

# Bebbras



國際運算思維能力測驗

bebras.csie.ntnu.edu.tw



2015 題庫與詳解





# 目錄

|                       |     |
|-----------------------|-----|
| 目錄.....               | i   |
| 國際運算思維能力測驗說明.....     | iii |
| 1 海狸家族.....           | 1   |
| 2 煙火.....             | 3   |
| 3 QB 碼.....           | 7   |
| 4 剪刀 石頭 布 蜥蜴 史巴克..... | 9   |
| 5 動畫.....             | 11  |
| 6 對齊鉛筆.....           | 13  |
| 7 碗工廠.....            | 15  |
| 8 二序列.....            | 17  |
| 9 香辣茄汁雞.....          | 19  |
| 10 轉彎.....            | 21  |
| 11 海狸鍊金師.....         | 25  |
| 12 棒棒糖.....           | 27  |
| 13 間諜.....            | 29  |
| 14 海賊.....            | 31  |
| 15 繪圖機器人.....         | 35  |
| 16 畫線機器人.....         | 37  |
| 17 劇院.....            | 39  |
| 18 搜集木頭.....          | 41  |
| 19 撿石頭.....           | 43  |
| 20 羅盤.....            | 45  |
| 21 越野賽.....           | 47  |





# 國際運算思維能力測驗



國際運算思維能力測驗（Bebras International Challenge on Informatics and Computational Thinking）幫助了解學生的運算思維能力（computational thinking skill）。本測驗於每年 11 月中的國際測驗週（World-Wide Bebras Week）舉行，旨在激起學生對於資訊科學之興趣，同時了解學生是否具備資訊科學之性向。本測驗利用淺顯易懂的方式呈現題目；每一題皆為情境式任務，讓學習者利用自己既有的知識進行解題。



- 激發學生對資訊科學之學習興趣

國際運算思維能力測驗之目的除了瞭解學生是否具備學習資訊科學之性向，更希望對學生介紹資訊科學或資訊教育的基本概念，激發他們的學習興趣。利用情境式的題型，讓學生瞭解生活中隨處可見資訊科學概念之運用；而解謎推理的題目敘述方式，更可以提升學生思考動機，並增進學生深層思考的能力。

- 提升學生利用資訊方法解決問題之能力

國際運算思維能力測驗題目包含家庭生活、團體合作、工作情境等。測驗內容多樣化，透過題目讓學生了解生活中的許多問題都能透過資訊科學之概念解決。

- 降低學生對資訊科學之恐懼

國際運算思維能力測驗將抽象的資訊科學題目具體化，呈現為日常生活中會碰到的問題，使未曾受過資訊科學教育的學生亦能利用邏輯、推理、運算等能力解題。另一方面，測驗內容有趣且生動，有助於降低學生對資訊科學的懼怕感。



## 測驗對象

國際運算思維能力測驗每年於 11 月施測，受測學生並無特定資格限制，凡是三年級至十二年級之在學學生(年紀約 9 至 18 歲)，皆能參與。受測學生依年齡分為五組，分別為：Little Beaver、Benjamin (初級)、Cadet (中級)、Junior (中高級) 及 Senior (高級) 組。每組之考題又分難、中、易三種等級。臺灣目前僅開放中級組及高級組之測驗。



## 計分方式

國際運算思維能力測驗依題目之難度計分：答對給分、答錯扣分，略過不答則不給分亦不扣分；為了避免負分，測驗之起始分數為扣分之總和。我國施行之測驗每次 15 題，各難度平均分配 5 題，依下表進行給分或扣分；故起始分數為 60 分，最低 0 分，最高 300 分。

| 難度 | 易  |    | 中  |    | 難  |    |
|----|----|----|----|----|----|----|
|    | 正確 | 錯誤 | 正確 | 錯誤 | 正確 | 錯誤 |
| 給分 | 12 | -3 | 16 | -4 | 20 | -5 |



## 本書說明

本書中每一題正文的上方皆有一個表格，其表示資訊如下所示：

| 年齡組別 | 該題於各年齡組之難易度 | 該題於各年齡組之答對率 | 該題於各年齡組之平均答題時間 |
|------|-------------|-------------|----------------|
| 中高級組 | 難 中 易       | ☑ 73.10%    | 🕒 1:40         |
| 高級組  | 難 中 易       |             |                |



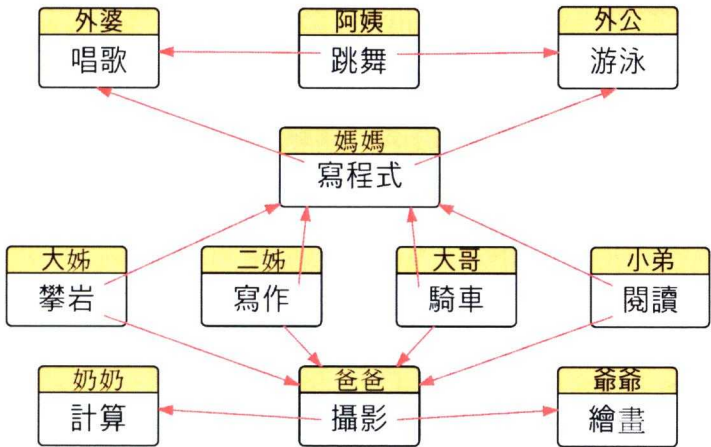
|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 73.10% | ⌚ 1:40 |
| 高級組  | 難 中 易 |          |        |



1 海狸家族

在海狸家族裡大家都各自有一些技能。女兒會承襲媽媽的所有技能，兒子會承襲爸爸的所有技能。除了那些繼承的技能以外，每個家庭成員都有一個額外的技能。

下圖顯示某海狸家族之間的關係，圖中媽媽除了承襲外婆唱歌的技能外，也擁有寫程式的技能。二姊除了承襲媽媽的兩種技能外，還增加了寫作的技能。



- 下列何者正確？
- (A) 大姊會閱讀、寫程式和歌唱
  - (B) 大哥承襲了奶奶的計算能力
  - (C) 阿姨會舞蹈和游泳
  - (D) 大哥會騎車、繪畫和攝影



## 正確答案是D

- 選項 D 是正確的，大哥承襲了爺爺的繪畫技能和爸爸的攝影技能。
- 選項 A 是錯的，大姊無法承襲小弟的閱讀技能。
- 選項 B 是錯的，因為規則是兒子只能承襲爸爸的技能，所以奶奶的技計算能力無法傳給大哥。
- 選項 C 是錯的，阿姨沒辦法繼承爸爸的游泳技能。



## 在資訊學科 上的意義

在「物件導向」程式設計中，繼承是一個重要的觀念，在規劃程式時，透過繼承可以讓程式碼被重複使用或延伸。例如，我們設計「狗」是一個類別，裡面包含了所有狗共有的資訊，當我們再仔細設計「臘腸狗」和「柴犬」時，就可以透過繼承「狗」類別，得到各種狗共通的部分，相同的程式就不用重複再寫一次。繼承除了能讓程式碼重複使用之外，還能为各類別新增額外功能。



## 關鍵字

程式設計、物件導向、繼承



## 相關網頁

[http://en.wikipedia.org/wiki/Inheritance\\_\(object-oriented\\_programming\)](http://en.wikipedia.org/wiki/Inheritance_(object-oriented_programming))

中高級組

高級組

難 中 易

難 中 易

48.69%

2:37

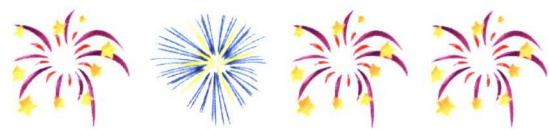


2 煙火

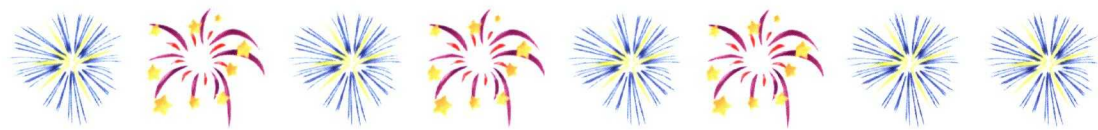
有兩隻海狸分別住在中間隔著廣大森林的兩端，他們利用燃放煙火到空中來傳遞訊息給對方。海狸只會五個詞彙，他們利用兩種煙火編排成下列組合來表達這五個詞彙：

| 詞彙 | 煙火編排組合                                                                                                                                                                                                                                                   |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 木材 |                                                                                        |
| 樹  |       |
| 石頭 |    |
| 河  |                                                                                    |
| 食物 |                                                                                                                                                                       |

例如若要發出「食物、木材、食物」的訊息，海狸會燃放下方的煙火：



請問下圖的煙火排序可以解讀成幾種意思？



- (A) 0

(B) 1

(C) 2

(D) 3

(E) 4



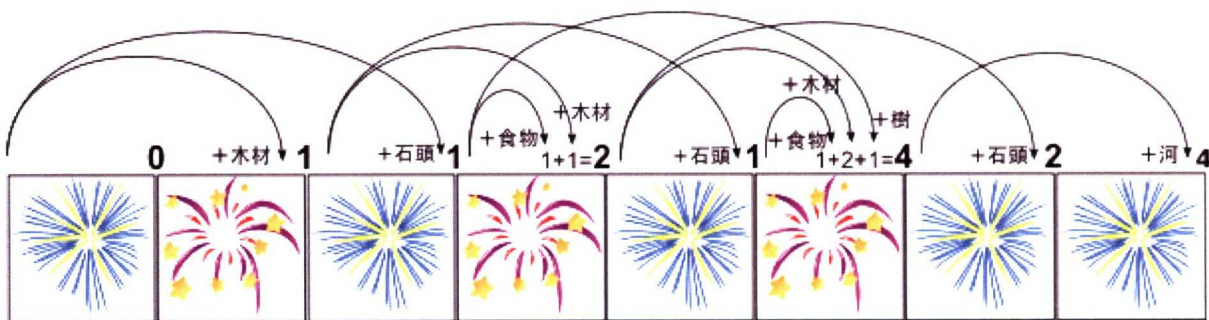


其可能的意思為：

- 木柴、石頭、食物、河
- 木柴、木柴、木柴、河
- 石頭、樹、河
- 石頭、食物、木柴、河

為了證明沒有其他的可能，你可以有系統地由第一個煙火開始計算：

1. 第一個煙火不代表任何一個訊息，所以寫 0。
2. 前兩個煙火合在一起只能代表「木材」，所以寫下 1。
3. 前三個煙火可能是一個訊息，也可能是兩個訊息合在一起。分別檢視三個煙火，新增的第三個煙火沒有任何意思，但三個煙火一起看則表示「石頭」的意思，所以寫下 1。
4. 加入第四個煙火就變有趣了。可能是第 2 點中的「木材」後面加上另一個「木材」；也有可能是第 3 點的「石頭」後面加上「食物」。目前可能性總數是  $1+1=2$ 。
5. 以此類推，每遇到新的煙火就回頭檢視前一個、前兩個、前三個煙火，看看是否新增「正確」的字彙。如果之前的訊息可以被延長，就用箭頭表示延續，可能性的總數也跟著往後移。
6. 最後一個箭頭所帶來的總數就是答案。



