

教育研究集刊
第六十六輯第一期 2020年3月 頁77-102

十二年國教資訊科技科目 學習次概念之探究

胡秋帆、王恩慈、吳正己、林育慈



摘要

十二年國教課綱增設科技領域資訊科技科目，該科學習內容包括「演算法」、「程式設計」、「系統平台」、「資料表示、處理及分析」、「資訊科技應用」及「資訊科技與人類社會」等類別，各類別下分別列出相關之學習概念，但所列出之學習概念範圍較大，致教師或學者常有不同解讀。本研究以德懷術探究學習概念下應包含之次概念，先邀請專家針對課綱草案41項學習概念草擬共117項次概念，經三回合調查後，達共識之次概念共92項，未達共識者有26項，惟其中15項已於課綱學習概念中明訂，故本研究最後提出107項建議學習之次概念。本研究所得結果獲課綱參採，納入於課綱附錄的學習概念補充說明中。建議

* 本文為碩士論文改寫，指導教授為吳正己教授。

胡秋帆，國立臺灣師範大學資訊教育研究所博士生

王恩慈，國立臺灣師範大學資訊教育研究所碩士生

吳正己，國立臺灣師範大學資訊教育研究所、學習科學跨國頂尖研究中心特聘教授（通訊作者）

林育慈，國立臺灣師範大學資訊教育研究所、學習科學跨國頂尖研究中心副教授

電子郵件：chihwu@ntnu.edu.tw

投稿日期：2019年03月30日；修改日期：2019年09月01日；採用日期：2019年12月18日

未來課綱修訂應以本研究方式進行學習次概念之調查，未來研究應增加國中資訊教師人數以提升代表性，並提供大學教授更多教學現場之背景資料及經驗。

關鍵詞：資訊科技課綱、德懷術、學習概念

Identifying Learning Concepts for the New 12-Year Basic Education ICT Curriculum: A Delphi Survey

Chiu-Fan Hu, An-Tsu Wang, Cheng-Chih Wu, Yu-Tzu Lin

Abstract

The new Information and Communication Technology (ICT) curriculum for the new 12-Year Basic Education in Taiwan consists of seven categories of knowledge: algorithms, programming, system platform, data representation and process, ICT application, and ICT & human society. The draft curriculum only outlined broader learning concepts under each category. This may be a flexible way but it might cause confusion when schools and teachers implement the curriculum. This study employed Delphi technique to identify sub-concepts under each learning concept of

Chiu-Fan Hu, Doctoral Student, Graduate Institute of Information and Computer Education,
National Taiwan Normal University

An-Tsu Wang, Master, Graduate Institute of Information and Computer Education, National
Taiwan Normal University

Cheng-Chih Wu, Distinguished Professor, Graduate Institute of Information and Computer
Education; Institute for Research Excellence in Learning Sciences, National Taiwan Normal
University (Corresponding Author)

Yu-Tzu Lin, Associate Professor, Graduate Institute of Information and Computer Education;
Institute for Research Excellence in Learning Sciences, National Taiwan Normal University

Email: chihwu@ntnu.edu.tw

Manuscript received: Mar. 30, 2019; Modified: Sep. 01, 2019; Accepted: Dec. 18, 2019.

the curriculum. Twenty-one computer science educators from schools and universities served as the Delphi expert panel. After three rounds of questionnaire survey, 107 sub-concepts were identified and recommended for students to learn. The results were adopted in the appendix of the final released version of the ICT curriculum. This study recommended that the secondary ICT curriculum or teaching guidelines should include sub-concepts; the number of junior high school computer teachers should be increased to match the expert representation; finally, more information and ICT course planning experience should be provided to professors.

Keywords: ICT curriculum, Delphi survey, learning concepts

壹、緒論

因應資訊科技時代需求，各國陸續修訂資訊科技課綱，無論是澳洲推出以運算思維為核心的中小學Digital Technologies課程（Australian Curriculum, Assessment, Reporting Authority [ACARA], 2013），英格蘭的Computing課程強調運算思維與創造力活動（Department for Education [DfE], 2013），美國的K-12 Computer Science Framework（2016）、K-12 Computer Science Standards（Computer Science Teachers Association [CSTA], 2017）及AP Computer Science Principles（College Board, 2017），均強調培養運算思維的重要。

臺灣的資訊科技課綱（以下簡稱課綱）發展也與國際趨勢同步。2014年公布之《十二年國民基本教育課程綱要總綱》（教育部，2014）增設科技領域，於國、高中設資訊科技必修科目。課綱核心理念為運算思維，以「演算法」、「程式設計」、「系統平台」、「資料表示、處理及分析」、「資訊科技應用」及「資訊科技與人類社會」等類別規劃學習內容（教育部，2018），例如：「程式設計」類別下列有「程式語言基本概念、功能及應用」、「陣列程式設計實作」等學習概念。課綱中所呈現的資訊科學學習概念，是用以支持學生運算思維的培養，它並非學習的最終目的，教師可根據教學現場需求，設計各種教學活動，讓學生理解重要概念即可。基於此理念，課綱訂定的學習概念是較高層次且較寬廣的，一方面避免流於知識細節，以及時日稍久後內容過時；另一方面也保留彈性，讓教師及教科書能依不同需求設計內容。然而，因應最新資訊教育趨勢所訂定之課綱，難免涵蓋新興詞彙或過去在中學較少涉及的學習內容，致使各界解讀課綱時感到艱深晦澀（何榮桂，2015）。

各國課綱內容不一，有些鉅細靡遺，甚至提供教學方法；有些則精簡規範，僅列出課程原則。以美國K-12 Computer Science Standards為例（CSTA, 2017），就列出重要學習概念（如「程式語言及演算法」下的「控制」）與次概念（如使用序列、簡單迴圈發展程式等），並提供教案及教學方法。英格蘭的Computing課綱則較簡略，僅列出各學習階段應達到的能力指標及概念（DfE, 2013）；但Computing at School則針對課綱內容加以補充（Computing at School [CAS],

2014)。臺灣課綱的呈現方式居於前述二者之間，除教育部公布的課綱外，國家教育研究院編有科技領域課程手冊（國家教育研究院〔國教院〕，2019），解析課綱學習重點，說明教材編寫原則，同時提供教案參考。

資訊科技課綱修訂草案公布後，不乏有些爭論，認為新課綱是大學資訊科學專業課程的濃縮，內容過於艱深，抹煞學生學習興趣。這樣的認知源於課綱涵蓋許多資訊科學的概念，雖然只是要讓學生學習基本概念，但因資訊科學語彙的使用，易誤導以為要將大學專業內容複製到中學。例如：課綱中的「演算法」、「程式設計」及「系統平台」等都是資訊科學的用語，「演算法效能分析」、「程式語言基本概念」及「網路技術概念」等也是大學教科書常見，由於課綱無細節陳述，導致學者專家會以大學學習內容解讀。其他的困擾還包括對學習概念應包含的具體內容有不一致的認知，以「程式語言基本概念、功能及應用」為例，有教師認為應介紹C++、Python等常用的程式語言；有些則認為應介紹高、低階程式語言的差異；而有些又認為應教各種程式設計派典（paradigm），如物件導向、程序導向程式設計等。此外，課綱橫跨國、高中實施，因而存在看似相近的學習概念，例如：八年級的「陣列程式設計實作」及高中的「陣列資料結構的程式設計實作」該如何區分，也是教師較為困擾的。

