellinger.org/education/ProgrammingForChildren.htm>
- Fagin, B. S., Merkle, L. D., & Eggers, T. W. (2001). Teaching computer science with robotics using Ada/Mindstorms 2.0. *ACM SIGAda Ada Letters*, XXI(4), 73-78.
- Fleming, M. L., & Malone, M. R. (1983). The relationship of student characteristic and student performance in science as view by meta-analysis research. *Journal of Research in Science Teaching*, 120(5), 481-495.
- Gardner, P. L. (1975). Attitudes to science: A review. *Students in Science Education*, 12(1), 41.
- Haladyna, T., Olsen, R., & Shaunghnessy, J. (1982). Relationships of students, teacher, and learning environment variables to attitudes towards science. *Science Education*, 66(5), 671-687.
- Hartly, H., & Beall, D. (1984). Toward the development of children's science curiosity measure. *Journal of Research in Science Teaching*, 21(4), 425-436.
- Jorgen, L. & Thomas, H. (2007). Does lego training stimulate pupils' ability to solve logical problems? *Computers & Education*(49), 1097 - 1111.
- Klassner, F. (2002). A case study of LEGO Mindstorms™ suitability for artificial intelligence and robotics courses at the college level, *Technical Symposium on Computer Science Education Proceedings of the 33rd SIGCSE technical symposium on Computer science education*, 8-12.
- Kelly, A. (1986). The development of girl's and boy's attitude to science A longitudinal study. *European Journal of Science Education*, 8(4), 399-412.
- Leeves, J. P. (1975). The home, the school and achievement in mathematics and science. *Science Education*, 59(4), 439-460.
- Linn, M. C., & Clancy, M. J. (1992). The case for case studies of programming problems. *Communications of the ACM*, 35(3), 121-132.
- Mannila, L., Peltomaki, M., & Salakoski, T. (2006) What about a simple language? Analyzing the difficulties in learning to program. *Computer science education*, 16(3), 211-227.
- McNerney, T. S. (2004). From turtles to tangible programming bricks: explorations in physical language design. *Personal and Ubiquitous Computing*, 8(5), 326-337.
- Olstad, R. G., & Haury. D. L (1984). A summary of research in science education. 1982. *Science Education*, 68, 3.
- Papert, S. (1993). *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books, 2nd edition.
- Power, C. (1981). Changes in student's attitudes toward, science in the transition between Australian elementary and secondary schools. *Journal of Research in Science Teaching*, 18(1), 33-39.
- Reid, D. J., & Tracey, D. C. (1985). The evaluation of a school science syllabus through objectives and attitudes. *European Journal of Science Education*, 7(4), 375-386.
- Schibeci, R. A., & Railey, J. P. (1986). Influence of student's background and perceptions on science attitudes and achievement. *Journal of Research in Science Training*, 23(3), 177-188.
- Shrigley, R. L. (1972). Sex differences and its implication attitudes and achievement in elementary school science. *School Science and Mathematics*, 72(9), 789-793.
- Smith, D. C., Cypher, A., & Tesler, L. (2000). Novice programming comes of age. *Communications of the ACM*, 43(3), 75-81.
- Soloway, E. (1986). Learning to program = Learning to construct mechanisms and explanations. *Communications of the ACM*, 29(9), 850-858.

Valente, J. A. (1995). Logo as a Window into the Mind. Logo Update, 4(1). Retrieved November 20, 2007, from <http://el.media.mit.edu/logo-foundation/pubs/logoupdate/>

Yager, R. E., & Yager, S. O. (1985). Changes in perceptions of science for third seventh, and eleventh grade students. Journal of Research in Science Teaching 22(4), 347-358.

Yager, R. E., & Penick, J. E. (1986). Perceptions of four age groups toward science classes, teachers, and the value of science. Science Education.

投稿日期 : 101 年 09 月 28 日

接受日期 : 101 年 10 月 31 日

The Effectiveness of Robot Programming Courses in PBL Programming Language Course for Female High school students

Yu-Te Wang*, Yuan-Tai Chen, and Ling-Huei Tseng

National Taichung Girls' Senior High School

Abstract

The purpose of the study was to understand the effectiveness of robot programming courses in programming language course for female high school students. The samples of this study were female high school students of a high school in the center of Taiwan. During the experimental period, two class of students were randomly chose, control group (42 students) was one class and experimental group (42 students) was the other. Students of two classes to be implemented basis programming language course for six weeks. All students implemented programming achievement and attitude pretest after taking basis programming language course. The control group implemented PBL programming language course for six weeks (12 hours). The experimental group implemented PBL robot programming courses for six weeks (12 hours). During the experimental period, randomly selected students of the experimental group did the semi-structured interview. All students implemented programming achievement and attitude pretest after taking PBL programming language course and PBL robot programming courses. The main statistic procedures employed for analyzing experimental data and testing the research hypotheses were MACOVA and Paired T-Test. The main findings of this study were that there were no significant differences in students' programming achievement between experimental group and control group. Students' programming attitude of the experimental group was significantly different to the experimental group. Data of semi-structured interview by qualitative analysis indicate that robot programming courses can generate students' motivation and interest.

Keyword: Robot, Programming Language, PBL(Problem-based Learning), Female High school students.