如此有系統的分段分析，並利用前一段的結果解決下一段問題，這種方法稱做動態規劃，它使過程變得比較簡單。



## 在資訊學科 上的意義

電腦中所有的數位訊息皆使用二進位表示，也就是說訊息只由 1 與 0 組合而成。透過一連串 0 與 1 的組合（如同本題中的「詞彙」），能表達多種不同的訊息，但由於電腦訊息不像一般文字有標點符號斷句，當一連串的 0 與 1 出現時，我們怎麼知道哪些 0 與 1 是同一組字呢？為了解決這個問題，大部份編碼都使用相同的長度，透過固定長度的分割，每段訊息就只有一種含義。但若有些詞彙使用頻繁，有些很少出現，但也都使用相同長度編碼，就比較沒有效率。所以當不同詞彙出現頻率落差很大時，就會用較短的編碼來表示常用的詞彙（例如此題的「食物」），用較長的編碼表示不常用的詞彙（例如此題的「石頭」）。這樣一來，訊息的長度就會降低。而為了解決不同長度編碼所帶來訊息斷句不清的問題，「前置碼」

（[http://en.wikipedia.org/wiki/Prefix\\_code](http://en.wikipedia.org/wiki/Prefix_code)）的設計可以解決此問題。前置碼的編碼方式要求每一個詞彙的編碼不能為其他詞彙的開頭，例如我們將 X 編碼為 01，則 010 不能拿來作為 Y 的編碼，這樣的編碼達到縮短頻繁出現的訊息而不模糊意思之目的。



## 關鍵字

編碼、前置碼、變數長度、壓縮



## 相關網頁

[http://en.wikipedia.org/wiki/Variable-length\\_code](http://en.wikipedia.org/wiki/Variable-length_code)

[http://en.wikipedia.org/wiki/Ambiguous\\_grammar](http://en.wikipedia.org/wiki/Ambiguous_grammar)





|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 63.25% | ⌚ 2:27 |
| 高級組  | 難 中 易 | ☑ 56.15% | ⌚ 2:22 |



3 QB 碼

海狸想將數字編碼，因此開發了 Quick-Beaver Code ( QB 碼 ) ，一種九宮格的圖形碼。每個格子都有個固定的值；其值如右圖所示，從右下方開始，由右至左再由下往上，每一格的值為前一格的值的兩倍。

為了要加密數字，海狸塗黑了某些格子，每個 QB 碼所代表的數字就是所有黑格子中數字的總和。例如下圖 QB 碼代表數字 17：

|     |     |     |
|-----|-----|-----|
| ... | ... | ... |
| ... | ... | 8   |
| 4   | 2   | 1   |

|  |   |   |
|--|---|---|
|  |   |   |
|  | ■ |   |
|  |   | ■ |

下方有個 QB 碼，請問如何旋轉它才能得到最大的值？

|   |   |   |
|---|---|---|
|   | ■ |   |
| ■ |   |   |
|   |   | ■ |

(A)

|   |   |   |
|---|---|---|
|   |   | ■ |
| ■ |   |   |
|   | ■ |   |

(B)

|   |   |   |
|---|---|---|
|   | ■ |   |
|   |   | ■ |
| ■ |   |   |

(C)

|   |   |   |
|---|---|---|
| ■ |   |   |
|   |   | ■ |
|   | ■ |   |

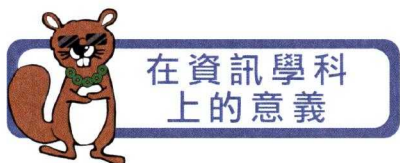
(D)

|   |   |   |
|---|---|---|
|   | ■ |   |
| ■ |   |   |
|   |   | ■ |



有注意到嗎？其實不用計算就可以得到答案。

仔細觀察九宮格，左上格的值為 256，是九個格子中值最大的。而其他每一格的值加總起來（ $128+64+32+16+8+4+2+1=225$ ）還是比左上格小。所以若要找出值最大的 QB 碼，就是由左上開始一一列由左到右尋找，找到第一個有被填滿的那一個 QB 碼。



QR 碼應用的範圍很廣，可以用在商務包裹查詢、娛樂、交通票券、產品行銷和商品標籤。日常生活中，我們可以使用手機掃描 QR 碼來收取簡訊、新增聯絡人、打開網頁，撰寫 e-mail。

在讀取 QR 碼時，若方位角度不同，所得到的圖像也會不同。故要先定位才會得到正確的值。下圖為 QR 碼的範例，QR 碼透過左上、右上、及左下的三個黑色正方形來定位，因此不管任何方向讀取都可以被辨識。



此 QR 碼是英文版維基百科的首頁網址 <http://en.m.wikipedia.org>



QR 碼、二進位系統、旋轉



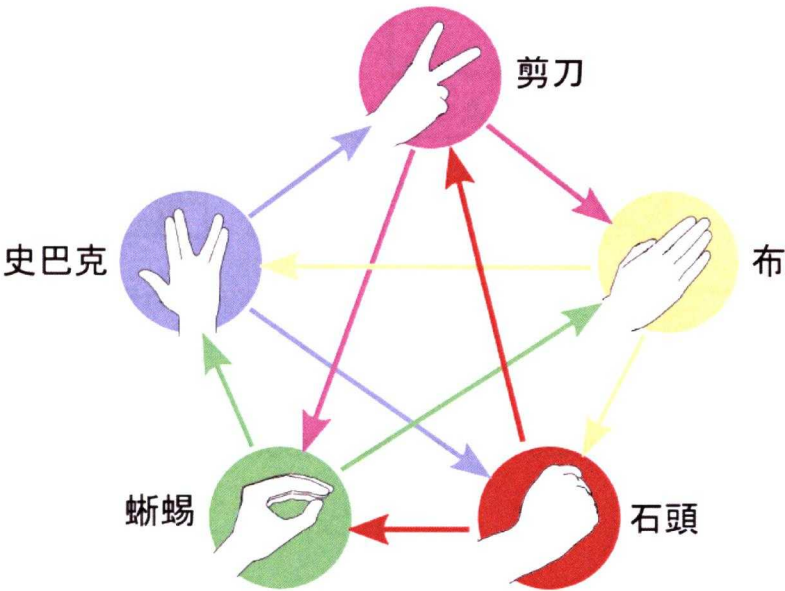
[http://en.wikipedia.org/wiki/QR\\_code](http://en.wikipedia.org/wiki/QR_code)

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 75.72% | ⌚ 2:02 |
| 高級組  | 難 中 易 |          |        |



4 剪刀 石頭 布 蜥蜴 史巴克

「剪刀石頭布」是個玩家必須在同時從剪刀、石頭、布中選擇一種武器的遊戲。石頭贏過剪刀，剪刀贏過布，布贏過石頭。不管玩家出什麼武器，只會有兩個結果：平手或者一方贏過另一方。海狸薛爾登和雷傑在玩「剪刀石頭布」的變種--「剪刀石頭布蜥蜴史巴克」。其規則如下圖：圖中有五個不同的手勢分別代表剪刀、石頭、布、蜥蜴、史巴克。箭頭指向每個手勢可以打敗的另一個手勢。例如：石頭贏過蜥蜴。如果某一局結果為平手，則玩家各得 0 分；若有一方得勝，則贏的一方得 1 分。



海狸薛爾登和雷傑玩了五局，薛爾登出拳的順序為：史巴克→蜥蜴→史巴克→石頭→史巴克。而雷傑出拳的順序為：史巴克→布→剪刀→剪刀→剪刀。

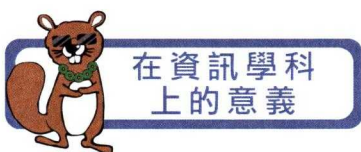
請問下列何者是最後的分數比（薛爾登 - 雷傑）？

- (A) 4 - 0
- (B) 5 - 0
- (C) 4 - 1
- (D) 3 - 2
- (E) 1 - 3





從各局的結果分別來看：第一局兩人平手（史巴克對上史巴克），第二局薛爾登勝利（蜥蜴吃掉布），第三局薛爾登贏（史巴克消滅剪刀），第四局還是薛爾登贏（石頭砸壞剪刀），最後第五局依然是薛爾登贏（史巴克消滅剪刀）。故四比零。



日常生活中，有時冗長的文字敘述不如用「圖」來表示清楚。特別的是，資訊科學中的「圖」並不是指一般的圖片或圖畫，而是利用點(vertex)和邊(edge)所形成的結構，可以用來表示關聯或關係，像是本題中各種手勢和彼此之間的勝負關係。在有了結構化的表示法後，更可以進一步用來計算或推論。



圖形



此遊戲的官網：<http://www.samkass.com/theories/RPSSL.html>

中高級組

難 中 易

☑ 79.24%

⌚ 1:28

高級組

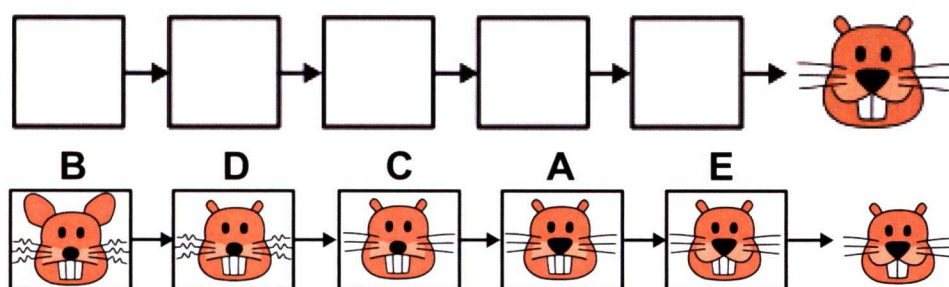
難 中 易



## 5 動畫

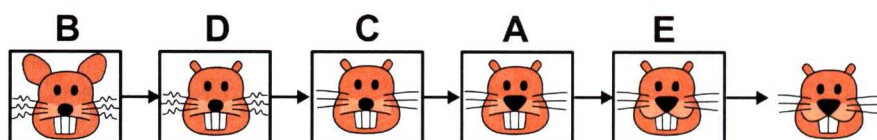
海狸太郎在製作一部臉的動畫，動畫由六張照片組成。若照片順序正確，動畫中的臉一次會改變一個部位，而且只改變一個部位。

不幸的是照片的順序被打亂了，海狸太郎只知道最後一張照片是哪一張，他必須將前面五張照片 A~E 依正確的順序排好。



請問這五張相片的正確順序為何？

- (A) D→B→E→C→A
- (B) C→B→D→A→E
- (C) D→B→C→E→A
- (D) B→D→C→A→E



每張照片和上一張照片不同的地方如下所述--

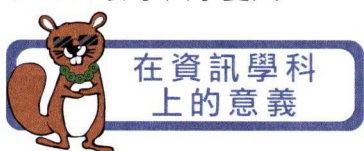
B→D：耳朵由大變小

A→E：嘴巴由放鬆變為微笑

D→C：鬍鬚由捲變直

E→最後一張照片：牙齒由三顆變為兩顆

C→A：鼻子由小變大



要找出照片間的差異，可以先把照片中決定長相的重要特徵擷取出來成為屬性，以及每個屬性可能的值列舉出來，例如：

耳朵：大、小

嘴：放鬆、微笑

鼻子：大、小

牙齒：兩顆、三顆

鬍鬚：捲、直

列舉完後再對各個(屬性 - 值)來描述每張照片，找出每次只有一個變異的順序。在資訊科學中，常常會把真實世界中的事物轉換成有「屬性」和「值」的物件來表示，透過比對相同屬性之間值的差異，可以計算事物彼此間的相似度或關聯性。



結構化數據、物件導向



[https://en.wikipedia.org/wiki/Attribute-value\\_system](https://en.wikipedia.org/wiki/Attribute-value_system)