資訊科技在十二年國教課綱中屬新科目，不若實施已久的國文、數學等科目，教師對學習概念已有相當高的共識。在新課綱實施初期，教師設計教學及教科書編寫都需要明確的方向，因此將課綱中較高層次之學習概念，進一步定義出較具體之「次概念」實屬必要工作。本研究在課綱草案公布後，以德懷術（Delphi Method，又稱大慧調查法）凝聚專家共識，針對課綱學習概念定義其下所包含的次概念。本研究所得結果，已獲參採於資訊科技科目課綱附錄中的「學習內容說明」。本文主要描述相關之研究過程及結果，並據以提出課綱教學實施的建議。

貳、文獻探討

以下探討相關國家資訊科技課綱之學習重點及呈現方式，除臺灣之課綱外，本研究僅挑選科技較先進、有獨立設科、且相關文獻可完整取得之國家，如美

國、英格蘭、澳洲等國。

一、臺灣資訊科技課綱

臺灣十二年國教資訊科技課綱在國中有6學分，高中2學分，合計必修8學分。2015年公布課綱草案（教育部，2015），以運算思維為理念，依「演算法」、「程式設計」、「系統平台」、「資料表示、處理及分析」、「資訊科技應用」及「資訊科技與人類社會」六大類別規劃其下學習概念，總計41項學習概念。2018年通過之課綱（教育部，2018），將學習概念減為38項，並於課綱公布前之最後修訂階段，於附錄中增列各學習內容之說明。調整的學習概念包含增列「文字式程式設計概念與實作」，合併及刪除「資訊科技與人類社會」下的相關學習概念。國教院（2019）編有《十二年國民基本教育課程綱要：科技領域課程手冊》，闡述課綱的教學內涵，包含建議的教學方法、工具或範例等，但並未說明各學習概念應包含之次概念。

二、美國資訊科技課綱

美國課綱主要由相關學術組織或團體所提出，重要課綱說明於下。

（一）K-12 Computer Science Framework

此課綱是由Association for Computing Machinery [ACM]及CSTA等組織合訂，目的是透過學習電腦科學核心概念與實作瞭解電腦科學（K-12 Computer Science Framework, 2016）。核心概念為運算系統、網絡與網際網路、資料與分析、演算法與程式設計及運算的影響，其下列有學習概念及次概念，例如：「資料與分析」包含集合、儲存、視覺化與轉換以及推論與模型等概念，並說明各階段應學習的次概念及範例，例如：十二年級學生在學習「集合」時，要知道資料是透過自動化程序持續的蒐集或產生等概念。

（二）CSTA K-12 Computer Science Standards

此課綱將學習階段分為Level 1A（5-7歲）、Level 1B（8-11歲）、Level 2（11-14歲）、及Level 3A（14-16歲），Level 3B標準則供對電腦科學有興趣繼續學習的學生（CSTA, 2017）。內容以K-12 Computer Science Framework為架構，說明各階段學習表現，並舉例說明重要學習概念。例如：Level 1A在「程式語言

及演算法」主題的「控制」學習概念下，說明學生要能用循序與簡單迴圈來表達想法或處理問題，指出控制結構，並以「說故事機」為例說明其重要性。美國加州K-12電腦科學課程課綱（California State Board of Education [SBE], 2018）也是參考此課綱制定。

（三）AP Computer Science Principles

此課程由College Board（2017）發展，係提供高中生先修大學電腦科學課程，主要發展學生跨學科應用的運算思維，架構包含運算思維實作及創造力、抽象化、資料與資訊、演算法、程式設計、網際網路及全球影響等關鍵概念，關鍵概念下指出學習要旨、目標及所需知識概念。例如：演算法包含「演算法是能被電腦和程式語言執行的一系列有次序、明確的指令」及「演算法能解決很多，但不是所有的運算問題」，以後者為例，學習目標有「說明在合理時間內執行和不能在合理時間內執行的演算法差異」及「解釋可解決與不可解決的電腦科學問題」，再指出達成各學習目標所需的知識。

三、英格蘭Computing課綱

Computing課程為中小學必修課（DfE, 2013），核心概念為抽象化、邏輯、演算法和資料表示方法，以培養用運算思維與創造力來理解與改變世界為目的。課綱中描述各階段學習目標及能力指標，例如：Key Stage 3（11-14歲）學習運算思維的重要演算法（如搜尋、排序），比較不同演算法的效率等。課綱內容較為簡略，但CAS組織針對課綱應學習的概念，撰寫教學引導手冊、組成教師社群，以及提供線上資源補充闡述課綱內容。

四、澳洲Digital Technologies課綱

澳洲課綱包含「數位系統的知識與理解」及「產生解決策略所需的技能」二大面向（ACARA, 2013）。各年段列出詞彙定義、學習概念及次概念或教學範例。以三、四年級的數位系統的知識與理解為例，包含「要分辨不同數位系統的設備及所傳遞不同的資料類型」、「能辨識不同的資料類型並探索如何使用不同的方法來表示相同的資料」等學習概念，以後者為例，又包含為辨識資訊系統能儲存或瀏覽的所有資料類型、利用表格重新組織句子、文字、數字或影像等資

訊、辨別與探索編碼與符號的資料表示法等概念。

綜合上述，各國課程的規劃以培養運算思維為主，讓學生具備以資訊科技解決問題的能力。各國課綱在學習內容的呈現上，有簡有繁。詳細的包含學習概念與次概念，也定義詞彙（如澳洲）、提供教學範例說明，如美國CSTA；簡略規範的英格蘭也由學術組織補充課綱內容。臺灣課綱的呈現居於二者之間，除教育部公布的課綱外，國教院（2019）也編有《十二年國民基本教育課程綱要：科技領域課程手冊》，說明輔助教學的工具與原則。

參、研究方法

本研究採用德懷術以取得專家對學習次概念之共識。德懷術是指一群專家匿名、獨立地表達意見，經由書面內容進行溝通與回饋，最終產生共識（Clayton, 1997），常用來解決較複雜的議題。過去已廣泛地被教育研究使用，包含課程發展、教師專業標準制定（吳淑禎，2011）等主題。

