

市本資訊科技領域教學綱要 國小組

程式、設計、思維

運算思維小組





運算思維課程內涵

程式設計課程的學習內涵，應區分為運算思維與設計思維兩個部分，運算思維：是一種解決問題的系統化方法，包含四個解決問題的步驟：

1. 理解問題並嘗試將問題拆解成容易操作的步驟
(培養解構能力，以小御大)
2. 覺察問題出現的樣式，找出規律
(培養洞察力與歸納能力，一葉知秋)
3. 將問題虛擬化並塑造成數據模型
(運用數位素養與學科知識來表徵問題，深入淺出)
4. 設計演算法解決問題
(找出系統化且通用的方法解決相同類型問題，見招拆招)

以小御大

培養解構能力：

- 將數據縮小，讓問題變容易。
- 找出問題的所有變因，並逐一想辦法控制，讓問題變單純。
- 運用演繹邏輯思考，依問題的先後順序細步化，每次只解決一小部份。將問題程序化。
- 運用歸納邏輯思考，從其他領域的類似問題導入解法。旁敲側擊法。

一葉知秋

培養洞察力與歸納能力：

- 觀察問題出現的時機點是否有規律性。
- 觀察伴隨現象或數據是否有規律性。
- 觀察處理程序是否有規律性。
- 觀察規律性是否有遞移或轉換現象，找出通用法則。



深入淺出

運用數位素養與學科知識來表徵問題：

- 使用圖像或圖表來表徵問題(例如:捷運路線圖、心智圖)
- 使用符號來表徵問題(例如: $K=(P+I)*S$) 知識=(內在經驗+外在資訊)×分享
- 選擇適當的資料結構來建立數學模型(例如:用陣列表示迷宮)
- 運用資料流程圖,解釋問題的成因(例如:電網效率分析)
- 運用程序流程圖,構想解決問題的程序(例如:SOP)



見招拆招

找出系統化且通用的方法解決相同類型問題：

- 設計處理程序，能根據數據變化提供預測或判讀資訊。
- 檢查處理程序是否能處理例外情形。
- 將不同問題解決流程整合成一體。
- 將處理程序模組化，以便反覆利用。

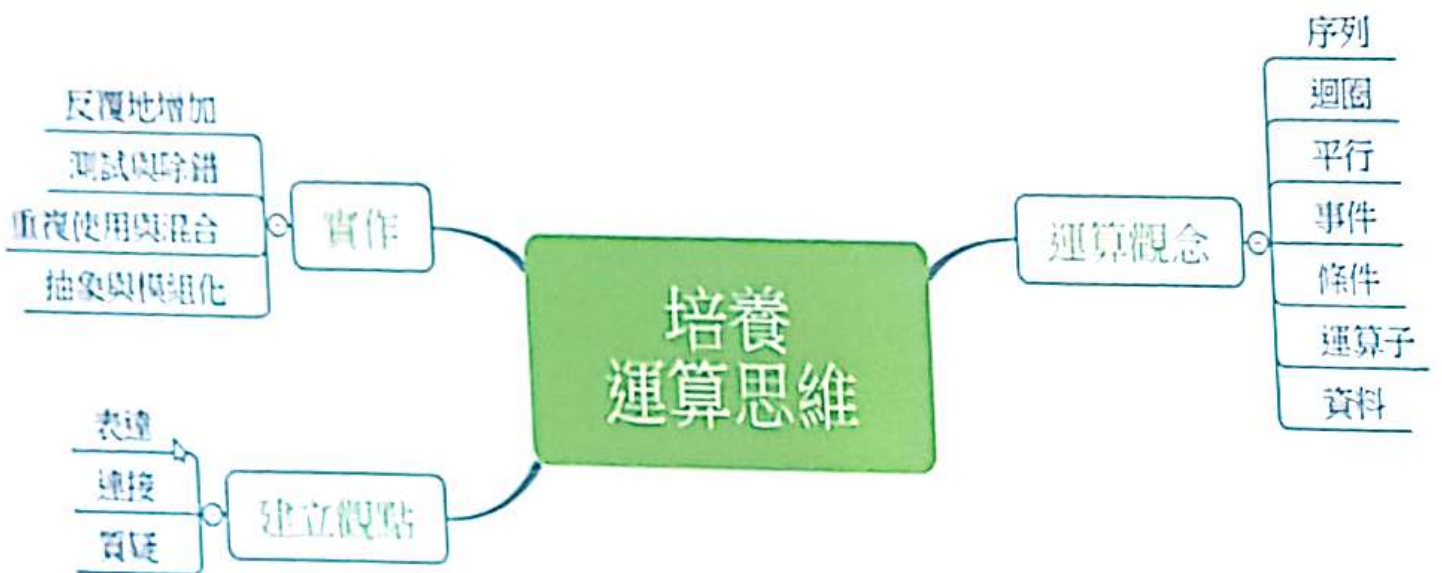


運算思維的先備能力

- 閱讀理解能力(語文)
- 符號運算能力(數學)
- 邏輯組織與分析能力(哲學)
- 數位素養(非必備, 資訊)

學生須具備以上基本能力, 才能掌握運算思維。美國教育界普遍認為國小四年級就可以在學習情境中運用運算思維。四年級以下可以先培養學習興趣並練習單項能力, 例如: 樣式識別。