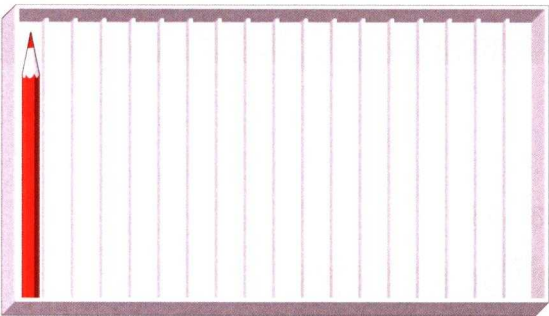
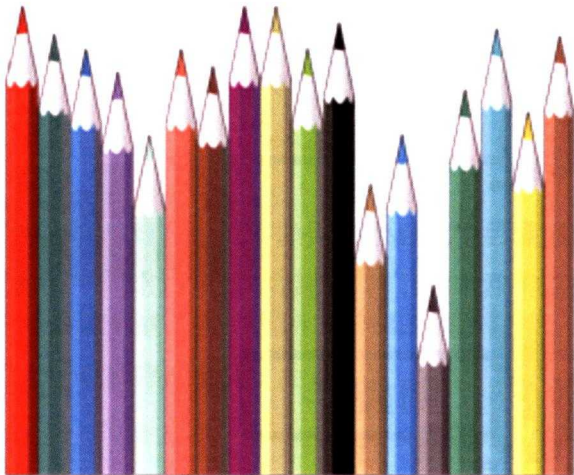
|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 67.18% | ⌚ 2:19 |
| 高級組  | 難 中 易 | ☑ 54.90% | ⌚ 2:23 |



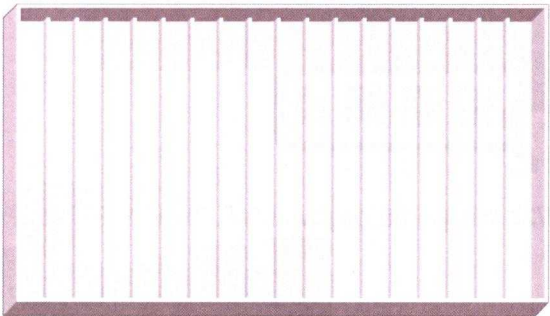
6 對齊鉛筆

有一隻小海狸覺得畫畫很無聊，就開始玩彩色鉛筆，他由左到右從盒子裡一支支取出彩色鉛筆，再依照下列規則由左至右把筆一支支放入爸爸或媽媽的鉛筆盒裡：

1. 第一支筆放在媽媽的鉛筆盒裡。
2. 其他鉛筆依序和前一次放入媽媽鉛筆盒中的筆相比，如果比前一次放入媽媽鉛筆盒的筆還要短，便一樣放在媽媽的鉛筆盒，反之則放入爸爸的。

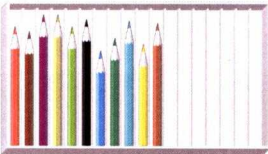


媽媽

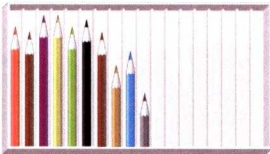


爸爸

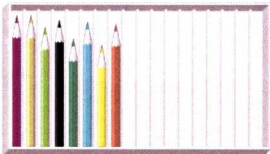
請問小海狸放完所有鉛筆後，爸爸的鉛筆盒會是什麼樣子？



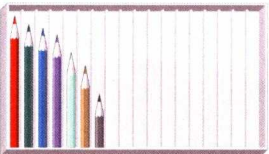
(A)



(B)



(C)



(D)





### 正確答案是A

因為第二支筆比第一支短，第三支比第二支短，第四支比第三支短，第五支比第四支短，所以把前五支鉛筆依序放進媽媽的鉛筆盒。接下來的四支鉛筆(第六至第九支筆)都比第五支筆長，依規則將它們都放入爸爸的鉛筆盒。第十支筆比第五支短，所以把它放到媽媽的鉛筆盒。剩下的鉛筆都比第十支長，所以都要放到爸爸的鉛筆盒。因此，選項 D 是最終媽媽的鉛筆盒的樣子，選項 A 則是爸爸的鉛筆盒的樣子。



### 在資訊學科 上的意義

這題題目中所描述的規則，就是演算法的範例。在日常生活中，我們常常必須遵循一些規則以完成生活中的任務，而這些規則也是就一種演算法。



### 關鍵字

演算法



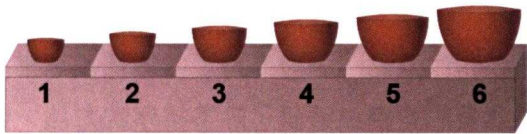
### 相關網頁

<https://zh.wikipedia.org/wiki/%E7%AE%97%E6%B3%95>

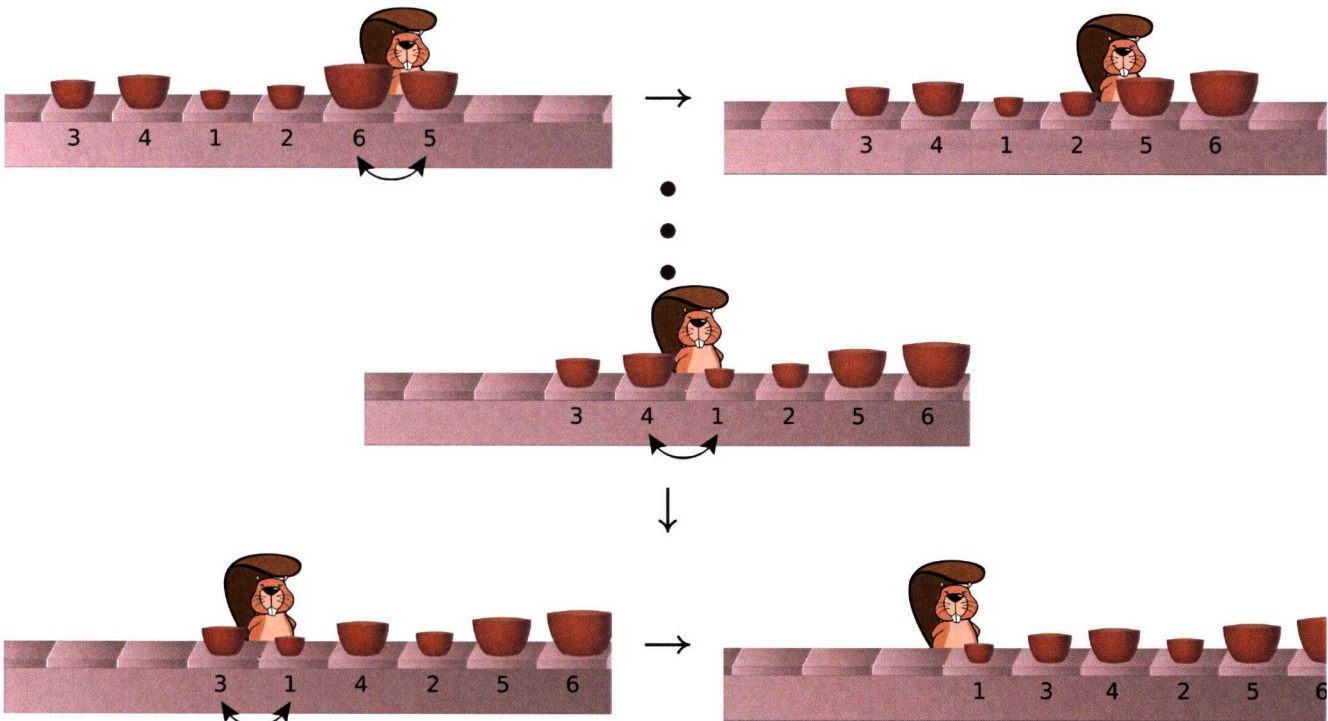
|      |       |          |      |
|------|-------|----------|------|
| 中高級組 | 難 中 易 |          |      |
| 高級組  | 難 中 易 | ☑ 19.03% | 3:08 |

## 7 碗工廠

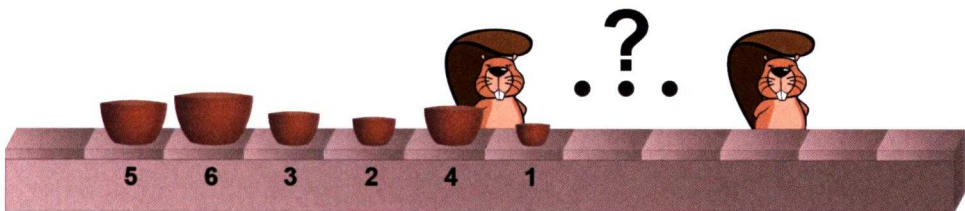
有間工廠生產一組六種大小不同的碗，碗都放置在傳送帶上由左向右一個個傳送。工廠會將同一組的碗集合後一起放到傳送帶上，隨意擺放。而在包裝這些碗之前，須將這些碗如下圖依小到大的順序排好：



工廠安排工人在傳輸帶旁將碗排序：每當相鄰的兩個碗經過工人面前時，若碗的順序有誤，他會將兩個碗交換位置。下圖為一組碗經過一位工人時碗順序的變化：

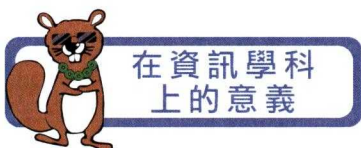


請問需要幾個工人才能將下圖的碗按順序排好？





如圖所示，該組碗的原本順序是：5, 6, 3, 2, 1, 4，而交換碗的順序是由右至左，所以經過第一個工人後，碗的順序為：1, 5, 6, 3, 2, 4（交換四次，碗 1 與其他所有碗交換）；經過第二個工人後，順序為：1, 2, 5, 6, 3, 4（交換三次，碗 2 與其他所有碗交換）。通過第三個工人後，順序則是：1, 2, 3, 5, 6, 4（交換兩次，碗 3 與其他所有碗交換）；最後第四個工人就可以正確把碗的順序排成 1, 2, 3, 4, 5, 6（交換兩次，碗 4 與其他所有碗交換）。



讓電腦自動處理數據是資訊學科的重點，一般而言，當數據按照某些標準排序之後，電腦處理數據會比較快速且容易；因此資訊科學家鑽研了許多排序的方法。如題目中碗的排序方法便是泡沫排序法；泡沫排序法藉由不斷交換序列中鄰近的兩個物件，直到所有物件順序正確為止。

泡沫排序法相對於其他演算法來說比較容易理解，但不幸的是它不太有效率。當排列 1000 個物件時，在最壞狀況下大概需耗費高達五十萬個步驟才能排序完成，但好一點的排序演算法大概只需要一萬個步驟。



泡沫排序法、下沉排序



[http://en.wikipedia.org/wiki/Bubble\\_sort](http://en.wikipedia.org/wiki/Bubble_sort)

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 49.20% | ⌚ 1:42 |
| 高級組  | 難 中 易 |          |        |



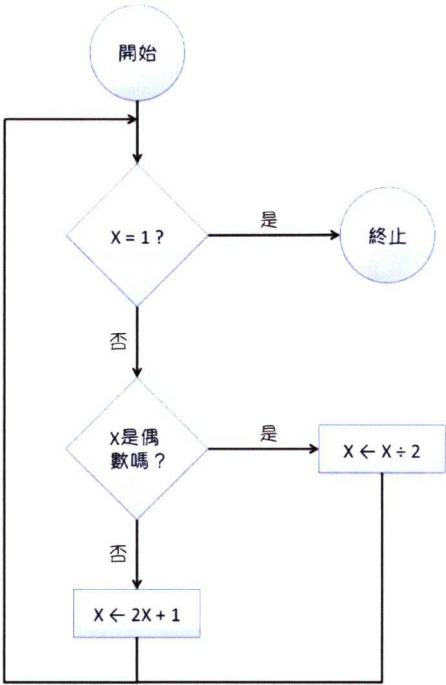
8 二序列

將任意正整數設為  $X$ ，並如圖執行以下指令：

- 步驟 1：如果  $X$  等於 1 就停止。
- 步驟 2：如果  $X$  是偶數，就將它除以 2；如果是奇數，就將  $X$  乘以 2 再加 1。
- 步驟 3：將步驟二產生出的值當作新的  $X$ ，並返回步驟一重新執行所有步驟。

