

## 大學程式能力檢定(CPE)

「ACM 亞洲區台灣賽區大專程式設計競賽」自 1995 年起，每年在台灣各大學輪流舉行。為了提升國內大學生的程式設計能力，各大學相關科系的教授於 2008 年組織了「國際計算機器協會程式競賽台灣協會」(ACM-ICPC Contest Council for Taiwan, 簡稱 ACM-ICPC Taiwan Council)，做為跨校交流與合作的平台。該協會下設三個委員會如下：

1. 推動委員會：負責資源與庶務之整合，原則上由參與學校之計算機中心（或等同單位）主任或資訊系系主任組成。
2. 技術委員會：由教練與命題老師組成，負責培訓與命題事務，原則上成員須具備程式培訓與命題之能力與經驗。
3. 大學程式能力檢定委員會 (Collegiate Programming Examination Committee, 簡稱 CPE Committee)：共同舉辦 CPE 程式檢定考試，由已參與及即將參與舉辦 CPE 檢定考試之學校代表參加，該學校代表原則上為該校考場負責人。

**大學程式能力檢定 (Collegiate Programming Examination, 簡稱 CPE) 旨在提升全台灣學生的程式設計能力，由學生透過線上程式設計，利用電腦自動評判，以檢測程式設計能力。CPE 每年辦理四次，大約為每年的 3、6、9、12 月。CPE 採電腦現場上機考試，以電腦自動評判，並由各校派員監考。考試時，會封閉與考試無關的網路。考生除紙本字典外，不能攜帶任何資料。考生若為大專學生，可免費報名。**

### 大學程式能力檢定：CPE 秘笈 (博客來推薦)

- 作者：[林盈達](#)、[黃世昆](#)、[楊昌彪](#)、[葉正聖](#)、[謝育平](#)
- 出版社：[美商麥格羅·希爾](#)
- 出版日期：2013 年 02 月 25 日
- 語言：繁體中文 ISBN：9789861579412

## 內容簡介

本教材設計包含三個面向：系統、基礎、題解。在系統方面，我們在第一章介紹 ACM-ICPC 及 CPE 的發展與規則；第二章介紹 CPE 線上(on-line)練習與現場(on-site)考試的系統與機制；第三章介紹一個本機端的練習軟體——瘋狂程設，它透過測資腳本與批改腳本讓學生在練習中減少語法、語意與邏輯的錯誤，也透過短碼競賽讓學生能更精簡地撰寫程式。

在基礎方面，第四章介紹 C 與 C++ 輸入輸出的函式與格式，減少初學學生因為程式輸出輸入問題造成上傳程式錯誤的可能；第五章有系統地講解解題技能，從理解題意到挑選合適的演算法（包括排序、搜尋、貪心(greedy)、動態規劃、圖形走訪、最小生成樹、最短路徑、最大流等常用演算法）不一而足，同時考量程式執行時間與記憶體用量，並交代如何設計測資以在上傳程式前檢驗程式的正確性；最後提醒要善用既有資源，利用函式庫來設計解題所需的功功能，以減少撰寫程式的時間。

在題解方面，我們依據難易等級提供三章共 84 題之題解，另有 8 題用在前面章節中做為例子。第六章至第八章分別是一顆星至三顆星題目之題解，除了區分難易度，再依據題型

分節，其中包含字元與字串、數學計算、大數運算、幾何、排序、圖論、模擬、動態規劃等。讀者可以透過本書進入程式設計之門，並培養精進解題與程式的實力。

### 作者簡介

#### 林盈達

現職：國立交通大學資訊工程學系教授

學歷：UCLA 電腦科學博士

專長領域：網路協定設計、實作、分析與測試；網路安全、無線通訊及嵌入式軟硬體效能

#### 黃世昆

現職：國立交通大學資訊技術服務中心副主任

學歷：交通大學資訊工程博士

專長領域：軟體自動測試研究、軟體安全、自動攻擊產生器(CRAX)

#### 楊昌彪

現職：國立中山大學資訊工程學系教授兼系主任

學歷：清華大學資訊科學博士

專長領域：演算法及其延伸應用；字串、檔案相似度比對相關演算法之設計與分析

#### 葉正聖

現職：銘傳大學資訊傳播工程系助理教授

學歷：臺灣大學資訊工程博士

專長領域：電腦圖學、虛擬實境、電腦視覺及互動技術

#### 謝育平

現職：銘傳大學資訊工程學系助理教授

學歷：臺灣大學資訊工程博士

專長領域：計算組合學、網路路由技術、數位典藏、圖書館自動化、光學文字辨識與課程自動化

## 目錄

- 第一章 程式能力檢定簡介
- 第二章 程式能力檢定評審系統
- 第三章 瘋狂程設軟體
- 第四章 C/C++基本輸入與輸出
- 第五章 由基礎至進階
- 第六章 一顆星題解
- 第七章 二顆星題解
- 第八章 三顆星題解