一、專家選定

德懷術需考量專家專業知識及代表性，本研究旨在提供課綱學習概念之建議，故邀請標準為：

（一）具資訊科學（工程）、資訊管理、資訊教育等領域碩士以上學位。

（二）熟悉或曾參與資訊科技課程綱要訂（修）定，或擔任審定本教科書審查委員。

考量服務場域之差異，本研究邀請大學教授及國、高中資訊教師擔任專家。由於108學年度前國中資訊課非必修課程，較不易邀得合於上述標準之國中教師，最終參與專家為九位大學教授、九位高中教師及三位國中教師，共21位。

二、問卷初稿

本研究初始問卷以2015年公布之資訊科技課綱草案為架構，針對中學必修課程六大學習類別之41項學習概念草擬次概念。初稿邀請八位課綱修訂委員，包含中學資訊科技教師、大學資訊領域教授及相關產業研究人員。每類別由二位

專家負責（有些專家參與二個類別）擬定次概念，共擬出117項次概念。此八位專家並不參與後續的德懷術問卷調查。問卷採李克式五點量表，5表示「非常同意」，1表示「非常不同意」，依此類推。

三、實施過程

調查實施時間為2016年3月到6月。調查中，專家需判斷各次概念是否契合所屬學習概念，並可提出修改建議。本研究共實施三次德懷術問卷調查，每次調查附前次統計結果與專家意見，專家僅針對未達共識之次概念繼續填答。三次調查結束後，召開一次專家會議，針對未達成共識之次概念進行討論，以作為調查結果解釋之參考。

四、共識標準

德懷術形成共識的標準可參考集中趨勢的結果，例如：平均數、中位數等，可由研究者決定達共識的結果（Clayton, 1997）。本研究設定的共識標準為：超過80%的專家勾選「同意」或「非常同意」，且其中有一半以上（9人）勾選「非常同意」。達到上述標準，即視為具共識納入應學習之次概念。反之，若超過80%的專家勾選「不同意」或「非常不同意」，且其中有一半以上勾選「非常不同意」，則「刪除」該次概念。

肆、結果與討論

初始問卷117項次概念經第一次調查後，有71項達納入共識，專家建議修改一項，並新增四項。第二次調查，新增11項納入。第三次調查，有九項納入，三項刪除。三次調查結束，初始問卷加上專家新增共121項次概念，最終92項達共識納入，三項達共識刪除，26項未達共識。以下以六大類別呈現調查結果。

一、程式設計

本類別課綱規劃七項學習概念，初始問卷擬定23項次概念，最終15項達共識納入（如表1），七項未達共識，一項刪除。

表1
 程式設計達共識納入之次概念

| 課綱學習概念 | 建議學習次概念 | 各等級人數 | | | | | 共識百分比 |
|---|---------------------------------------|-------|---|---|---|----|-------|
| | | 1 | 2 | 3 | 4 | 5 | |
| 七年級 | | | | | | | |
| 1.1 程式語言基本概念、功能及應用 | 1.1.1 程式語言的概念：資料型態、變數、輸入／輸出、算術運算、邏輯運算 | 0 | 3 | 1 | 8 | 9 | 81% |
| | 1.1.2 程式的功能與應用 | 0 | 0 | 1 | 5 | 15 | 95% |
| 1.2 結構化程式設計—循序與選擇結構、重複結構 | 1.2.1 循序指令 | 0 | 1 | 2 | 2 | 16 | 86% |
| | 1.2.2 選擇結構、巢狀選擇結構 | 0 | 0 | 1 | 7 | 13 | 95% |
| | 1.2.3 重複結構、巢狀重複結構 | 0 | 1 | 3 | 6 | 11 | 81% |
| | 1.2.4 選擇與重複結構之整合運用 | 0 | 1 | 2 | 7 | 11 | 86% |
| 八年級 | | | | | | | |
| 1.3 陣列程式設計實作 | 1.3.1 一維陣列 | 0 | 1 | 1 | 1 | 18 | 90% |
| | 1.3.2 一維陣列與解題實作：陣列與迴圈的整合應用 | 0 | 2 | 2 | 2 | 15 | 81% |
| 1.4 模組化程式設計的概念 | 1.4.1 模組化的優勢：重用性、結構性、開發效率（易於分工、除錯） | 0 | 1 | 2 | 3 | 15 | 86% |
| | 1.4.2 函數：函數傳遞參數、函數回傳值 | 1 | 1 | 1 | 5 | 13 | 86% |
| 1.5 模組化程式設計與問題解決實作 | 1.5.1 函數程式設計與問題解決：歸納模式、規劃解題步驟、分解問題 | 0 | 2 | 2 | 6 | 11 | 81% |
| 高中 | | | | | | | |
| 1.6 陣列資料結構的程式設計實作 | 1.6.1 一維陣列與解題實作 | 0 | 0 | 3 | 5 | 13 | 86% |
| | 1.6.2 二維陣列與解題實作 | 0 | 1 | 3 | 7 | 10 | 81% |
| 1.7 重要演算法的程式設計實作—遞迴結構、搜尋演算法、排序演算法、分而治之演算法 | 1.7.1 循序搜尋與二分搜尋之解題實作 | 0 | 1 | 3 | 5 | 12 | 81% |
| | 1.7.2 選擇與氣泡排序之解題實作 | 0 | 1 | 2 | 6 | 12 | 86% |

未達共識的次概念含遞迴、分而治之演算法、程式語言的種類、變數範圍與生命週期、記憶體配置方式、有序陣列中插入或删除元素的時間複雜度等。達共識刪除的次概念為八年級的「多維陣列」，原因是多維陣列較為抽象，矩陣的相關概念至高中才會學到，專家建議可延後學習。此想法可從高中二維陣列達共識列為次概念得到驗證。

二、演算法

本類別課綱規劃六項學習概念，初始問卷擬定22項次概念，德懷術專家新增一項，最終11項達共識納入（如表2），11項未達共識，一項刪除。

表2
演算法達共識納入之次概念

| 課綱學習概念 | 建議學習次概念 | 各等級人數 | | | | | 共識百分比 |
|-----------------------|--|-------|---|---|----|----|-------|
| | | 1 | 2 | 3 | 4 | 5 | |
| 七年級 | | | | | | | |
| 2.1 演算法基本概念—問題解析、流程控制 | 2.1.1 問題描述、簡化、模型建立、類比 | 0 | 0 | 2 | 10 | 9 | 90% |
| | 2.1.2 流程圖運用與繪製 | 0 | 1 | 1 | 7 | 12 | 90% |
| | 2.1.3 演算法簡介 | 0 | 3 | 1 | 5 | 12 | 81% |
| | 2.1.4 演算法與程式設計的關係 | 0 | 1 | 3 | 4 | 13 | 81% |
| | 2.1.5 問題解析：歸納模式（Finding pattern）、規劃解題步驟、分解問題 | 0 | 2 | 2 | 8 | 9 | 81% |
| | 2.1.6 流程控制：選擇、重複 | 0 | 0 | 3 | 4 | 14 | 86% |
| 八年級 | | | | | | | |
| 2.2 陣列資料結構概念與應用 | 2.2.1 一維陣列的概念與應用 | 0 | 1 | 2 | 4 | 14 | 86% |
| 高中 | | | | | | | |
| 2.4 重要資料結構的概念與應用—樹、圖 | 2.4.1 樹狀結構 | 1 | 0 | 2 | 5 | 13 | 86% |
| | 2.4.2 二元樹、二元搜尋樹 | 0 | 0 | 4 | 6 | 11 | 81% |