請問  $X$  初始值須為何才能使以上程序終止？

- (A) 任一偶數
- (B) 2 的次方數
- (C) 無論初始值為何，此程序永遠無法停止
- (D) 任意數

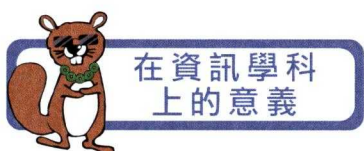






正確解答是選項 B。若要程序終止，那初始值必須為 1，或是經過該程序之後  $X$  等於 1。若初始值是非 1 的奇數，我們可以發現該程序會成為無限迴圈且  $X$  值會變得無窮大。若初始值是偶數，則程序為不斷將它除以 2，而唯有 2 的次方數才會在不斷除以 2 之後變為 1。所以只有 1 或是 2 的次方數可以使過程終止。

選項 A 中，若  $n$  為偶數但不是 2 的次方數，則  $n$  在過程中不斷倍減，直到它變成奇數後，就會成為無限迴圈不會終止。而因為此程序有可能終止，所以選項 C 錯誤。若初始值為奇數，則會變成無限迴圈，所以選項 D 也是錯的。



這題的題目是個簡單的流程圖，流程圖通常用來說明一個過程，透過視覺化的輔助，學生知道如何逐步執行與條件判斷。因此，對於程式設計的初學者來說，設計流程圖常常也被視為程式設計的前置作業，閱讀並理解流程圖是資訊科學的基本能力。在本題中，學生可發現某些初始值會讓程式無限迴圈，也是學習程式設計中很重要的一環。



奇數與偶數、程式、條件迴圈、卡拉茲猜想、無限



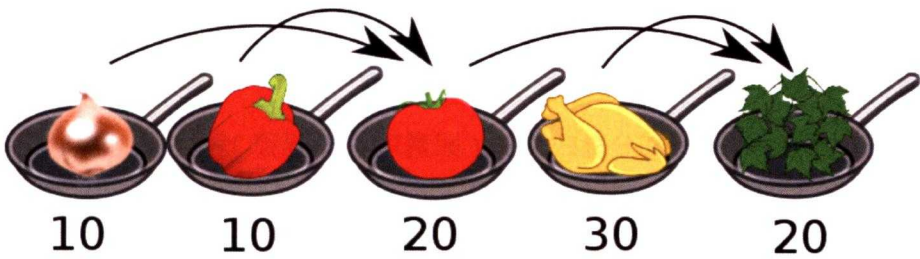
此題類似卡拉茲猜想，不過此題有解，參考網頁：

<https://zh.wikipedia.org/wiki/%E8%80%83%E6%8B%89%E5%85%B9%E7%8C%9C%E6%83%B3>

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 18.45% | ⌚ 2:15 |
| 高級組  | 難 中 易 |          |        |



9 香辣茄汁雞



海狸瑟吉喜歡做菜，他的拿手菜是香辣茄汁雞。

瑟吉喜歡用後院裡的單一火爐做菜，而且他有自己的一套順序規則：

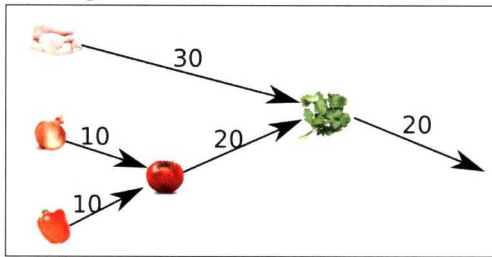
- 步驟一：( 10 分鐘 ) 燉煮一顆洋蔥
- 步驟二：( 10 分鐘 ) 燉煮一顆甜椒
- 步驟三：( 20 分鐘 ) 將煮過的洋蔥和甜椒混合再放入番茄燉煮
- 步驟四：( 30 分鐘 ) 燉煮一隻雞
- 步驟五：( 20 分鐘 ) 將步驟三跟四的食材混合，加上一些香料一起燉煮

料理香辣茄汁雞共需花費九十分鐘。當在室內裡做菜時，為了要加快速度，瑟吉會使用多個火爐來做菜。

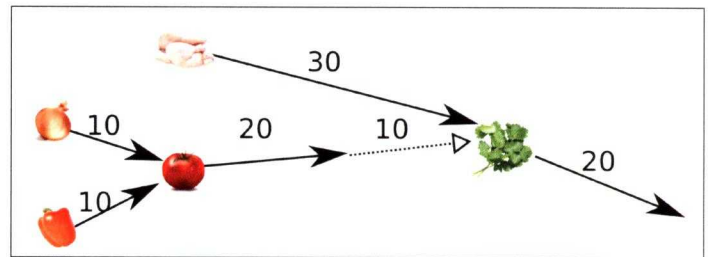
- 請問下列敘述何者錯誤？
- (A) 瑟吉使用兩個火爐可以節省 10 分鐘的烹調時間
  - (B) 瑟吉使用兩個火爐可以節省 30 分鐘的烹調時間
  - (C) 瑟吉使用三個火爐可以節省 40 分鐘的烹調時間
  - (D) 瑟吉使用四個火爐可以節省 50 分鐘的烹調時間



## 正確答案是D



利用三個鍋子烹調



利用兩個鍋子烹調

左圖說明用三個鍋子可減少 40 分鐘的時間（即選項 C 之描述），右圖則說明用兩個鍋子可減少 30 分鐘的時間（即選項 A 和 B）。



## 在資訊學科 上的意義

在此題中火爐就如同電腦的資源，當只有一項資源時，所有的工作就只能一件一件依序處理；若擁有多項資源，則可以將幾項工作同時處理，如何安排工作的順序和使用資源的分配，達到最佳的效率？這部分就是資訊科學中排程和平行運算的議題。



## 關鍵字

排程、平行運算



## 相關網頁

[https://en.wikipedia.org/wiki/Program\\_optimization](https://en.wikipedia.org/wiki/Program_optimization)

[http://en.wikipedia.org/wiki/Parallel\\_computing](http://en.wikipedia.org/wiki/Parallel_computing)

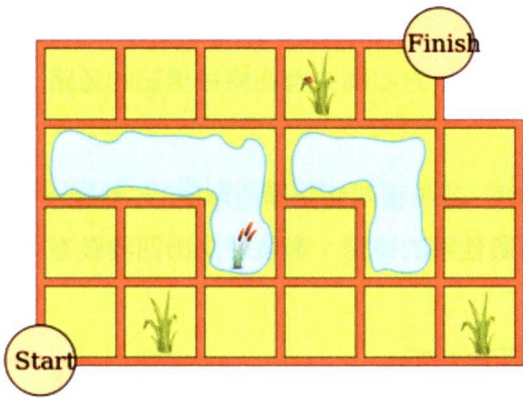
[http://en.wikipedia.org/wiki/Scheduling\\_\(computing\)](http://en.wikipedia.org/wiki/Scheduling_(computing))

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 56.95% | ⌚ 1:49 |
| 高級組  | 難 中 易 | ☑ 48.01% | ⌚ 1:53 |



10 轉彎

有位國王很喜歡坐馬車，所以國王命令車夫遇到叉路時不可以直走，所以車夫一定要向右轉或向左轉，即使叉路是山澗小路。右圖是這個國家的地圖，每兩個相連的路口中間皆相隔一公里。



國王需要由左下角至右上角，圖中用圓形標示。然而車夫只想要盡快到達目的地。

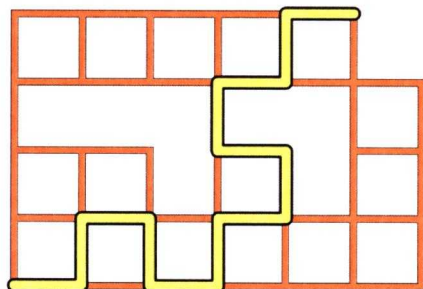
請問在不違反國王的規定下，起始點到終點最短路徑有多長？

- (A) 11 公里
- (B) 13 公里
- (C) 15 公里
- (D) 不可能在遵守規定之下到達目的地

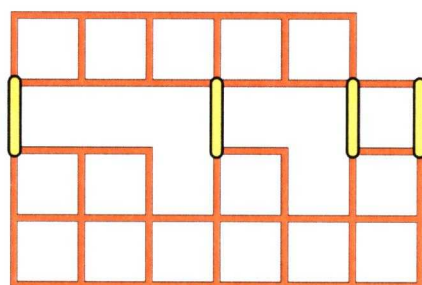




下方左圖為 13 公里的行徑路線：



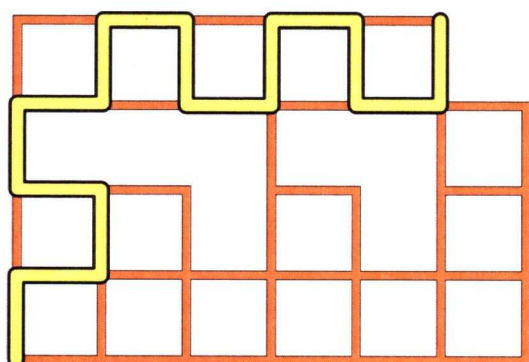
( 左圖 )



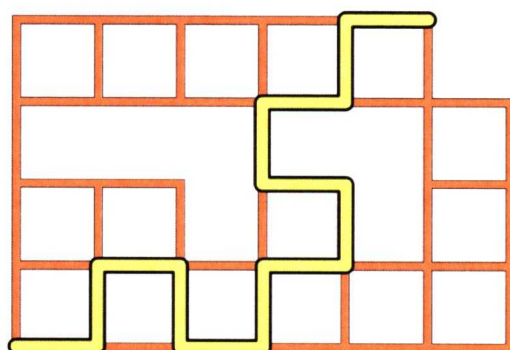
( 右圖 )

為什麼這個路徑才是最佳解？在上方右圖中有四條被標記的道路，國王的行駛路線只少會經過這四條路中的其中一條。

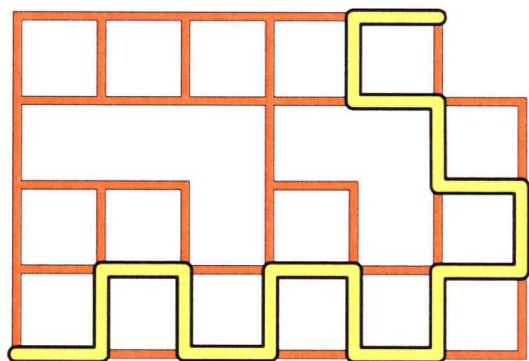
分別檢視四個可能性可以發現，若由這四條路開始解題，不論往前或往後，能有的選擇都不多。而過程中可以刪去明顯不是最佳解的道路，最後便選出四種較有可能是最短路徑的圖（如下所示），再由此選出最佳解。



