

【2019 全國科學探究競賽-這樣教我就懂】

國中組 成果報告表單

題目名稱： 推箱不迷路，麋鹿來推箱~經典推箱遊戲解題策略探究與 App 研發
摘要： <p>我們研究以 MIT App Inventer 2 積木式程式語言實作「推箱子」遊戲 App。在探究的過程中我們經由問題拆解找到對應的資料結構，加上死結型態識別，並設計成輔助解題的程式，提供初學者迴避死結的解題邏輯，幫助初學者建立解題策略。我們也建立了題庫，在選單中分成 1.經典、2.黑洞+連推、3.滑冰等 3 種變形規則。遊戲的操作介面讓玩家可在平板螢幕上操控遊戲主角“麋鹿”在關卡中移動並推動箱子達成任務，並有推箱計步器、死結提示輔助。可自行啟用輔助解題功能。遊戲 App 設計完成後，找到 20 名遊戲初學者給予相同的規則解說及操作測驗，發現經由本輔助系統的訓練，由原先總平均 54 秒/題，後來總平均 33 秒/題，所花時間平均每題減少 21 秒。短期內解題速度便提升 64%。</p>
探究題目與動機： <p>近幾年來各種桌遊與益智手遊大肆流行得到大眾的喜愛。我們在學校社團曾接觸到迷宮類型的遊戲，此次要探究的「推箱子」為其類似的形式，關卡的難度頗高、破解成就感滿溢，組員們幾經討論於是決定來探究如何實作這一個益智遊戲 APP。另外我們也想試試藉由實作 App 來探索程式輔助解題，進而尋找自動出題的可行性。</p> <p>「推箱子」開始畫面會有一個玩家(麋鹿)、數個箱子、及數個目標放置點。遊戲要求玩家在二維地圖的活動範圍將所有箱子推到目標放置點，其中牽涉到大量的空間邏輯推理。活動空間會有牆壁形同迷宮一樣的限制，遊戲的原生條件是：</p> <ol style="list-style-type: none">1.不能用拉箱子的方式移動。2.不可同時推動兩個或以上並排的箱子。3.不可將箱子推入牆角或貼至牆面，否則會使的箱子無法推出而使解題失敗。 <p>箱子未送達目標且無法推動的情形稱為死結，不同的型態的死結需要逐一分析並且避開。長寬範圍、箱子數目，或牆壁限制的增加，都會使解題的難度增加，更不易找出最少搬運次數的最佳解。盤面可能的組合情形在數學上被證明是 PSPACE-完全的，已經接進圍棋的複雜度，不再適合全面的暴力搜尋。因此調查死結、做反向路徑搜尋的確為適合的解題策略。</p>
探究目的與假設： <ol style="list-style-type: none">一、 研究遊戲策略、分析探討關卡之資料結構與可行路徑的搜尋策略。二、 以 MIT App Inventer 2 實作「推箱子」遊戲 APP 之研究開發。<ol style="list-style-type: none">(一)、 實作整合控制介面，建立經典及另外 2 種變形規則的關卡，用以驗證各種推論策略的可行性。(二)、 調查死結實作解題輔助功能，協助玩家學習解題策略。三、 假設解題輔助可提升初學者成功解題的速率並驗證。

探究方法與驗證步驟

一、主要介面配置說明：

「麋鹿推箱子」App 的遊戲頁面，如圖 1-1。主要的顯示介面為關卡地圖右下方為方向控制鍵用來操作主角移動。棋盤左上角有「結束返回」按鈕，關卡地圖上方顯示有關卡題號，而關卡地圖的上方可看到給玩家作為提示的「推箱計步器」，圖 1-2 為遊戲模式的選單頁面，可重頭開始逐題破解亦可選擇挑戰特定關卡。關卡的右上方有提供「死結提示」按鈕，按下提示按鈕後會顯示為如圖 1-3 的死結提示地圖。1-4 為 a 經典、b 黑洞+連推、c 滑冰 等 3 種不同規則的類別。