皮亞傑認知發展論認為7~11歲為具體運思期, 11~16歲為形式運思期。MIT早期教育發展中心研究發現孩童從2歲可以辨識顏色後, 就具備了基本的邏輯推理能力, 例如: 分類、找相似、找不同。



什麼是設計思維？

- 設計的設計
- 關於設計的思考
- 以人類思考模式為基礎的設計（而非以電腦運算的模式）
- 基於人性的創意發想

為什麼要強調設計思維？

- 程式設計並非只是寫程式 (Coding) , 也要學習設計原理和方法 (Programing)
- 掌握運算思維幫助我們解決問題, 掌握設計思維則幫助我們落實創意
- 創客 + 設計思維 = 文創產業



設計思維課程內涵

程式設計課程的學習內涵，至少應包含運算思維與設計思維，設計思維：是一種創意發想的系統化方法，包含四個執行專案的步驟：

- 了解並分析使用對象的需求
(培養敏察能力, 將心比心)
- 透過設計提供感動的使用者經驗
(培養美感與表達能力, 人因工程)
- 測試與除錯, 展示與溝通, 並持續改善
(培養溝通技巧與負責的態度, 誠意行銷)
- 從使用者的回饋進行評估與決策
(培養社會責任、永續環保、價值取向等信念, 創造價值)

學習目標

- 程式設計的學習，重點在設計方法的學習，而非程式語言的學習
- 運算思維可以遷移，而程式語言總會消失
- 程式設計學習的目標，應該是透過寫程式，學習如何透過設計去解決問題，然後把解決問題的經驗內化成運算思維

教學設計原則

- 運算思維是拿來用的, 不是拿來教的
- 教師應設計學習活動讓學生練習使用運算思維, 而不要去解釋它
- 教師應運用運算思維於課程活動設計中, 將複雜問題拆解成簡單的討論程序



- **程式神秘箱**

學生透過作品的執行體驗，在看不到程式碼的情形下，預測程式內容並試做驗證。

- **大家來找碴**

老師提供有邏輯錯誤的程式碼，並說明程式的目的，學生必須發現錯誤並嘗試修正，讓程式可以如預期運作。

- **祕技與外掛**

學生從現成的範例遊戲出發，透過改變難度、修改操作方式、增加關卡...等方式設計成更好玩的遊戲。

- **異想天開**

就老師給定的主題，學生自由發想故事情節，並設計成舞台劇動畫。

- **黑客松**

在限制規則內學生進行分組競賽，限制規則可包含：時間限制、積木限制、角色限制、輸入裝置限制...等。



教學綱要

學習階段	編號	學習內容	建議實施節數
低年級	TP-I-1	序列式動畫設計	4
	TP-I-2	使用案例圖	4
中年級	TP-II-1	迷宮解題	4
	TP-II-2	物件導向概念	2
	TP-II-3	連鎖式動畫設計	8
	TP-II-4	壓力激盪會議	2
	TP-II-5	3D建模浮雕設計	4

課程規劃

學習內容: 序列式動畫設計

教學資源: 紙卡、平板(可使用之APP: Scratch Jr、Daisy the Dino、The Foos...)

課程規劃:

此課程內容可以使用平板或紙卡進行(可參考 code.org 紙本教材或自行設計), 學生透過前進、後退、左轉、右轉、跳躍.....等簡單指令(不帶參數的指令), 串接成序列, 以進行解題或動畫設計, 剛開始先使用單一角色, 若學生程度較好, 可以導入多個角色, 學習多線式動畫設計, 例如: 舞會動畫設計。



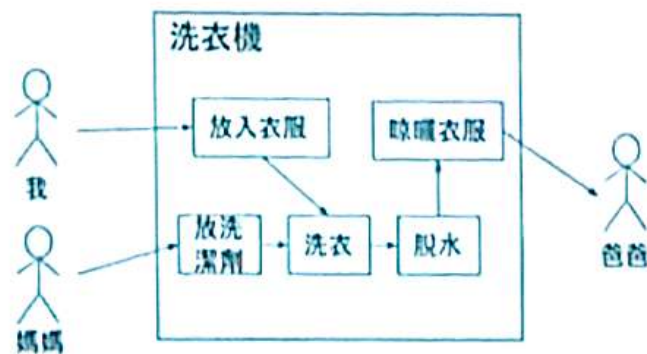
課程規劃

學習內容:使用案例圖

教學資源:紙筆、平板(心智圖軟體)或電腦(自由軟體 Dia、雲端服務 Cacao)

課程規劃:

此課程可使用UML軟體(如: Dia、Cacao...)或紙本手繪方式進行。學生先練習將生活中常用的科技產品及設備,繪製成使用案例圖,藉以認識人機介面設計中,參與者、動作、處理及回饋等要素。發展活動可以請學生設計幻想中的科技產品,藉由使用案例圖表達該產品的功能與使用的便利性。



課程規劃

學習內容 3D建模浮雕設計

教學資源 3D建模軟體或網站(www.tinkercad.com)

課程規劃:

3D建模依圖難易度區分,約略可分為:浮雕、立體單一形體、立體組合形體。此階段應學習如何建立2.5D浮雕模型,例如:名牌、杯墊、鑰匙圈...等,表現技法可以是陰刻或陽刻。若有輸出設備,應讓學生輸出作品以獲得動手做的成就感。