# 序

## 拔擢頂尖 vs. 提升平均

由於 ACM-ICPC 亞洲區黃金雄主任（美國德州大學教授）的鼓勵與協助，「國際計算機器協會程式競賽台灣協會」（ACM ICPC Contest Council for Taiwan）於 2008 年成立。成立之初，協會的思考重點都在如何拔擢頂尖，讓更多的頂尖學生參加 ACM-ICPC 區賽（Regional Contest）及總決賽（World Final）。但我們很快地發現，拔尖只關注大約前 5% 的少數學生，大部分學生對這些國際競賽幾乎沒有投入甚或注意。根據我們的觀察，由於程式作業抄襲或修改容易，約四分之一的資訊系學生不太會寫程式（各校比率略有差異），四分之三的學生程式寫得不夠多（除了修課作業與專題要求之外不寫程式），未來會選擇從事程式設計的學生不到二分之一，其他傾向轉而投入不太需要撰寫程式的工作。這種「生態」當然影響了國家資通訊產業設計產品的「產能」，而產業界對於大學生的程式設計訓練不夠紮實也是抱怨聲不斷。因為這導致他們在徵聘新人時，必須透過自己設計的程式測驗才能挑選出適合的人才。

許多大學教授因有研究論文發表的壓力，對於研究生的研究要求，著重於創新設計與理論分析的突破，而較少要求系統實作之苦工。他們因為專心致力於研究，而無暇顧及基礎的程式教學訓練。有心於大學部程式訓練課程的教授，單憑一己之力也很難改變整個生態。

上述的情形讓協會開始在拔尖之餘，開始省思如何提升整體平均。思考的大方向是將 ACM-ICPC 國際賽的題庫拿來做為標準測驗的題目，然後推廣至各校共同辦理測驗，並且採認於大學部畢業要件與研究所入學參考。因為 ACM-ICPC 國際賽的題目都經過歷練，有相當的品質與水準，所以不需要擔心題目品質的問題；也因為題庫夠大（目前已經超過 3600 題，並且陸續增加中），所以也不需要擔心學生可能做過我們所挑選的題目。如果學生做過題目而且在考試時也可以寫得出程式，其程式能力必定也相當不錯，畢竟程式設計需要理解與邏輯思考，無法單靠死背。有了題目品質的保證後，我們設計了「大學程式能力檢定」（Collegiate Programming Examination, CPE）做為考試的形式與機制，希望透過 CPE 及各校的共同參與來改變上述的生態，藉此提高台灣資訊產業的產能與競爭力，並增加參加國際競賽的可能人口。

### **CPE 配方：多校、千人、同步、遠距、同一份題目之程式能力檢定**

CPE 具有獨特的配方，而有別於現有的競賽與檢定系統，例如 ACM-ICPC 區賽與總決賽使用單一實體場地最多 100 隊同時競賽，或電腦技能檢定在電腦教室隨時有人考試，而考題由遠端的伺服器隨機抽取。上述兩種實體場地或遠端題庫的模式都無法滿足千人同時考試的目標，必須結合兩者才能達到。所以「CPE 的配方」是多校、千人、同步、遠距、同一份題目；多校的場地才能免除舟車勞頓，又能達到千人的規模，而遠距題庫能支援各校同步舉辦來考同一份題目。

由於 CPE 是一項檢定考試，監考、防弊與系統穩定度非常重要。學生在各校的電腦教室考試，用戶端的電腦軟體必須確保學生無法連線到非 CPE 伺服器以外的地方，而 CPE 係透過虛擬主機的機制達到這項限制。此外，各校考場需避免學生在現場交談作弊，甚或冒名考試，所以電腦教室要有專人監考。伺服器端除了支援多校千人同步存取題目，以及自動評審學生

上傳的程式之外，還要克服系統延展性與穩定性的問題。目前 CPE 系統架構有多台前端與後端伺服器，可兼具支援大量考生及達到伺服器穩定備援的目的。

前述提到，題目是由 ACM-ICPC 區賽與總決賽比賽過而收錄於 UVA 題庫，品質沒有問題，可以避免多年前資訊教育界推動 TGRE 因題庫品質問題，造成考試成績鑑別力不足的情形。但是，UVA 題庫並沒有公布上傳程式的測試資料（簡稱測資），所以 CPE 在挑選題目時必須自行準備測資。另外，ACM-ICPC 區賽與總決賽進行五小時，三人組成一個隊伍，題目大約在 8 至 10 題左右。相較之下，CPE 是個人考試檢定，進行三小時，題目有 7 題，除了時間較短且題目較少外，選擇的題目也朝向簡單化。在一至五顆星難易等級中，ACM-ICPC 國際賽一顆星與兩顆星大約各只有一題，其餘題目都是三顆星以上；而 CPE 一顆星有三題，二顆星有二題，三顆星以上有二題。