（續下頁）

| 課綱學習概念 | 建議學習次概念 | 各等級人數 | | | | | 共識百分比 |
|------------------------------|----------------|-------|---|---|---|----|-------|
| | | 1 | 2 | 3 | 4 | 5 | |
| 2.5 重要演算法的概念與應用—遞迴結構、分而治之演算法 | 2.5.1 迭代與重複結構 | 0 | 0 | 4 | 9 | 8 | 81% |
| 2.6 演算法效能分析 | 2.6.1 時間複雜度之概念 | 0 | 0 | 4 | 7 | 10 | 81% |

未達共識的次概念包含虛擬碼、搜尋、排序、樹與圖的應用、遞迴結構、演算法效能分析及空間複雜度等。達共識刪除的為八年級的「二維陣列的概念與應用」。主要原因是對國中生而言，二維陣列屬較難的內容。

三、系統平台

本類別課綱規劃六項學習概念，初始問卷擬定19項次概念，專家新增二項，最終18項達共識納入（如表3），二項未達共識，一項刪除。

表3
系統平台達共識納入之次概念

| 課綱學習概念 | 建議學習次概念 | 各等級人數 | | | | | 共識百分比 |
|----------------------|-----------------------------------|-------|---|---|----|----|-------|
| | | 1 | 2 | 3 | 4 | 5 | |
| 九年級 | | | | | | | |
| 3.1 系統平台重要發展與演進 | 3.1.1 系統平台演進歷程：大型主機、個人電腦、後PC時代 | 0 | 0 | 2 | 4 | 15 | 90% |
| 3.2 系統平台之組成架構與基本運作原理 | 3.2.1 系統軟硬體架構：作業系統、CPU、memory、I/O | 0 | 0 | 2 | 3 | 16 | 90% |
| | 3.2.2 開源軟硬體的趨勢 | 0 | 0 | 4 | 7 | 10 | 81% |
| 3.3 網路技術的概念與介紹 | 3.3.1 電腦網路的概念與發展歷史 | 0 | 0 | 3 | 4 | 14 | 86% |
| | 3.3.2 網際網路的重要概念 | 0 | 0 | 1 | 3 | 17 | 95% |
| | 3.3.3 無線區域網路的重要概念 | 0 | 0 | 3 | 4 | 14 | 86% |
| | 3.3.4 無線個人區域網路的重要概念 | 0 | 1 | 3 | 9 | 8 | 81% |
| | 3.3.5 行動通訊系統的重要概念 | 0 | 0 | 1 | 10 | 10 | 95% |

（續下頁）

| 課網學習概念 | 建議學習次概念 | 各等級人數 | | | | | 共識百分比 |
|------------------------------------|-------------------------------|-------|---|---|---|----|-------|
| | | 1 | 2 | 3 | 4 | 5 | |
| 3.4 網路服務的概念與介紹 | 3.4.1 網際網路服務 | 0 | 1 | 3 | 3 | 14 | 81% |
| | 3.4.2 物聯網系統 | 1 | 0 | 0 | 8 | 12 | 95% |
| | 3.4.3 雲端運算系統 | 0 | 0 | 1 | 6 | 14 | 95% |
| | 3.4.4 其他新興的網路服務 | 0 | 0 | 4 | 5 | 12 | 81% |
| | 3.4.5 社群平台介紹 | 0 | 1 | 2 | 4 | 14 | 86% |
| 高中 | | | | | | | |
| 3.5 系統平台之運作原理—工作管理與資源分配、分散式系統、網路路由 | 3.5.1 常見的排程演算法，如FIFO、LRU、LFU等 | 0 | 0 | 2 | 8 | 11 | 90% |
| | 3.5.2 分散式系統的概念與常見的應用系統 | 0 | 1 | 3 | 6 | 11 | 81% |
| 3.6 系統平台之未來發展趨勢 | 3.6.1 物聯網系統 | 0 | 1 | 3 | 5 | 12 | 81% |
| | 3.6.2 資料中心 | 0 | 1 | 2 | 6 | 12 | 86% |
| | 3.6.3 雲端運算 | 0 | 2 | 1 | 5 | 13 | 86% |

未達共識的次概念包含資訊科學發展重要人物與高速運算。達共識刪除的為「網路網路階層式的路由方法，以及常見的路由演算法」，許多專家認為路由相關演算法對高中生來講太難，多數專家提及學生只要知道路由器、會判斷路徑遞送等概念即可，不必強調演算法。

四、資料表示、處理與分析

本類別課網規劃五項學習概念，初始問卷擬定17項次概念，最終12項達共識納入（如表4），五項未達共識。未達共識的皆為「資料處理概念與方法」下的次概念。許多專家受到大學課程內容的影響，認為資料壓縮、資料轉換等概念須涉及較深的數學與演算法。

表4
資料表示、處理與分析達共識納入之次概念

| 課綱學習概念 | 建議學習次概念 | 各等級人數 | | | | | 共識百分比 |
|--------------------|--|-------|---|---|---|----|-------|
| | | 1 | 2 | 3 | 4 | 5 | |
| 九年級 | | | | | | | |
| 4.1 資料數位化之原理與方法 | 4.1.1 數位化原理與方法：取樣、量化 | 0 | 0 | 1 | 7 | 13 | 95% |
| | 4.1.2 資料數位化應用（圖片、地圖圖資、音訊、視訊） | 0 | 0 | 1 | 7 | 13 | 95% |
| 4.2 數位資料的表示方法 | 4.2.1 資料表示法的概念 | 0 | 0 | 2 | 7 | 12 | 90% |
| | 4.2.2 編碼概念 | 0 | 0 | 1 | 9 | 11 | 95% |
| | 4.2.3 二進位資料表示法：位元概念、數值編碼、文字編碼、繁體中文碼、統一碼 | 0 | 0 | 4 | 6 | 11 | 81% |
| 高中 | | | | | | | |
| 4.4 巨量資料的概念 | 4.4.1 巨量資料4V特性 | 0 | 0 | 1 | 7 | 13 | 95% |
| | 4.4.2 巨量資料處理與分析簡單實例：資料清理、資料理解、資料分析、資料視覺化 | 0 | 0 | 0 | 7 | 14 | 100% |
| | 4.4.3 巨量資料的挑戰與價值：巨量資料在不同領域的應用 | 0 | 1 | 3 | 7 | 10 | 81% |
| 4.5 資料探勘與機器學習的基本概念 | 4.5.1 資料分類概念 | 0 | 1 | 1 | 9 | 10 | 90% |
| | 4.5.2 機器學習的概念與應用實例 | 0 | 1 | 0 | 8 | 12 | 95% |
| | 4.5.3 資料探勘的概念與應用實例 | 0 | 1 | 0 | 8 | 12 | 95% |
| | 4.5.4 模式識別的概念與應用實例（如電腦視覺、自然語言處理、生物特徵識別等） | 1 | 2 | 1 | 8 | 9 | 81% |