圖 1-1



圖 1-2

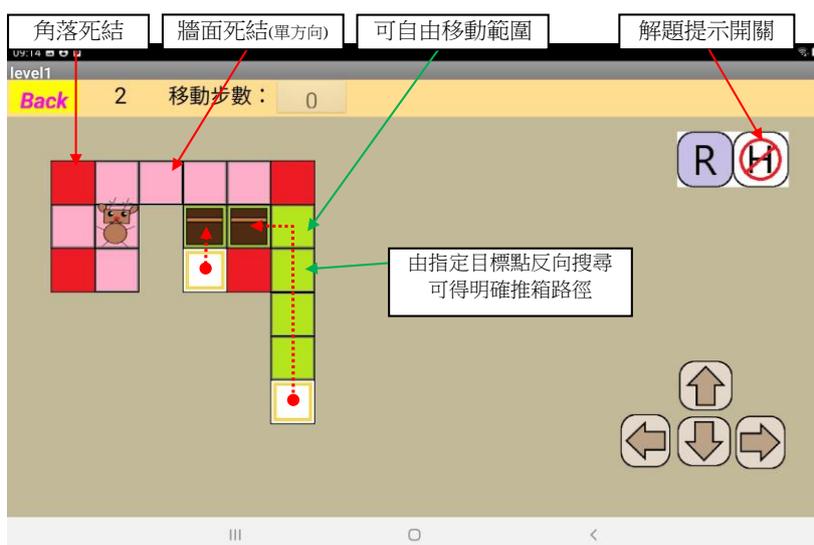


圖 1-3

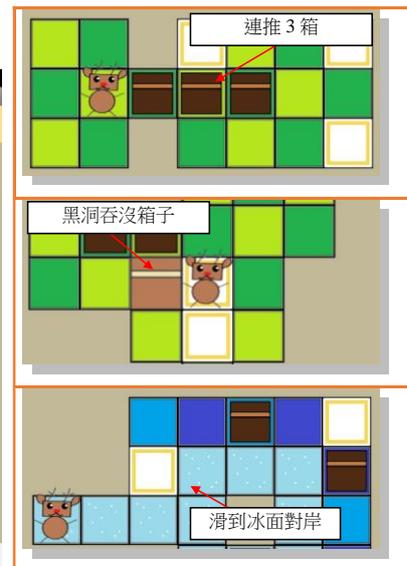


圖 1-4

二、重點程式積木塊分析及功能說明：

A. 初始化各類常數清單：

1 初始關卡清單：如圖 2，為關卡及格中物狀態的模型，原為一維清單，用以模擬 Warren Smith 棋盤模型的實作於此用二維說明。此為本程式主要的資料結構。

外圍的黃色方框處為棋盤邊界以代號 0 標記。中央 8X10 為各關卡設計的最大活動範圍，以 0 標記非活動區；以左右 2 位字碼標記活動區，

左字碼代表格中物：p 代表**玩家(麋鹿)**，b 代表**箱子**，x 代表**空位**，h 代表**黑洞**位置。

右字碼代表地板類別：1~4 為正常地板的不同配色用以區別適用不同的遊戲規則，

5 代表**黑洞(會吃掉箱子)**，6 代表**冰面(會一路滑行到冰面對岸)**，7 代表**牆面死結**，

8 代表**角落死結** 9 代表箱子的**放置目標區**。

	A	B	C	D	E	F	G	H	I	J
1	0	0	0	0	0	0	0	0	0	0
2	0	x1	x2	x1	x2	x1	x2	0	0	0
3	0	x2	p1	0	b1	b2	x1	0	0	0
4	0	x1	x2	0	x9	x1	x2	0	0	0
5	0	0	0	0	0	0	x1	0	0	0
6	0	0	0	0	0	0	x2	0	0	0
7	0	0	0	0	0	0	x9	0	0	0
8	0	0	0	0	0	0	0	0	0	0