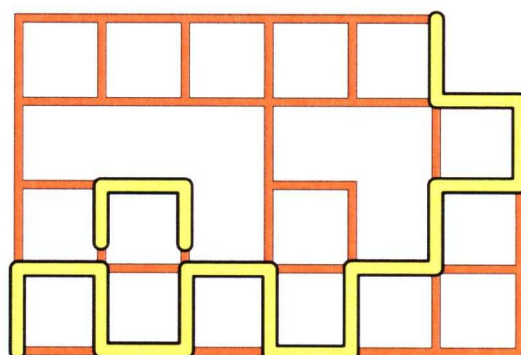
路線 1：15 公里



路線 2：13 公里（正解）



路線 3：17 公里



路線 4：15 公里



### 在資訊學科 上的意義

本題測驗學生尋找最佳路徑的能力。生活中，常常需要找出用最短時間或最低成本在兩點之間移動的變法。而因為這一題有些條件限制，所以比一般路線搜尋較複雜。在現實生活中，司機也會受到某些限制，譬如說某些交叉路口會禁止左轉。



### 關鍵字

最短路徑、限制路徑



### 相關網頁

[http://en.wikipedia.org/wiki/Shortest\\_path\\_problem](http://en.wikipedia.org/wiki/Shortest_path_problem)



|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 65.06% | ⌚ 1:49 |
| 高級組  | 難 中 易 | ☑ 55.25% | ⌚ 1:54 |

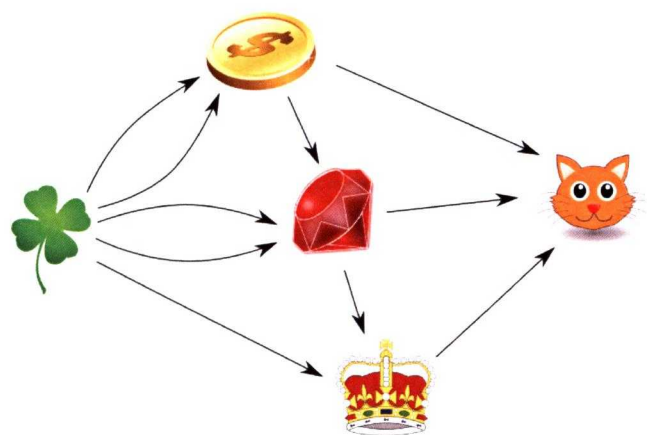


## 11 海狸鍊金師

海狸鍊金術師可以把某種物品轉換成另一種物品；下列為他可以轉換的物品：

- 兩朵幸運草轉換成一枚硬幣
- 一枚硬幣和兩朵幸運草轉換成一顆紅寶石
- 一顆紅寶石和一朵幸運草轉換成一頂皇冠
- 一枚硬幣、一顆紅寶石和一頂皇冠轉換成一隻小貓。

要注意的是，一旦轉換成另一個物品後，原本的物品將會消失。



請問海狸鍊金術師需要幾朵幸運草來合成一隻小貓？

- (A) 5
- (B) 10
- (C) 11
- (D) 12





轉化方式如下：

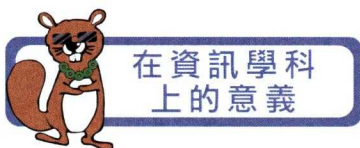
1 枚硬幣 = 2 朵幸運草

1 顆紅寶石 = 2 朵幸運草 + 1 枚硬幣 = 4 朵幸運草

1 頂皇冠 = 1 顆紅寶石 + 1 朵幸運草 = 4 朵幸運草 + 1 朵幸運草 = 5 朵幸運草

1 隻小貓 = 1 枚硬幣 + 1 顆紅寶石 + 1 頂皇冠 = 2 朵幸運草 + 4 朵幸運草 + 5 朵幸運草 = 11 朵幸運草

選答 5 朵幸運草的同學並未讀懂題目，不知道無論合成一枚硬幣或一顆紅寶石都需要兩朵幸運草，或是不知道合成一顆紅寶石需要一枚硬幣來，合成一頂皇冠需要一顆紅寶石。選答 10 朵的同學誤以為所需的幸運草總數與邊總合相同。選答 12 朵的同學則是計算錯誤。



這個題目展示了如何用圖表示項目間的相依性。在資訊學裡，圖是一種包含點(vertex)和邊(edge)的結構，常用來表示項目之間的關係。比起以純文字敘述關係，圖能使工作內容更加具體化，在有了結構化的表示法後，更可以進一步用來計算或推論。



圖



[http://www.wikiwand.com/en/Graph\\_\(abstract\\_data\\_type\)](http://www.wikiwand.com/en/Graph_(abstract_data_type))

|      |       |                                            |        |
|------|-------|--------------------------------------------|--------|
| 中高級組 | 難 中 易 | <input checked="" type="checkbox"/> 16.50% | ⌚ 2:22 |
| 高級組  | 難 中 易 | <input checked="" type="checkbox"/> 15.43% | ⌚ 2:25 |



12 棒棒糖

海狸叔叔開了一間棒棒糖店，店裡一支棒棒糖售價海狸幣 12 元。包裝好的棒棒糖則分為一袋 2 支裝售價 20 元、一袋 4 支裝售價 44 元、一袋 8 支裝 72 元，另外還販售一箱 16 支裝 150 元。



請問如果在可以將多出來的棒棒糖分送出去的前提之下，買 21 支棒棒糖最少需要多少海狸幣？

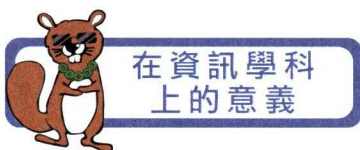


若我們檢視不同包裝裡的棒棒糖價格：

- 一支棒棒糖海狸幣 12 元
- 一袋兩支裝的棒棒糖每支  $20 \div 2 = 10$  元
- 一袋四支裝的棒棒糖每支  $44 \div 4 = 11$  元
- 一袋八支裝的棒棒糖每支  $72 \div 8 = 9$  元
- 一箱十六支裝的棒棒糖每支  $150 \div 16 = 9.375$  ( 小於 10 但大於 9 )

單支棒棒糖的單價最高，所以買超過一支是沒有意義的。四支裝和十六支裝簡直是在敲詐：買兩包兩支裝的比買一包四支裝的還划算，買兩包八支裝的也比較划算，而且一包八支裝的比四包兩支裝更便宜。所以我們跳過這兩種組合，再來試試下面幾種買 21 隻棒棒糖的搭配方式：

- 三包八支裝的總價  $72 \times 3 = 216$
- 兩包八支裝和三包兩支裝的總價  $72 \times 2 + 20 \times 3 = 204$
- 兩包八支裝、兩包兩支裝及一支棒棒糖的總價  $72 \times 2 + 20 \times 2 + 12 = 196$



這題是資訊科學家口中的「背包問題」的一個例子。

背包問題是種題型的總稱，這類題目要求用不同大小或金額的物品填滿背包；而且有很多種的變形：有時會規定每種物品只能拿一個，有時切割物品是被允許的，有時物品的大小必為整數.....。

在現實中，資訊科學家研究這類問題不是為了整理背包，而是為了優化記憶體的使用、確保通訊安全或是解決其他看似無關連的問題。



背包問題



[http://en.wikipedia.org/wiki/Knapsack\\_problem](http://en.wikipedia.org/wiki/Knapsack_problem)

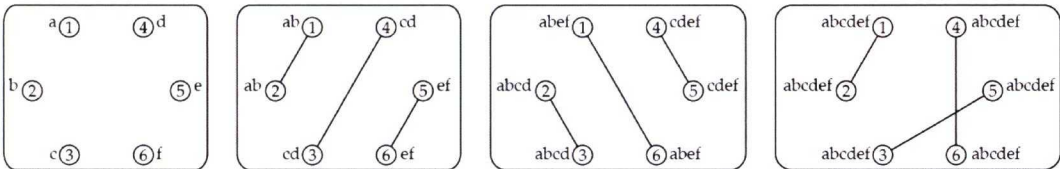
|      |       |                                                                                           |
|------|-------|-------------------------------------------------------------------------------------------|
| 中高級組 | 難 中 易 | <input checked="" type="checkbox"/> 28.69% <span style="margin-left: 20px;">⌚ 2:08</span> |
| 高級組  | 難 中 易 |                                                                                           |



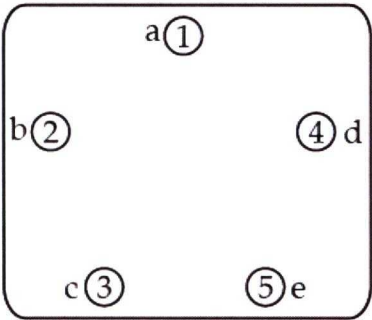
13 間諜

六位間諜們會在每個星期五交換他們在當週所蒐集到的資訊。為了避免被察覺，每位間諜不能同時和一位以上的間諜見面，所以他們必須分批進行兩人會議，來交換截至當下兩人所得到的所有訊息。

間諜 1 號至間諜 6 號這六位間諜，只需進行三輪的會議就能交換所有的秘密：如下圖，會議前每位間諜都握有一項訊息（如：間諜 1 號知道訊息“A”，間諜 2 號握有訊息“B”）。在第一輪間諜 1 號和間諜 2 號會面並交流資訊，所以他們現在都握有訊息“A 和 B”。圖中用一條線來代表兩位間諜之間的會議，也指出每位間諜所擁有的資訊。經過三輪的會面後，所有的資訊都傳達完畢。



一場國際事件後，其中一位間諜從此缺席。請問剩下的五位間諜至少需要舉行幾輪的會議，才能交換所有的訊息？





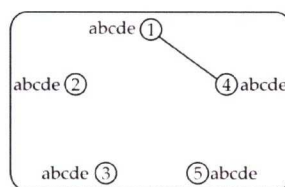
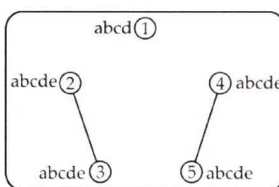
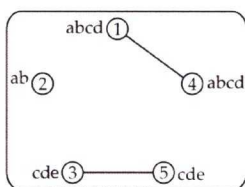
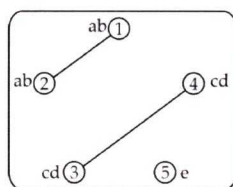


## 正確答案是 4

這個答案有點讓人出乎意料：因為減少了一位間諜，所以我們預期答案應該會是三或比三更少。何況如果只有四個間諜交換訊息，答案很明顯是兩輪，這樣一來這題的答案就更奇怪了。

然而，經過幾次錯誤的嘗試便會發現問題的根源：因為間諜的個數為奇數，每一輪中都會有一位間諜是閒置的。假設間諜 5 號未參與第一輪會議，直接參與第二輪；因此在第二輪會議後，只有兩位間諜握有到訊息 e。在第三輪中，這兩位間諜再與另外兩位間諜會面之後，只有四位間諜擁有訊息 e，還需要進行第四輪會面來將訊息 e 傳送給最後一位間諜。

因此我們證明了至少需要四輪的會議。為了顯示四輪就足夠了，我們畫了一張四輪會議的圖。



## 在資訊學科上的意義

電腦交換訊息時，時常是成對交換的。有時會出現像「如何在最短時間內在整個網路中分享所有訊息」這樣的問題，所以資訊專家需要解決類似這樣的問題。這種問題被稱作「八卦問題」（<http://mathworld.wolfram.com/Gossiping.html>）。若你試著解不同數量的間諜的問題，或許會發現它有趣的規則。