五、資訊科技應用

本類別課綱規劃三項學習概念，初始問卷的12項次概念皆達共識，詳如表5。

表5
資訊科技應用達共識納入之次概念

| 課綱學習概念 | 建議學習次概念 | 各等級人數 | | | | | 共識百分比 |
|-----------------------------------|---|-------|---|---|---|----|-------|
| | | 1 | 2 | 3 | 4 | 5 | |
| 七年級 | | | | | | | |
| 5.1 資料處理應用專題—資料搜尋、資料組織與表達、資料運算與分析 | 5.1.1 以實作方式整合應用資料搜尋、組織與表達、運算與分析之概念與方法 | 0 | 0 | 2 | 3 | 16 | 90% |
| | 5.1.2 定義問題 | 0 | 1 | 3 | 6 | 11 | 81% |
| | 5.1.3 有效率進行文件、影像、聲音、圖像與其他數位資料之搜尋 | 0 | 0 | 2 | 3 | 16 | 90% |
| | 5.1.4 應用工具進行資料視覺化 | 0 | 1 | 2 | 7 | 11 | 86% |
| | 5.1.5 應用工具組織與整合各種資訊 | 0 | 1 | 2 | 6 | 12 | 86% |
| | 5.1.6 應用工具有效陳述概念與表達 | 0 | 0 | 2 | 7 | 12 | 90% |
| | 5.1.7 應用工具進行資料運算與分析以獲取所需資訊 | 0 | 0 | 3 | 6 | 12 | 86% |
| 九年級 | | | | | | | |
| 5.2 資訊科技應用專題—多媒體應用專題、程式設計應用專題 | 5.2.1 多媒體應用專題：須能以實作方式應用多媒體處理與分析之概念與方法 | 0 | 1 | 1 | 5 | 14 | 90% |
| | 5.2.2 程式設計專題：須能以程式實作方式應用「七年級或八年級所學之程式設計概念與方法」以及「九年級所學之系統平台或資料表示處理與分析之概念與方法」 | 0 | 1 | 3 | 1 | 16 | 81% |
| 高中 | | | | | | | |
| 5.3 數位合作共創的概念與工具使用 | 5.3.1 專案管理概念 | 0 | 0 | 4 | 5 | 12 | 81% |
| | 5.3.2 版本控制概念 | 0 | 0 | 1 | 8 | 12 | 95% |
| | 5.3.3 合作共創工具的應用 | 0 | 0 | 1 | 5 | 15 | 95% |

六、資訊科技與人類社會

本類別課綱規劃14項學習概念，初始問卷擬定24項次概念，專家新增一項，結果24項達共識納入（如表6），一項未達共識。未達共識的為程式安全防護，

部分專家認為此概念內容較為廣泛與複雜，國中生只要知道關於病毒或惡意程式防護的觀念即可。

表6
資訊科技與人類社會達共識納入之次概念

| 課綱學習概念 | 建議學習次概念 | 各等級人數 | | | | | 共識百分比 |
|---------------------|-----------------------------|-------|---|---|---|----|-------|
| | | 1 | 2 | 3 | 4 | 5 | |
| 七年級 | | | | | | | |
| 6.1 個人資料保護 | 6.1.1 網路與檔案資料的隱私權保護 | 0 | 0 | 0 | 2 | 19 | 100% |
| 6.2 資訊科技合理使用原則 | 6.2.1 數位作品之合理重製、公開播送或公開傳輸原則 | 0 | 0 | 1 | 1 | 19 | 95% |
| | 6.2.2 創用CC | 0 | 0 | 1 | 5 | 15 | 95% |
| 6.3 資訊安全 | 6.3.1 資料安全防護 | 0 | 0 | 1 | 3 | 17 | 95% |
| | 6.3.2 通訊安全防護 | 0 | 0 | 2 | 4 | 15 | 90% |
| 八年級 | | | | | | | |
| 6.4 資訊科技重要社會議題 | 6.4.1 網路成癮 | 0 | 0 | 0 | 3 | 18 | 100% |
| | 6.4.2 網路交友 | 0 | 0 | 1 | 3 | 17 | 95% |
| 6.5 資訊倫理與法律 | 6.5.1 網路言論之法律責任 | 0 | 1 | 0 | 5 | 15 | 95% |
| | 6.5.2 網路霸凌 | 0 | 0 | 0 | 4 | 17 | 100% |
| | 6.5.3 網路詐欺 | 0 | 0 | 1 | 5 | 15 | 95% |
| | 6.5.4 網路駭客 | 0 | 0 | 3 | 3 | 15 | 86% |
| 九年級 | | | | | | | |
| 6.6 資訊科技對人類生活之影響 | 6.6.1 資訊科技對食衣住行之影響與衝擊 | 0 | 0 | 3 | 5 | 13 | 86% |
| 6.7 資訊科技相關職業類科之升學進路 | 6.7.1 資訊相關科系之學習內容概述 | 0 | 1 | 2 | 3 | 15 | 86% |
| 6.8 資訊科技相關職業類科之生涯發展 | 6.8.1 資訊科技相關職業簡介 | 0 | 1 | 2 | 3 | 15 | 86% |
| 高中 | | | | | | | |
| 6.9 資訊科技的合理使用原則 | 6.9.1 數位作品之合理引用與編輯原則 | 0 | 0 | 2 | 2 | 17 | 90% |
| | 6.9.2 創用CC | 0 | 0 | 4 | 3 | 14 | 81% |
| | 6.9.3 數位作品之授權 | 0 | 0 | 2 | 4 | 15 | 90% |

(續下頁)

| 課綱學習概念 | 建議學習次概念 | 各等級人數 | | | | | 共識百分比 |
|-----------------------|-----------------------------------|-------|---|---|---|----|-------|
| | | 1 | 2 | 3 | 4 | 5 | |
| 6.10 個人資料的保護 | 6.10.1 科技對個人隱私之影響 | 0 | 0 | 2 | 3 | 16 | 90% |
| | 6.10.2 資訊科技對個人隱私之保護方法 | 0 | 0 | 1 | 5 | 15 | 95% |
| 6.11 資訊科技的重要社會議題 | 6.11.1 網路之公民參與 | 0 | 0 | 4 | 4 | 13 | 81% |
| 6.12 資訊科技對人類社會影響 | 6.12.1 資訊科技對各行業、人類文明、社會變遷之影響與未來發展 | 0 | 0 | 2 | 6 | 13 | 90% |
| 6.13 資訊科技領域性向之自我理解 | 6.13.1 選擇資訊相關科系所需之性向與特質 | 0 | 0 | 2 | 5 | 14 | 90% |
| 6.14 資訊科技相關行業之進路與生涯發展 | 6.14.1 大學資訊科學相關科系簡介 | 0 | 1 | 2 | 5 | 13 | 86% |
| | 6.14.2 資訊科技相關行業簡介與生涯規劃 | 0 | 1 | 2 | 5 | 13 | 86% |