或

	A	B	C	D	E	F	G	H	I	J
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	x3	x4	b3	x4	x9	0	0
3	0	0	0	x9	x6	x6	x6	b4	0	0
4	0	p6	x6	x6	x6	0	x6	x3	0	0
5	0	0	0	0	x6	b6	x6	x4	0	0
6	0	0	0	0	0	x3	x4	x9	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0

圖 2-1 原關卡地圖

	A	B	C	D	E	F	G	H	I	J
1	0	0	0	0	0	0	0	0	0	0
2	0	x8	x7	x7	x7	x7	x8	0	0	0
3	0	x7	p7	0	b1	b1	x1	0	0	0
4	0	x8	x7	0	x9	x8	x1	0	0	0
5	0	0	0	0	0	0	x1	0	0	0
6	0	0	0	0	0	0	x1	0	0	0
7	0	0	0	0	0	0	x9	0	0	0
8	0	0	0	0	0	0	0	0	0	0

或

	A	B	C	D	E	F	G	H	I	J
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	x8	x7	b3	x7	x9	0	0
3	0	0	0	x9	x6	x6	x6	b7	0	0
4	0	p8	x8	x8	x7	0	x6	x7	0	0
5	0	0	0	0	x8	b6	x6	x7	0	0
6	0	0	0	0	x8	x7	x9	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0

圖 2-2 提示關卡地圖

解題輔助系統本程式的一大特色，主要根據死結的辨識，並用**紅色地板**標註**角落死結**，用**粉色地板**標註**牆面死結**。因為若玩家將箱子推到角落便再也無法將箱子推出，於是無法完成解題。而推入牆面死結位置之後，只能在沿著牆面方向(上下或左右)來回推移，再也無法脫離牆面。若將這 2 種死結狀態標註清楚，再配合由終點反向搜尋至箱子處，便可定出箱子移動的路徑，而玩家只要再注意一下轉彎時的變換位置便可輕鬆解題，對於解題策略的建立十分的有幫助。解題輔助系統設計由一個開關切換，並且叫用**正常關卡地圖**亦或叫用**提示關卡地圖**，由玩家個人所需選用。若確定落入死結狀態可以單關重玩。