### **CPE 現況**

CPE 自 2010 年 6 月第一次由中山大學與交通大學合辦開始，每季舉辦一次，時間為平日 18:00 至 21:40（包含考前練習）。截至 2012 年 12 月已舉辦十次，參與協辦的學校數由 2、6、9、19、17、21、25、30 校成長到約 40 校，參加檢定考試的學生數也成長到將近 1000 人。2011 年至 2013 年由中山大學主辦（選題與測資準備、統籌相關行政事務與對外宣傳），交通大學負責技術支援（伺服器維護），其他學校參與協辦（考場準備與監考）。CPE 也提供學生申請中英文成績單，其中會顯示絕對成績（總題數 7 題、答對題數、成績等級）及相對成績（參加學生數、排名）。由於考試人數夠多、題目具有鑑別力，所以這份成績單具有相當的可信度。程式能力不好的學生，可能一題也解不出來；能解四題的學生，程式能力已是相當優秀；而一次解出兩題是許多大學設定的及格標準。

CPE 的推行，需要舉辦與採認同時並進，如果沒有學校採認，空有許多學校舉辦，並無法吸引夠多的學生來考試。目前已有多所大學將 CPE 設定為：(1) 大學部之畢業要件（單次考試答對兩題或多次考試累積答對三題或四題）；(2) 碩士班甄試入學之參考資料（列入申請表、推薦信或招生簡章）；(3) 在課程上使用當作期中期末上機考試的方式。

### **教材設計：系統、基礎、題解**

儘管有不少線上資源可以讓學生了解 ACM-ICPC、UVA、CPE，但仍欠缺一本整合各種資訊的書籍，學生及教師都會需要一本 CPE 入門的書，讓學生可以準備檢定考試，也讓教師能善加利用 CPE 於課程、入學與畢業。本教材設計包含三個面向：系統、基礎、題解。在系統方面，我們在第一章介紹 ACM-ICPC 及 CPE 的發展與規則；第二章介紹 CPE 線上(on-line)練習與現場(on-site)考試的系統與機制；第三章介紹一個本機端的練習軟體——瘋狂程設，它透過測資腳本與批改腳本讓學生在練習中減少語法、語意與邏輯的錯誤，也透過短碼競賽讓學生能更精簡地撰寫程式。

在基礎方面，第四章介紹 C 與 C++ 輸入輸出的函式與格式，減少初學者因為程式輸出輸入問題造成上傳程式錯誤的可能；第五章有系統地講解解題技能，從理解題意到挑選合適的演算法（包括排序、搜尋、貪心(greedy)、動態規劃、圖形走訪、最小生成樹、最短路徑、最大流等常用演算法）不一而足，同時考量程式執行時間與記憶體用量，並交代如何設計測資以在上傳程式前檢驗程式的正確性；最後提醒要善用既有資源，利用函式庫來設計解題所需的功能，以減少撰寫程式的時間。

在題解方面，我們依據難易等級提供三章共 84 題之題解，另有 8 題用在前面章節中做為例子。第六章至第八章分別是一顆星至三顆星題目之題解，除了區分難易度，更依據題型分節，其中包含字元與字串、數學計算、大數運算、幾何、排序、圖論、模擬、動態規劃等。我們希望讀者可以透過本書進入程式設計之門，並培養精進解題與程式的實力。

### 本書特點

多校、千人、同步、遠距、同一份考題之大學程式能力檢定(CPE: Collegiate Programming Examination)之入門教材。

學生只需具備 C 或 C++ 基礎程式能力即可上手。

線上練習系統與現場考試系統之介紹。

基礎輸入輸出與進階解題技能之講解。

ACM-ICPC UVA 題庫中精選 92 題題目之題解。

依難易度蒐集各類題型：字元與字串、數學計算、大數運算、幾何、排序、圖論、模擬、動態規劃等。

每一題解包含 UVA/CPE 編號、題意、解法、程式碼及程式碼註解。

## §內文 1

### CHAPTER1 程式能力檢定簡介

#### 1.1 ACM 國際大學程式競賽

ACM ICPC (International Collegiate Programming Contest, 國際大學程式競賽) 是由 ACM (Association for Computing Machinery, 國際計算機器協會) 主辦的一年一次程式設計競賽。藉由競賽方式來展現大學生創新能力、團隊精神，以及在壓力下編寫程式、分析和解決問題的能力。自從 1970 年代開創以來，經過三十多年的發展，ACM ICPC 已成為全球電腦界中歷史最悠久且最具影響力的程式競賽。以 2011 至 2012 年為例，參加各地區域賽的隊伍超過 8000 隊，涵蓋 88 個國家及超過 2000 所大學。區域賽優勝隊伍會再集中於一處參與世界總決賽的競逐。以下列舉 ACM ICPC 的重要里程碑：