七、未達共識但仍應納入學習之次概念

德懷術調查未達共識的26項次概念中，但其中15項在課綱的學習概念中已明確敘明（如表7，「課綱學習概念」欄粗體字），因此仍須納入建議學習之次概念。

表7
未達納入共識但課綱學習概念已包含的次概念

| 學習類別 | 課綱學習概念 | 建議學習次概念 |
|------|----------------------|----------------------------|
| 程式設計 | 1.7 重要演算法的程式設計 | 遞迴、函式呼叫與堆疊 |
| | 實作一遞迴結構、搜尋 | 分而治之演算法之解題實作（例如：快速排序、合併排序） |
| 演算法 | 2.3 基本演算法的介紹一搜尋、排序 | 分而治之演算法 |
| | 2.3 基本演算法的介紹一搜尋、排序 | 搜尋：循序、二元 排序：選擇、氣泡 |
| | 2.4 重要資料結構的概念與應用一樹、圖 | 樹的走訪 圖的表示法：陣列或串列 |

（續下頁）

| 學習類別 | 課綱學習概念 | 建議學習次概念 |
|------------|------------------------------------|--|
| | 2.5 重要演算法的概念與應用—遞迴結構、分而治之演算法 | 遞迴結構：與迭代之關係、終止條件、遞迴關係 分而治之演算法之概念與應用（例如：快速排序、合併排序） |
| | 2.6 演算法效能分析 | 演算法之效能分析與比較（例如：搜尋或排序演算法的分析與比較） |
| 系統平台 | 3.5 系統平台之運作原理—工作管理與資源分配、分散式系統、網路路由 | 網際網路階層式的路由方法，以及常見的路由演算法，例如：RIP、OSPF、BGP等 |
| 資料表示、處理與分析 | 4.3 資料處理概念與方法—資料整理與整合、資料壓縮、資料轉換 | 資料清理：整合重複的資料，找出錯誤的資料，填補遺失的資料 資料壓縮原理 資料轉換概念 資料格式轉換 基本資料轉換方法 |

課綱學習概念已涵蓋卻未達共識的次概念多為進階概念。例如：程式設計與演算法下的遞迴與分而治之，主要是大學教授認為這兩個演算法較為困難，若要撰寫程式，需要較多課程實施時間；但資訊教師提出，高中數學科已有與遞迴相關的數學歸納法學習內容，藉由遞迴程式可以展現電腦運算能力，且同時澄清數學歸納法的概念，認為應可納入高中階段的學習概念，但重點放在觀念理解而非程式設計與演算法細節。美國CSTA課程也有類似的學習規劃，CSTA將遞迴列為14-16歲學生整合學習數學斐氏數列的重要概念。臺灣課綱之訂定，對於遞迴與數學歸納法在不同課程中教授，也認為可以促進學生概念的統整與應用，更符合課綱規劃理念，讓學生理解運算思維原理，整合運用以解決問題。

演算法中的搜尋與排序，部分專家認為較複雜，國中生不易理解，也有專家提出搜尋與排序各教一個演算法即可。此外，建議只要讓學生知道不同演算法的效能及使用時機即可，不需要學習演算法的內容及程式實作細節等。事實上，搜尋與排序的概念也是國外課綱所列的重要概念，例如：CSTA及加州（SBE, 2018）課綱中，排序與搜尋是許多學習階段的重要學習概念，在較低的學習年段，建議透過不插電的學習活動讓學生體驗搜尋與排序在日常生活中的應用。樹

與圖下的次概念無法達成共識的原因是，次概念很多，有限的時間內該包含哪些是專家考量的主因，最後專家會議討論以樹與圖的基本概念為主即可。在演算法之效能分析與比較上，大學教授認為中學階段演算法的重點應為問題解決，以及學習影響程式效能的因素，讓學生瞭解效能分析和最佳化的概念即可，不必實際分析各種複雜的演算法效能。因此在演算法的次概念上，專家在會議中表達一致的想法：只要能扣緊運算思維的精神，不納入過於細節或艱深的理論，演算法的教學仍能培養學生透過運算解決問題的能力。

在系統平台中，「網際網路階層式的路由方法，以及常見的路由演算法，例如：RIP、OSPF、BGP等」是達共識刪除的次概念。專家認為路由相關演算法對高中生來講太難，多數專家提及學生只要知道路由器、會判斷路徑遞送等概念即可，不必涵蓋演算法。但在專家會議中，中學教師認為演算法對中學生並沒有外界想像中的困難，建議教學重點為路由演算法的基本觀念與原理，演算法的細節則可省略，多數專家皆認可這樣的想法。資料處理概念與方法下的次概念都未有共識，主要是大學教授提出資料處理的概念與方法的學習重點在瞭解資料處理的必要性與優點，以及資料處理的方法與實作，不需要瞭解深奧理論或演算法細節，憂心教學過度強調理論內容。但中學教師認為在巨量資料驟增的時代，學生需要具備基本觀念，雖然資料清理等在過去是較為專精的領域特殊概念，但現今已不再獨為領域專家所需要。經會議討論後，多數專家認同資料清理、壓縮與轉換的基本觀念確實為重要概念，例如：澳洲課綱（ACARA, 2013）也強調資料格式、壓縮等概念的學習。

八、教學實施建議

分析研究結果發現，為服膺新課綱運算思維之精神，於規劃複雜、艱深學習概念的教學時，可以從問題解決、教學策略以及跨領域學習三個方向思考。

（一）艱深的學習概念應聚焦在觀念於問題解決之應用，而非理論知識

以課綱中壓縮、遞迴與排序為例，在資訊專業領域屬於進階的概念，並有其自成一套之理論知識範疇。然而，由於資訊教育的進步，這類觀念的學習年段必須向下延伸。但中學教師若單從字面意義來看，容易因刻板印象，認為這些概