2 地板元件清單：(圖略)對應於各圖格的圖像精靈元件，用於提供顯示行動範圍的格子以確認玩家(麋鹿)和箱子的位置，並用圖片顯示目前格子中的物件。

B. 重要副程式：

- 「關卡開局」副程式：如圖 3，設定關卡地圖及玩家、箱子及終點目標位置的初始狀態。

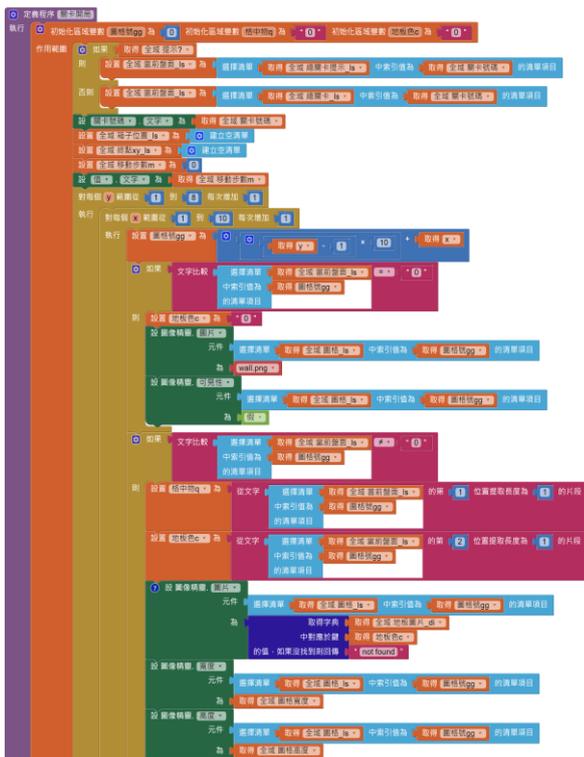


圖 3-1「關卡開局」副程式

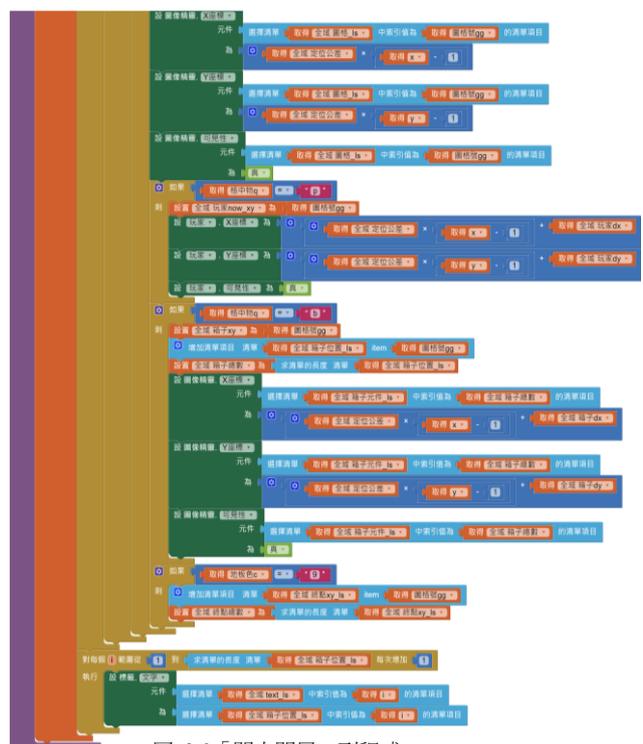


圖 3-2「關卡開局」副程式

- 「地板初始化」副程式：如圖 4，「關卡開局」會同時叫用，將地板圖片設置完成。
- 「圖控初始化」副程式：如圖 5，是將四個方向鍵及 R 鍵、H 鍵的圖控元件設定完成。

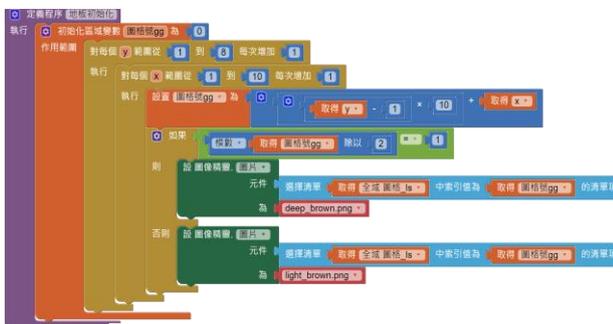


圖 4「地板初始化」副程式

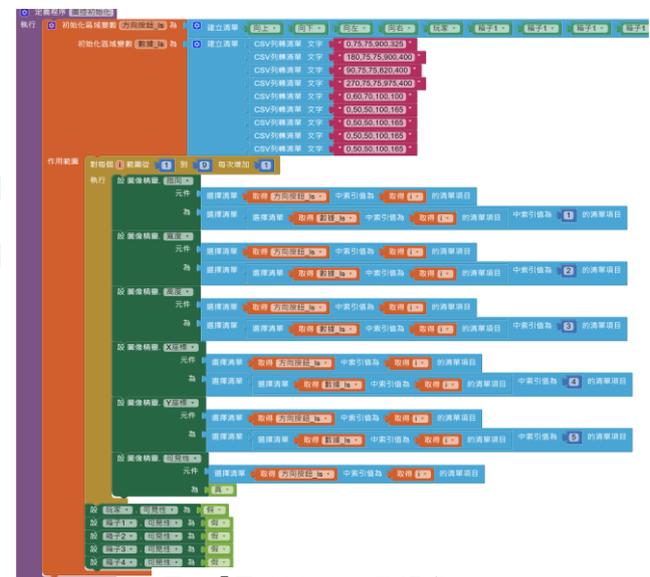


圖 5「圖控初始化」副程式

- 四個「方向鍵」副程式：如圖 6，每當玩家按下一個方向按鈕，「方向鍵」副程式就必須建立玩家前進方向前方 4 格地板及格中物的資料，用以判定是否需要推箱，判定推箱數量是否超過規定，判定最前面的箱子是否落入黑洞，若無推箱也需判定玩家是否可以往前移動。並且將盤面的所有變化更新記錄下來，並且更新移動步數。最後還要判定是否可以完成任務過關。