資訊科學家最早在 1975 對這類問題提出通解，這類問題和其他相似的問題都跟資訊科學中不同領域有關，如：交換資料、通訊網路和密碼學等。



## 關鍵字

八卦問題



## 相關網頁

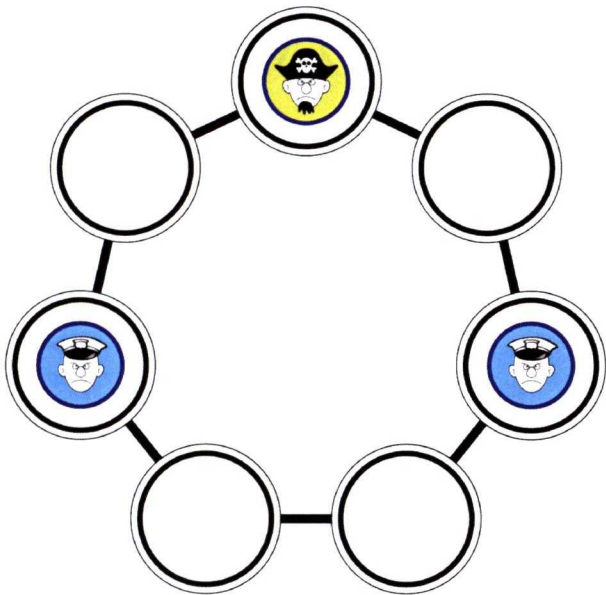
Walter Knoedel: New Gossips and Telephones, Discrete Mathematics 13 (1975),  
[http://ac.els-cdn.com/0012365X75900904/1-s2.0-0012365X75900904-main.pdf?\\_tid=9666b184-dfca-11e4-be3e-00000aabb0f6b&acdnat=1428702372\\_fd4cf501cf0c9fa9a2a41af0878acf00](http://ac.els-cdn.com/0012365X75900904/1-s2.0-0012365X75900904-main.pdf?_tid=9666b184-dfca-11e4-be3e-00000aabb0f6b&acdnat=1428702372_fd4cf501cf0c9fa9a2a41af0878acf00)

|      |   |   |   |          |        |
|------|---|---|---|----------|--------|
| 中高級組 | 難 | 中 | 易 |          |        |
| 高級組  | 難 | 中 | 易 | ☑ 18.44% | ⌚ 2:03 |



14 海賊

有一個叫「海賊獵人」的兩人桌上遊戲；下圖是這個桌遊的圖示。其中一位玩家扮演海賊，有一顆黃色棋子；另一位玩家則是海軍，有兩顆藍色棋子。兩個玩家輪流移動，海軍每一步只能移動其中一個棋子到隔壁的格子；海賊行動比較快，所以每一步總是移動兩個格子。海軍只能移動到空格子，不能移動到被海賊或另一位海軍所佔據的格子。當海賊被迫移動到被海軍所佔領的格子時（如圖），遊戲便結束且海軍勝利。也就是說只要輪到海賊移動時，棋盤呈現圖中的局勢，海軍就贏了。



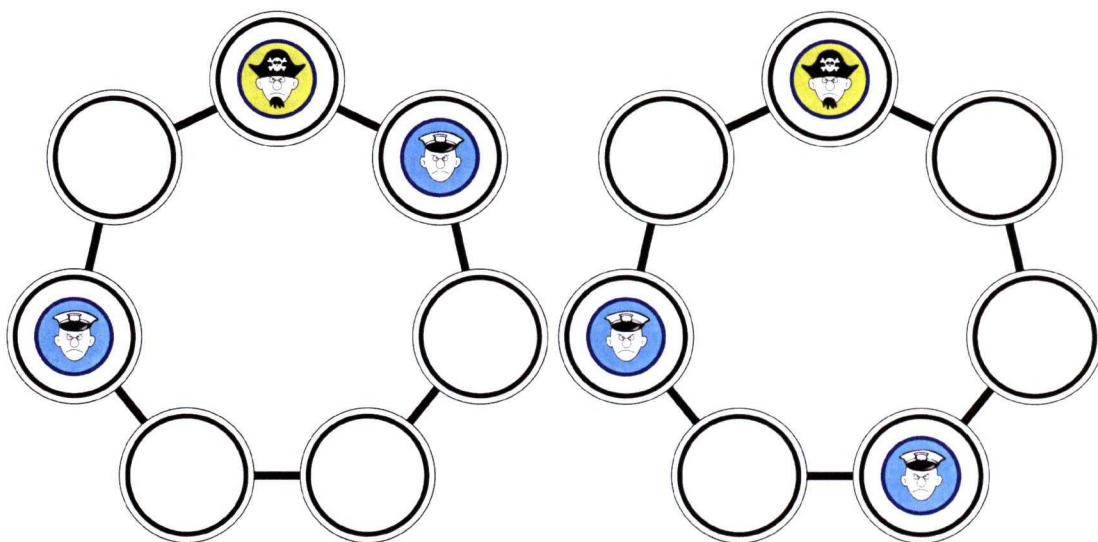
小強和吉兒常常玩「海賊獵人」。吉兒非常擅長扮演海賊，不管初始狀態為何，聰明的你要是幫著小強，要走幾步才會贏呢？

- (A) 2
- (B) 3
- (C) 5
- (D) 小強不可能贏

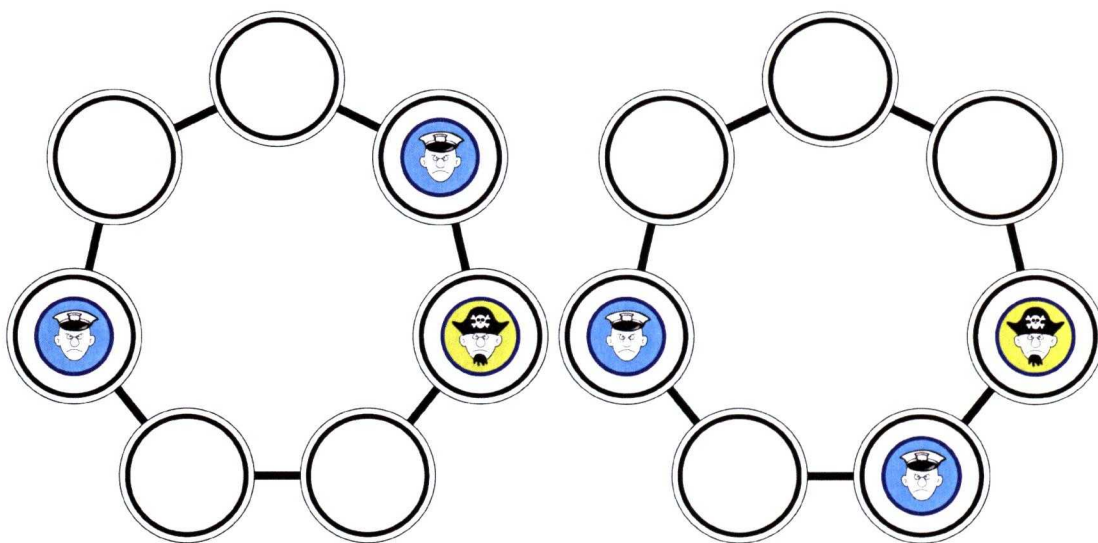


假如吉兒沒有失誤，小強就沒辦法贏。

假設小強成功使棋盤形成圖中的情況，也就是吉兒輸了。在小強走到這步的前，必須移動其中一位海軍向下或向上一步。假設前一步移動右邊的海軍；因為轉盤是對稱的，就算移動的是左邊的海軍，結果也一樣。因此，前一步會是下方兩種情況之一：



由上方兩個圖回推到上上一步：海賊必定是從右邊來。因此上上步的棋盤一定是以下其中一個：



海賊被捕的結局，只能從上方這兩種狀態（還有它們鏡射後的情況）推演而來。但是在這兩種局勢下，如果吉兒是個很厲害的玩家，一定會把海賊向左移，而非向上；因此，小強不可能贏。



## 在資訊學科 上的意義

棋盤遊戲的程式的運作方式是從遊戲狀態的「圖」計算可能的路徑。本題是從結束的狀況往回推；但實際上在設計遊戲程式時，通常是從當下的狀態為出發點，計算電腦本身以及其對手可能的移動方式。例如設計遊戲會用到的「極小化極大演算法」（Minimax），就是先假設對手移動時絕對會作最好的選擇，再由此計算電腦本身可以移動的所有路線中最好的選擇。在複雜的遊戲中，例如西洋棋，電腦至少會分析到特定的步數（有時多達 15 步），並用近似法來計算下一步的位置。



## 關鍵字

圖論、棋盤遊戲、極小化極大算法



## 相關網頁

<http://en.wikipedia.org/wiki/Minimax>

[http://en.wikipedia.org/wiki/Computer\\_chess](http://en.wikipedia.org/wiki/Computer_chess)





|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 43.15% | ⌚ 1:59 |
| 高級組  | 難 中 易 |          |        |



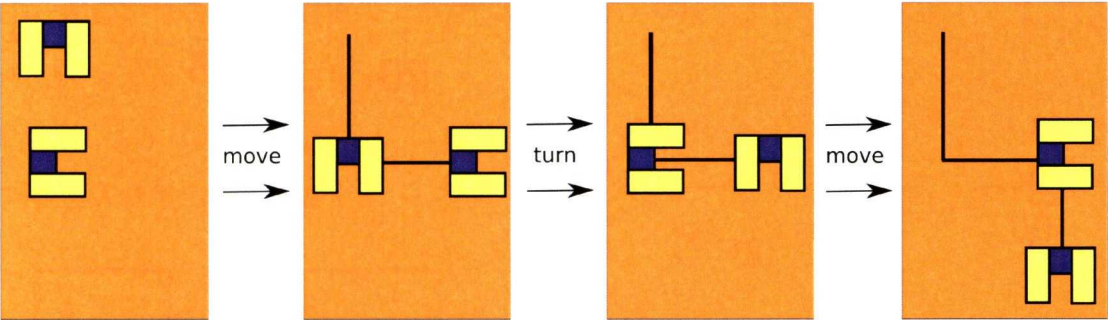
15 繪圖機器人

詹姆斯創造了兩個會依照指令繪圖的機器，它們可以執行的指令有：

move: 機器人向前一步

turn: 機器人轉動 90 度

他把兩個機器人放在地板上，並分別對每個機器人發送三個指令，如下圖：



詹姆斯注意到其中一個機器人把指令"turn"解讀成向右轉 90 度，但另一個機器人把指令"turn"解讀成向左轉 90 度。

接著詹姆斯把機器人放在地板上的不同位置來作畫，但每次都發送和上方相同的指令。而這兩個機器人不能同時出現在相同的位置。

哪張圖片不可能是這兩個機器人畫出來？

(A)

(B)

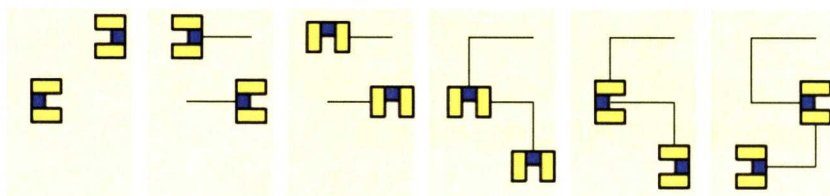
(C)

(D)