念在過去皆於大學階段教授而連結到複雜的理論與演算法，導致教學方向的偏差。事實上，在中學階段，教學重點應是此等觀念於問題解決之應用。如會議討論中，有些專家特別指出，教學重點應為讓學生瞭解基本概念及相關應用。以基本演算法的介紹——搜尋、排序為例，學生不需要記憶演算法的詳細程序，只需理解為何這樣的程序可以達成排序的任務，以及如何將其應用於解決需要排序任務的問題。如專家所言：「讓學生理解循序和二元搜尋的差異即可」、「找兩個排序演算法為例，體驗不同演算法在效能上的優缺點，不宜過度強調排序演算法本身」、「多數人只需要使用`sort()`去解決問題，不見得需要知道`sort()`是怎麼製作的」。又以資料處理概念與方法為例，在資料轉換上，不需探索演算法細節，如專家所述：「只需要知道資料轉換的必要性、優點、代價，並不需要知道JPEG、MPEG、ZIP等演算法」。同理，學生不需要知道壓縮演算法所涉及的數學理論，但應理解為何壓縮演算法能達成資料壓縮，如此一來，學生可發展出在面對各領域不同的問題時思考問題解決策略的能力。類似的觀點亦適用於其他複雜概念，例如：演算法效能分析中的時間複雜度。有專家明確指出：「最好是實際上讓學生學習使用簡單的效能分析工具和方法，瞭解效能分析和最佳化的概念」。因此，重要概念的教學重點應強調基礎知識的理解與應用，以培養透過運算原理解決問題之能力，勿探究艱深的理論，以免流於理論知識的記誦，妨礙學生的興趣發展。

（二）運用適當教學策略與工具幫助學生概念理解，並增進學習動機

對於複雜的概念，教學時應掌握概念的主要精神，透過範例與實作體驗概念與問題解決之關係，藉由適當的教學策略（例如：動手實作、用適當的範例展示概念、問題導向式的引導等），以及適當的教學工具（例如：視覺化程式設計工具、演算法模擬動畫、網路模擬工具等），「低門檻、高極限」是選擇運算思維學習工具的重要原則，讓學生能專注設計，將學習焦點放在高階思維（林育慈、吳正己，2016），能引發學生學習興趣並降低學習難度。以程式設計為例，有專家便直接建議「七年級生第一次接觸資訊科技，應該讓他們使用視覺化的程式語言以啟發和維持興趣」及「多用簡單的範例帶領學生體會`for`、`while`、`if else`的使用方式和時機」。在基本演算法部分，不需要深入探討搜尋與排序演算法的詳細

程序，以培養學生對資訊領域的興趣。有專家便明確說明：「……應著重在實作，讓學生體驗如何透過程式與電腦溝通，不宜講解太多艱深的觀念」。專家也提醒應留意實作的難易度：「可帶些程式語言發展的歷史故事引起學生興趣，實際程式設計的課程不宜太過艱深」。不同類別亦可搭配教學，專家建議：可挑選程式設計、演算法、系統平台，以及資料表示、處理與分析等類別中的相關概念進行教學設計，以提供學生整合不同學習概念之機會，進而瞭解運算的不同面向如何搭配運作，例如：瞭解演算法與程式設計如何影響系統平台的運作、體驗資料處理演算法的設計如何幫助系統運作等。系統平台的演進則可透過故事引發學生學習動機，避免流於歷史知識的背誦。而基礎知識，專家認為宜多用實際範例來介紹程式的功能與應用，藉此闡述運算思維，自然理解演算法、資料、網路、運算平台等實際作用與重要性，引發學生學習興趣，避免背誦程式語言種類與語法等內容。

（三）規劃跨領域學習，幫助學生理解、應用與整合複雜的概念

課綱理念為培養學生應用資訊科技的概念理解、歸納分析或解決生活中的問題，並於實作中培養運算思維。教學上可採用STEM科際整合教學，讓學生在問題解決的過程中，有機會使用多個領域知能分析問題、形成問題與假設並驗證（Lin, Wang, & Wu, 2019）。過程中，不但能體驗運算之能力，亦能透過電腦運算幫助學生探究其他領域的複雜現象。例如：在國中可整合生物課所學，藉由運算工具探索細胞分裂的現象，從中體會細胞分裂複雜的動態過程，同時應用、整合與測試資訊科技陣列與迴圈的觀念。跨域學習亦能讓學生理解其他領域問題能藉由電腦運算更有效地被解決，例如：在高中，學生在數學課學習到矩陣與數學歸納法，若能於程式設計或演算法的教學活動中，整合已學的數學觀念，讓學生思考與體驗電腦如何透過二維陣列與遞迴結構處理複雜的數學運算，學生則更能體會資訊科技與數學間的關係，亦更能理解相關概念。這樣的理念便在專家會議中被提出：「高中數學已有數學歸納法的概念，在資訊科技裡帶入是蠻適合的」。因此，跨域學習能讓學生理解如何應用資訊科技在其他學科學習，更能幫助學生理解複雜的資訊科技概念。當然，有些看似相同的觀念在不同學科可能有不同的意義，也是教學上應該注意與釐清的，例如：程式設計中的變數與運算概念。由於變數的使用與運算子的順序在不同的語言有不同的用法，可能會混淆正

在學習數學變數觀念的學生。

伍、結論與建議

本研究經過三回合的德懷術問卷調查後，提出十二年國教資訊科技科目課綱中學生應學習之107項次概念，其中92項為專家共識，15項雖未達共識但已明訂於課綱，故仍應納入學習。未達共識的皆為較複雜的概念，且多集中於程式設計與演算法類別，例如：遞迴、分而治之及搜尋與排序演算法等；其他則包含網路路由及資料壓縮與轉換等。整體而言，專家們認為中學在程式設計上不宜太艱深，且不應強調演算法細節。教師設計教學活動及教材時，可參考本研究所提出之次概念進行規劃。

本次資訊科技課綱之修訂，為保留教學彈性，僅規範較高層次之學習概念，但在公聽會及課審會階段有相當多意見希望能呈現更具體的次概念，本研究適時於該階段完成初步調查，乃提供課綱相關內容撰寫之參考，最後於課綱附錄中加入相關學習概念之說明。建議未來課綱之修訂，無論是要將學習次概念納入課綱，或是以附錄呈現，或是納入課綱課程手冊之中，均應將學習次概念之調查研究納入課綱修訂的程序之中，透過類似本研究的調查方式，取得專業學者與教師的共識，以提供學生最適切的學習內容。本研究在專家的選取上，因受限於同時兼具資訊專業及具課綱修訂經驗之國中資訊教師難覓，最後僅有三位國中資訊教師納入專家小組，未來在資訊科技納入國中必修課程後，應有更多符合相關要件之教師，未來研究應依課綱國、高中學習年段學習內容的比例，納入相對比例之教師，以提升德懷術專家代表性。

從研究過程中發現，本研究邀請的大學教授與中學教師的看法存在差異。可能原因是，二者的授課對象及教學經驗不同，以及專業背景的差異所致。課綱中的學習概念是以資訊專業術語呈現，例如：演算法、機器學習、巨量資料等，大學教授很自然地從大學資訊專業授課經驗思考，將其對應到大學教科書中複雜且深入的理論，因此認為不適合國、高中生學習。他們認為中學的資訊課程應以實作、啟發學習興趣為主，較進階、偏理論或是較難實作的概念皆不適合在中學教學。然而，在問卷調查後的專家討論會議中，他們也認同理論思考學習對培養運