圖 6-1 四個「方向鍵」副程式

圖 6-2 四個「方向鍵」副程式

- 「合法移動判斷」副程式：如圖 7，判定是否為合乎規則的玩家移動與推箱動作(又分一次 1 箱或連推 3 箱)。

圖 7-1 「合法移動判斷 (一次 1 箱)」副程式

圖 7-2 b 「合法移動判斷 (連推 3 箱)」副程式-部分程式

- 「玩家移動_盤面修正」副程式：如圖 8，當玩家移動時，就會藉由這個副程式將關卡盤面資料更新。
- 「過關判定」副程式：如圖 9，當所有的箱子、人物的移動完成後便依此副程式判定是否過關，若是便啟動新的關卡。



圖 8 「玩家移動_盤面修正」副程式

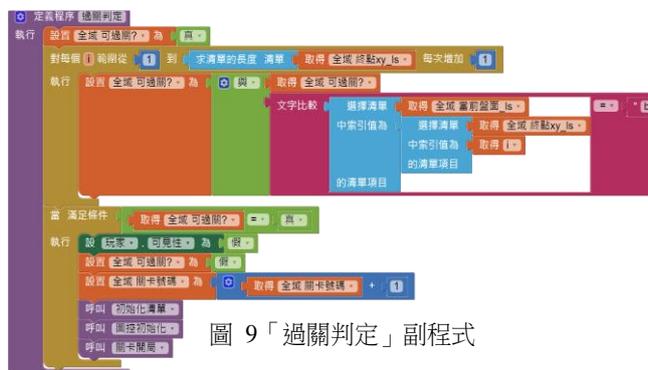


圖 9 「過關判定」副程式

二、操作實驗：假設解題輔助可提升初學者成功解題的速率

實驗：程式設計完成後，經由 20 位同學參與實驗，由系統中的挑選同樣的 3 個關卡，每個關卡玩 3 次，第 1 次使用正常地圖，第 2 次使用死結提示輔助，第 3 次用再正常地圖，每次記錄解題所需時間，最後比較第 3 次和第 1 次的成績。

結果：經由本輔助系統的訓練，第 1 次總平均 54 秒/題，第 3 次總平均 33 秒/題，所花時間平均每題減少 21 秒。解題速度提升 64%。

結論與生活應用

- 一、我們的題庫共有 a 經典、b 黑洞+連推、c 滑冰 等 3 種不同規則的類別，可重頭開始逐題破解亦可選擇挑戰特定關卡，可幫助不同程度初學者無痛學習解題策略。
- 二、我們的 App 可降低解題難度達到寓教於樂的功效，透過輔助解題程式協助初學者建立解題策略，短期內便可表現出解題能力變好(減少死結發生)，解題速度提升的進步。
- 三、MIT App Inventor 2 積木式程式語言適合無程式碼編寫基礎的初學者，經由我們的介紹，同學們也可開始學習自製 App，並且可以從中學習運算思維。
- 四、自行開發 App 充滿樂趣又經濟實惠，未來還可嘗試將更高階的技術加入實作。

參考資料

1. 科技大觀園。進擊的全民寫程式！你今天 coding 了嗎 <https://scitechvista.nat.gov.tw/c/sZLK.htm>
2. 科技大觀園。一小時寫程式！下一個全民必修的第二外語 <https://www.gvm.com.tw/article.html?id=32001>
3. 科技大觀園。電腦如何下象棋 <https://scitechvista.nat.gov.tw/c/sTkf.htm>
4. 推箱子介紹。取自 <https://zh.wikipedia.org/wiki/%E5%80%89%E5%BA%AB%E7%95%AA>
5. 蔡宜坦(2018)。《App Inventor 2 互動範例教本 第三版》。旗標科技。