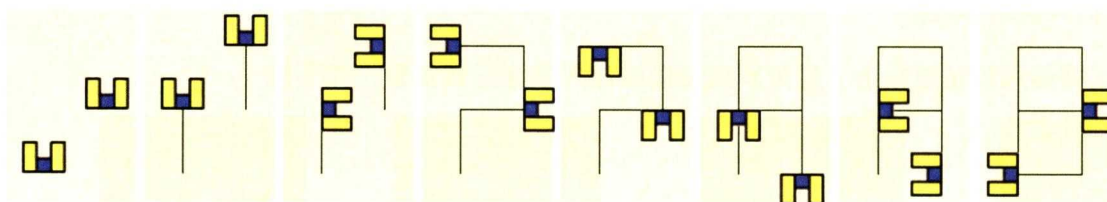


正確答案是選項 D：兩個機器人不可能畫出“L”。首先用圖展示每張圖是如何畫出來的：

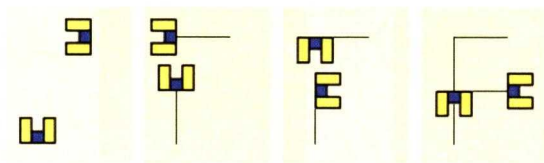
(A)



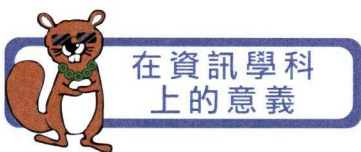
B)



C)



而要如何證明沒辦法畫出“L”呢？首先“L”左邊的線不可能被同一隻機器人畫出，因為機器人在走了一步後必須轉向，無法連續移動兩步。機器人必須轉向  $90^\circ$  再畫出左線的其中一半或是底部的線；但只要畫出左線的任意一部分，其中一個機器人若要轉向就只能轉  $180^\circ$  度，不能向左轉或向右轉。因此不可能畫出“L”。



本題中給機器人的指令是演算法的一種表現。兩個機器人對於指令解讀不同，進而形成不尋常的狀態。理解指令所造成的任何影響，就算是不尋常的情況，是構成演算法思考很重要的一環。



機器人、指令、對稱

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 65.78% | ⌚ 1:17 |
| 高級組  | 難 中 易 | ☑ 55.77% | ⌚ 1:10 |



## 16 畫線機器人

湯姆製作了一個可以畫垂直線和水平線機器人，並利用一連串的數字來操控它。

第一個數字代表機器人必須畫的垂直線長度，若為正數則向上畫，負數則向下畫。

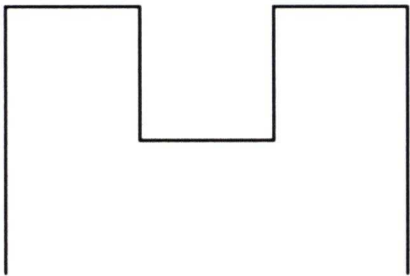
第二個數字代表要畫出的水平線長度，若為正數向右畫，負數則向左，且須從上一條線結束的地方開始畫。

第三個數字代表另一條垂直線，第四個數字代表另一條水平線，依此類推。

例如，若輸入下方這一串數列：

2, 1, -1, 1, 1, 1, -2

則機器人會畫出這樣的圖形：



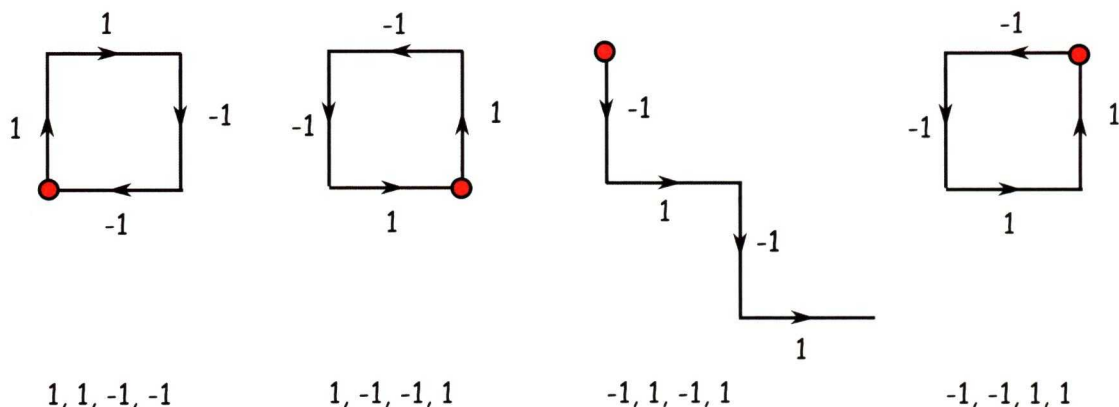
請問下列哪一組序列無法讓機器人畫出正方形？

- (A) 1, 1, -1, -1
- (B) 1, -1, -1, 1
- (C) -1, 1, -1, 1
- (D) -1, -1, 1, 1



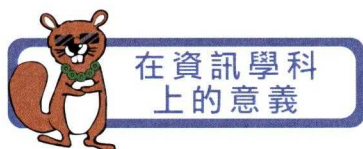


下圖顯示機器人紅點開始執行各選項的指令之後的結果：



或著以下列的角度來思考：

所有的選項都畫了四條線，所以正方形的每個邊都只被畫過一次。機器人只要移動就會畫線，所以最後一條線的終點必須為第一條線的起點。為了在第一條線的起點結束，垂直移動量，也就是第奇數個數字的總和，必須為零，且水平移動量，也就是第偶數個數字的總和，也必須為零。



繪畫指令是演算法應用的範例。演算法是計算的具體步驟的集合。在這一題中，具體步驟就是劃線的指令，這些指令為依序地一個一個地出現，並利用特殊的方式來編碼，也就是一連串的正數及負數。理解編碼後的指令對於想認真投入資訊科學的人來說，是很重要的能力。



演算法繪圖



<http://http://en.wikipedia.org/wiki/Algorithm>

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 66.14% | ⌚ 1:34 |
| 高級組  | 難 中 易 | ☑ 54.40% | ⌚ 1:28 |



17 劇院

在海狸市劇院裡的舞臺，是由三盞聚光燈來照明，一盞是紅色，一盞是綠色，另一盞是藍色。舞台的顏色取決於三盞燈的開關組合，如下表所示：

| 紅燈 | 綠燈 | 藍燈 | 舞臺顏色 |
|----|----|----|------|
| 開  | 關  | 關  | 紅色   |
| 關  | 關  | 開  | 綠色   |
| 關  | 關  | 開  | 藍色   |
| 開  | 開  | 關  | 黃色   |
| 開  | 關  | 開  | 洋紅色  |
| 關  | 開  | 開  | 青色   |
| 開  | 開  | 開  | 白色   |
| 關  | 關  | 關  | 黑色   |

從演出開始，三盞燈光會按照以下的規則重複循環運作：

- 紅燈：關閉一分鐘，開啟一分鐘。
- 綠燈：關閉半分鐘，開啟半分鐘。
- 藍燈：開啟兩分鐘，關閉兩分鐘。

請問演出開始一分鐘之後，接下來的半分鐘舞臺是什麼顏色？

- (A) 紅
- (B) 綠
- (C) 白
- (D) 洋紅



正確答案是C

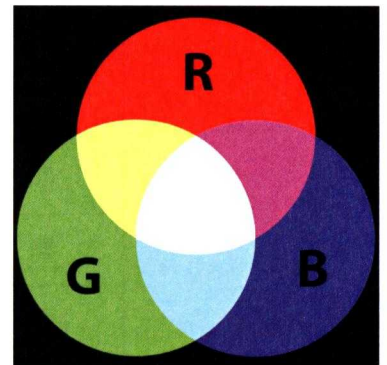
| 時間   | 0:30 | 1:00 | 1:30 | 2:00 | 2:30 | 3:00 | 3:30 | 4:00 |    |
|------|------|------|------|------|------|------|------|------|----|
| 紅燈   | 關    | 關    | 開    | 開    | 關    | 關    | 開    | 開    |    |
| 綠燈   | 關    | 開    | 關    | 開    | 關    | 開    | 關    | 開    |    |
| 藍燈   | 開    | 開    | 開    | 開    | 關    | 關    | 關    | 關    |    |
| 舞台顏色 | 藍色   | 青色   | 洋紅色  | 白色   | 黑色   | 綠色   | 紅色   | 黃色   | 藍色 |



在資訊學科  
上的意義

RGB 模型是利用光的三原色構成顏色的方法。RGB 模型被廣泛運用在很多領域，例如電腦或電視螢幕顯示圖片的方式。圖像是由像素所構成，由每個像素就是利用三原色的組合來表現顏色。這也是電腦硬體和電腦圖學中的重要原理。

本題還包含了理解一系列的動作，也就是題目中開關特定燈光的順序；這是演算法思維很重要的一部份。此外，本題也是平行執行三個獨立序列的一個例子。



關鍵字

RGB 模型、燈光序列、平行執行



相關網頁

[http://en.wikipedia.org/wiki/RGB\\_color\\_model](http://en.wikipedia.org/wiki/RGB_color_model)

[http://en.wikipedia.org/wiki/Parallel\\_computing](http://en.wikipedia.org/wiki/Parallel_computing)

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 55.70% | ⌚ 3:19 |
| 高級組  | 難 中 易 | ☑ 51.21% | ⌚ 3:22 |

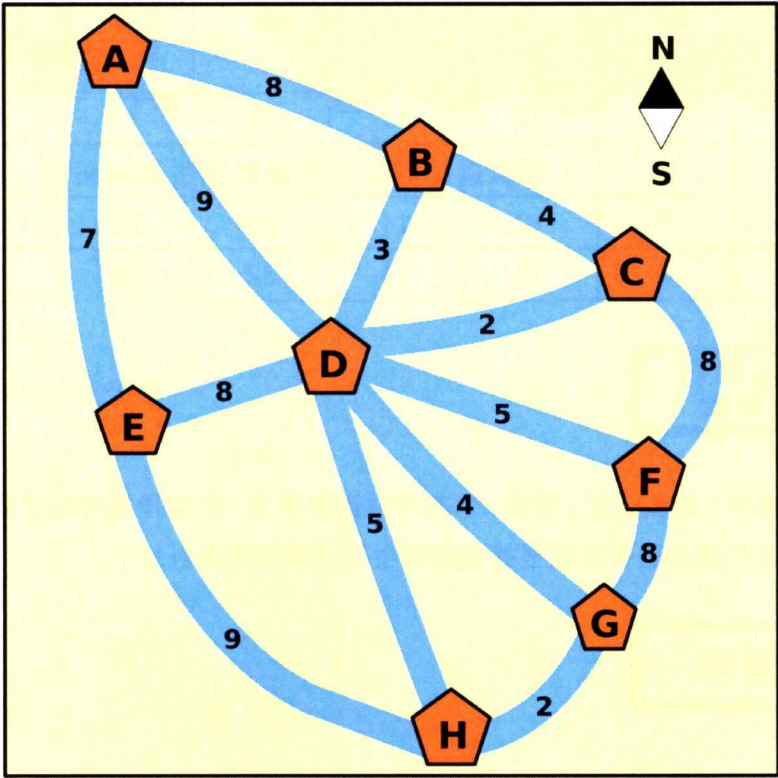


# 18 搜集木頭

建造水壩是海狸們每年最重要的工作。他們從森林裡搜集了許多木頭來當建材。為了搬運這些笨重的木頭，聰明的海狸們發明了一套利用河流系統來運送木頭的規則：

1. 海狸們在河岸邊搜集木頭，並把這些木頭放到木筏上。
2. 在每一段河流中，他們只能將特定數量的新木頭放到木筏上。
3. 木筏只能往南邊流。

海狸們必須依照這些規則，找到能搜集最多木頭的路線，來建造他們的水壩。



上圖說明了每條河流能夠增加的新木頭數量，圖中指針標示出南北方向。如果海狸乘著一艘木筏從 A 流到 H 處，他最多能搜集到多少木頭？

- (A) 32
- (B) 31
- (C) 19
- (D) 40





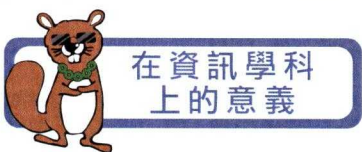
最佳的路徑是  $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow H$ ，而海狸可以搜集到  $8+4+2+8+9=31$  根木頭。