- 1970 年：美國 Texas A&M University 大學程式設計比賽。
- 1977 年：第一次舉辦世界總決賽。
- 1977 至 1989 年：參與比賽的大學主要來自美國與加拿大。
- 1989 年：建立區域賽(regional)制度，優勝隊伍才能參加世界總決賽。
- 1991 年：亞洲首支隊伍（台灣交通大學）參加世界總決賽。
- 1995 年：首度舉辦亞洲區域，並在台灣舉行，由國立政治大學辦理。
- 1996 年以前：歷年贊助廠商依序為 Apple、AT&T 和 Microsoft。
- 1997 年之後：IBM 公司為此競賽的主要贊助商。
- 1997 年：參賽隊伍 1100 隊，來自 560 個大學。總決賽地點在美國聖荷西，代表台灣的台灣大學榮獲總決賽第 4 名，這是台灣隊伍首次進入前十名。
- 2002 年：中國的隊伍首度獲得總決賽冠軍——上海交通大學。
- 2010 年：參賽隊伍 7900 隊，台灣的隊伍獲得有史以來最好的成績，為總決賽第三名——台灣大學。
- 2011 年：世界總決賽原訂於埃及沙姆沙伊赫舉行，但由於埃及當時發生暴動，因而將總

決賽地點更換為美國奧蘭多。

自 1997 年以來的參賽隊伍數量與冠軍隊伍，詳列於表 1.1。

表 1.2、表 1.3 與表 1.4 分別列出 2010 年、2011 年與 2012 年世界總決賽排名較為前面的隊伍。2010 年前 13 名的隊伍，俄羅斯與中國合計佔了 9 個；2011 年前 12 名的隊伍，俄羅斯與中國合計佔了 7 個；2012 年前 12 名的隊伍，俄羅斯與中國合計佔了 6 個。由此可以看出世界頂尖的優秀隊伍超過半數集中於俄羅斯與中國，代表這兩個國家對於程式設計的能力非常重視。

## 1.2 ACM ICPC 題目庫

ACM ICPC 舉行三十餘年，所累積的寶貴資源，就是歷次的比賽題目。有些題目已經收錄於 UVa 線上評審網站 (UVa online judge, 網址為 <http://uva.onlinejudge.org/>), 其中 UVa 乃指西班牙瓦拉多利大學 (Universidad de Valladolid), 目前累積已經超過 3600 題。全世界各地有許多人士在其上註冊帳號，進行練習，以提升程式設計能力。

該網站也會列出各題被解決的情形，以便讓人區分難易程度，如圖 1.1 所示。中間的部分代表所有送出的程式碼被線上評審伺服器認可為正確的比例。由於線上評審伺服器可隨時評審程式碼，故使用者對於尚未被評審認可的程式碼可以再次遞送，直到正確為止。圖的最右側代表該題最後遞送出正確程式碼的使用者比例。

所有 ACM ICPC 題目均有固定格式，每題包含 General Description (一般描述)、Input Format (輸入格式)、Output Format (輸出格式)、Sample Input (輸入範例)、Sample Output (輸出範例) 共五大部分。每題長度大約一至三頁左右。圖 1.2 為題目範例。

雖然收錄於 UVa 網站的 ACM ICPC 題目庫，對於每一題均有遞送正確程式碼的百分比，以及正確解題的使用者百分比，但仍不足以完全分辨其難易程度。我們為了讓學習者可以瞭解適合練習的題目，並讓教師可以配合授課課程內容做為學生之實作或測驗題目，乃將題目區分為五個等級，如下所示：

- 一顆星(level 1)：學習完計算機概論之後即可解答 (專家級設計師大約可於 10 分鐘撰寫完畢)。
- 兩顆星(level 2)：學習完資料結構之後才能解答或是苦工題 (專家級設計師大約可於 10 至 30 分鐘撰寫完畢)。
- 三顆星(level 3)：需良好的演算法或數學方法才能解答 (專家級設計師大約可於 30 至 100 分鐘撰寫完畢)。
- 四顆星(level 4)：需要特殊的演算法或是綜合多種演算法才能解答 (專家級設計師需要超過 100 分鐘才能撰寫完畢)。
- 五顆星(level 5)：超越四顆星的極特殊題目。