算思維之重要性，惟需佐以適當教學方法。中學教師因熟悉中學教學環境，瞭解如何透過簡單範例或合適工具與策略進行教學，因而對涉及理論的次概念多持正向態度；但另一方面，反而易將概念與電腦軟硬體操作（例如：文書處理、影像編修、影音編輯、網路維修等）連結，忽略運算思維的精神。未來研究應注意此二類專家意見的平衡，以及選擇對教育現場有更多經驗的專家，並提供專家更多相關背景資訊。

最後，本研究針對教學內容與教學策略提出相關建議。在教學內容的選擇上，課綱規範的學習概念是完整架構國、高中生在不同階段應能理解的資訊科學核心概念。本研究進一步提出此架構下之次概念，能讓教師有所依循，規劃適切的教學內容，以使學生習得課綱所包含的學習概念。在教學策略的設計上，由本研究德懷術實施的過程與結果可以發現，不同教學場域的教師具備共通的理念：教授在傳統被視為進階的學習概念時，應避免鑽研複雜的理論內容，而應聚焦在概念於問題解決之應用。因此，教師應以課綱核心理念為目標，設計與發展適合的教學策略（動手實作、跨域學習等）與工具（視覺化與模擬等）。例如：教師在教授課綱「基本演算法的介紹－搜尋、排序」學習概念時，不宜著眼於排序法演算法的細節，而可透過各種排序法的動畫模擬，讓學生體驗不同排序方法的歷程與時間複雜度等，藉以習得「選擇排序法」、「氣泡排序法」的基本觀念，並理解資料排序的目的與使用時機。此外，由於重要概念的學習年段向下延伸，教師宜根據學生認知層次調整教學方法，並適時提供鷹架，讓學生專注於思考，且透過專題製作等方式提供學生問題解決與驗證概念的機會。

國中資訊科技科目屬於新增學科，相較於其他學科，教師及學科專家們恐需有較長時間的磨合。該如何實際透過教學，將生活情境脈絡與這些知識概念結合，培育學生跨領域整合與應用的通用能力，瞭解教學實施的有效性，落實以核心素養為導向的課程，值得更多研究注入，建議未來相關研究可以參考本研究結果以及教學實施建議深入發展教學模式並進行評估。此外，本次課綱中有些學習概念屬新興科技（例如：深度學習、巨量資料分析），部分資訊教師可能缺乏相關專業知識或教學經驗，需開發更多的教學活動與教材提供教師參考。此外，課綱部分學習概念橫跨國、高中（例如：搜尋與排序），兩個階段的學習內容該如何銜接，亦值得未來研究探討。

致謝：感謝科技部補助「以運算思維為核心的電腦課程教材發展與評估」計畫（103-2511-S-003-023-MY2）支持。

DOI: 10.3966/102887082020036601003

參考文獻

- 何榮桂（2015）。試論十二年國民基本教育「資訊科技」課程綱要規劃草案。《教育研究月刊》，250，48-64。
- [Ho, R.-G. (2015). Evaluation of the draft computer and information technology curriculum in twelve-year national fundamental education. *Journal of Education Research*, 250, 48-64.]
- 林育慈、吳正己（2016）。運算思維與中小學資訊科技課程。《教育脈動》，6。取自https://teric.naer.edu.tw/wSite/ct?ctNode=655&mp=teric_b&xItem=1852991&resCtNode=453&OWASP_CSRFTOKEN=RLO9-WGBQ-57ZC-S3FJ-AOPZ-ZGLW-KKI0-ALXI
- [Lin, Y.-T., & Wu, C.-C. (2016). Computational thinking and 12-Year Basic Education ICT curriculum. *Pulse of Education*, 6. Retrieved from https://teric.naer.edu.tw/wSite/ct?ctNode=655&mp=teric_b&xItem=1852991&resCtNode=453&OWASP_CSRFTOKEN=RLO9-WGBQ-57ZC-S3FJ-AOPZ-ZGLW-KKI0-ALXI]
- 吳淑禎（2011）。中等學校師資培育課程的發展特色與教育專業課程的能力指標分析。《教育研究集刊》，57（4），1-41。
- [Wu, S.-C. (2011). The curriculum development characteristics and competence indicator analysis of the secondary school teacher education programs. *Bulletin of Educational Research*, 57(4), 1-41.]
- 教育部（2014）。十二年國民基本教育課程綱要總綱。臺北市：作者。
- [Ministry of Education, Taiwan. (2014). *Curriculum guidelines of 12-year basic education: General guidelines*. Taipei, Taiwan: Author.]
- 教育部（2015）。十二年國民基本教育課程綱要：國民中小學暨普通型高級中等學校「科技領域」（草案）。臺北市：作者。
- [Ministry of Education, Taiwan. (2015). *Curriculum guidelines of 12-year basic education: Technology learning area (draft)*. Taipei, Taiwan: Author.]
- 教育部（2018）。十二年國民基本教育課程綱要：國民中小學暨普通型高級中等學校「科技領域」。臺北市：作者。

- [Ministry of Education, Taiwan. (2018). *Curriculum guidelines of 12-year basic education: Technology area*. Taipei, Taiwan: Author.]
- 國家教育研究院 (2019)。十二年國民基本教育課程綱要：科技領域課程手冊。新北市：作者。
- [National Academy for Educational Research. (2019). *Curriculum guidelines of 12-Year Basic Education: Curriculum handbook for technology learning area*. New Taipei, Taiwan: Author.]
- Australian Curriculum, Assessment, Reporting Authority. (2013). *Digital technologies*. Retrieved from <https://www.australiancurriculum.edu.au/f-10-curriculum/technologies/digital-technologies/>
- California State Board of Education. (2018). *Computer science education*. Retrieved from <https://www.cde.ca.gov/be/st/ss/computerscicontentstds.asp>
- Clayton, M. J. (1997). Delphi: A technique to harness expert opinion for critical decision-making tasks in education. *Educational Psychology*, 17(4), 373-386.
- College Board. (2017). *AP computer science principles*. Retrieved from <https://apcentral.collegeboard.org/pdf/ap-computer-science-principles-course-and-exam-description.pdf?course=ap-computer-science-principles>
- Computer Science Teachers Association. (2017). *K-12 computer science standards*. Retrieved from <http://www.csteachers.org/page/standards>
- Computing at School. (2014). *Computing in the national curriculum*. Retrieved from <https://community.computingatschool.org.uk/resources/2619/single>
- Department for Education. (2013). *National curriculum in England: Computing programmes of study*. Retrieved from <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study>
- K-12 Computer Science Framework. (2016). Retrieved from <https://k12cs.org/>
- Lin, Y. T., Wang, M. T., & Wu, C. C. (2019). Design and implementation of interdisciplinary STEM instruction: Teaching programming by computational physics. *The Asia-Pacific Education Researcher*, 28(1), 77-91.