找出最佳路徑的方法是列出一張表格來動態紀錄各節點在海狸由北往南移動時各節點能搜集到最大木頭數目（如下表）。開始計算之前，把每一個節點的紀錄值填上 0。以 A 作為出發節點，並重複下列的步驟：

- 對於從出發節點「往南的每一條路徑」到下一個目的節點
- 目的節點的數值 = 出發節點的數值 + 路徑上的數值
- 如果（新的目的節點數值）>（原來目的節點紀錄的數值）就填上新的數值，並把目的節點設定為出發節點，並停用原來的出發節點

下表表示執行過程的數值變化（紅色斜體字代表被設定為出發節點時的數值）：

|     | A        | B        | C         | D         | E         | F         | G         | H         |
|-----|----------|----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 第一步 | <i>0</i> | 0        | 0         | 0         | 0         | 0         | 0         | 0         |
| 第二步 | 0        | <i>8</i> | 0         | <i>9</i>  | <i>7</i>  | 0         | 0         | 0         |
| 第三步 | 0        | 8        | <i>12</i> | <i>11</i> | <i>17</i> | <i>14</i> | <i>13</i> | <i>14</i> |
| 第四步 | 0        | 8        | 12        | <i>14</i> | <i>19</i> | <i>20</i> | <i>15</i> | <i>16</i> |
| 第五步 | 0        | 8        | 12        | 14        | <i>22</i> | 20        | <i>28</i> | <i>19</i> |
| 第六步 | 0        | 8        | 12        | 14        | 22        | 20        | 18        | <i>31</i> |



本題屬於圖論問題中「最長路徑」問題。它有個很重要應用，也就是為排程問題找到關鍵路徑。從起點開始到終點的最長路徑可以根據有向圖中的拓撲排序來計算。



最長路徑問題、拓撲排序



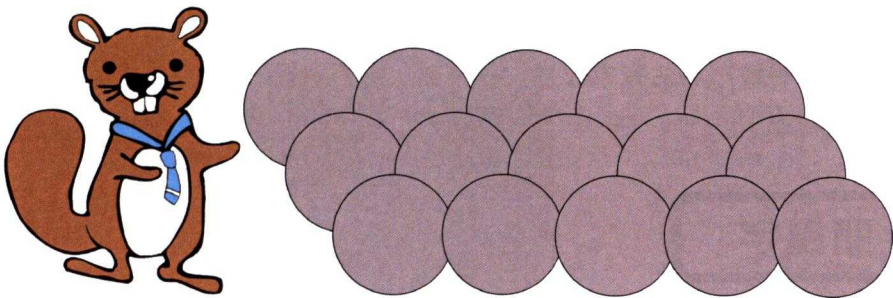
[http://en.wikipedia.org/wiki/Longest\\_path\\_problem](http://en.wikipedia.org/wiki/Longest_path_problem)

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 |          |        |
| 高級組  | 難 中 易 | ☑ 27.06% | ⌚ 1:28 |



19 撿石頭

你正在和一隻海狸玩遊戲。在這個遊戲中有一些石頭，玩家們要輪流撿起這些石頭。在每回合，玩家輪流撿起 1、2 或 3 顆石頭，撿起最後一顆石頭的玩家獲勝。這個遊戲最早的版本是從 9 顆石頭開始玩，而歷史顯示若只有兩個玩家時，只要第一個玩家開局時只撿一顆石頭，接下來不論對手撿幾顆石頭，第一玩家都能獲勝。但現在你和海狸要玩的版本開局時有 15 顆石頭。

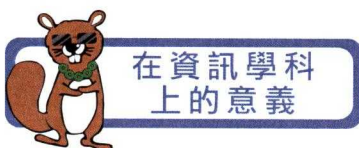


若遊戲從你開始，你應該要撿起幾顆石頭才能保證獲勝？

- (A) 1
- (B) 2
- (C) 3
- (D) 沒有任何方法能保證獲勝



如果遊戲開始時只有 1、2 或 3 顆石頭，輪到的玩家只要撿起所有的石頭就能獲勝。然而，如果遊戲開始時有 4 顆石頭，不管輪到的玩家撿起幾顆石頭，接下來的玩家都能撿起剩下的石頭來取得勝利。因此依照相同的概念，當開始時有 5、6 或 7 顆石頭，你只要留給海狸 4 顆石頭就能獲勝。從上面的推論得知，你可以在每一輪都用四顆石頭當作必勝組合。所以，如果你每次撿起石頭後留給海狸的石頭數目都是 4 的倍數，遊戲的最後你就可以保證並獲勝。



本題和賽局理論有關，而賽局理論是在研究決策戰略。在我們日常生活中，可以發現賽局理論的例子，例如公司爭取市場佔有率或是選舉策略。賽局理論也被廣泛應用在經濟學、生物學、社會科學和電腦科學。



賽局理論、決策、致勝策略



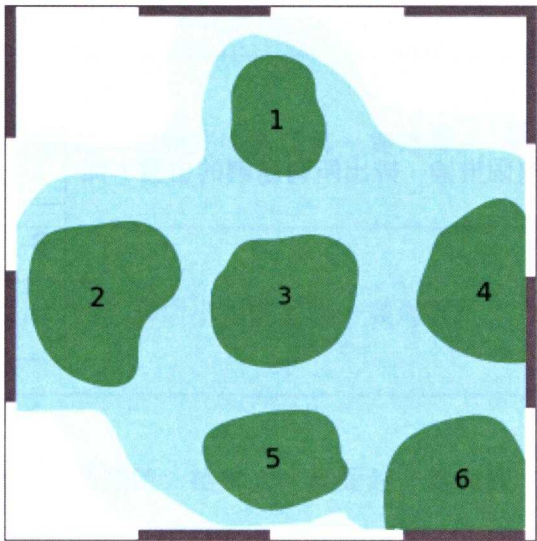
[http://en.wikipedia.org/wiki/Game\\_theory](http://en.wikipedia.org/wiki/Game_theory)

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 70.23% | ⌚ 2:23 |
| 高級組  | 難 中 易 |          |        |



20 羅盤

下方地圖中，標上數字的區塊代表島嶼，而地圖方位如右方的羅盤所標示。



我們只知道這些島嶼的名字分別稱做 A, B, C, D, E 和 F，但不知道每個島嶼確切位置。幸好有以下三個線索：

- C 在 A 的南方、並且在 D 的東南方。
- B 在 F 的西南方、並且在 E 的西北方。
- D 在 F 的南方。

請問區塊 6 的是哪座島嶼？（ 僅填英文字母 ）





由「C 在 A 的南方、並且在 D 的東南方」線索中得出了右表的位置：

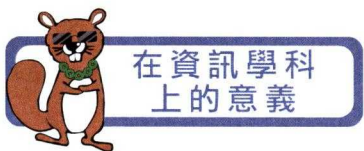
|   |   |
|---|---|
| D | A |
|   | C |

|   |   |
|---|---|
|   | F |
| B |   |
|   | E |

由「B 在 F 的西南方、並且在 E 的西北方」線索中得出了左表位置。

「D 在 F 的南方」使我們能合併上方兩個推論，排出所有島嶼的位置（如下圖），同時得知區塊 6 為島嶼 C。

|   |   |   |
|---|---|---|
|   | F |   |
| B | D | A |
|   | B | C |



有時候，我們需要藉由一步步的邏輯推理和反覆測試來解決問題。而本題需要根據線索一步步地找出島嶼的位置，並且測試過後，再進行下一步驟。



抽象、推理、測試、解答

|      |       |          |        |
|------|-------|----------|--------|
| 中高級組 | 難 中 易 | ☑ 59.30% | ⌚ 2:00 |
| 高級組  | 難 中 易 | ☑ 60.43% | ⌚ 1:57 |

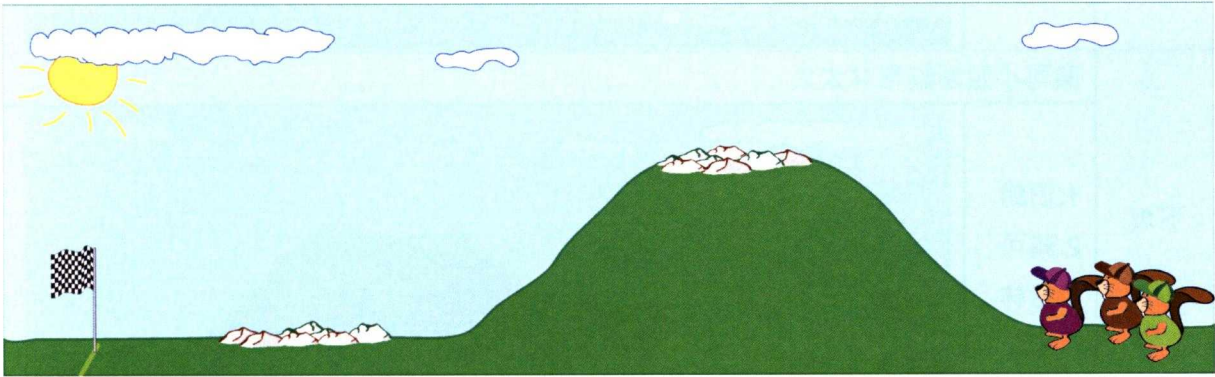


## 21 越野賽

三位勇猛的海狸跑者將在一場越野賽中較勁，在平坦的道路上，三位跑者的速度幾乎一樣，但他們在不同地形的時候三隻海狸各有不同的特殊能力。

|                                                                                   |                     |
|-----------------------------------------------------------------------------------|---------------------|
|   | 在上坡時，伯朗先生能超越一隻海狸。   |
|   | 在下坡時，蘋可小姐能超越一隻海狸。   |
|  | 跑過石頭路時，格林太太能超越一隻海狸。 |

比賽路線的地形如下圖所示：先上坡，接著石頭路，然後下坡，再接著石頭路。目前的狀態是蘋可小姐跑在最前面，接著是伯朗先生和格林太太。



他們到達終點的順序為何？

- (A) 蘋可小姐、伯朗先生、格林太太
- (B) 伯朗先生、格林太太、蘋可小姐
- (C) 格林太太、蘋可小姐、伯朗先生
- (D) 伯朗先生、蘋可小姐、格林太太



正確答案是B

| 路況         | 順序                      | 圖示 |
|------------|-------------------------|----|
| 開始         | 1. 蘋可<br>2. 伯朗<br>3. 格林 |    |
| 伯朗先生超越蘋可小姐 |                         |    |
| 上坡         | 1. 伯朗<br>2. 蘋可<br>3. 格林 |    |
| 格林太太超越蘋可小姐 |                         |    |
| 石頭路        | 1. 伯朗<br>2. 格林<br>3. 蘋可 |    |
| 蘋可小姐超越格林太太 |                         |    |
| 下坡         | 1. 伯朗<br>2. 蘋可<br>3. 格林 |    |
| 格林太太超越蘋可小姐 |                         |    |
| 石頭路        | 1. 伯朗<br>2. 格林<br>3. 蘋可 |    |



### 在資訊學科 上的意義

工程師需要經常仔細觀察他們寫的程式會如何執行，尤其是當程式出現問題時。當程式無法正確執行時，工程師必須仔細的檢視並確認每一行程式執行的結果。

本題跟上述情況相似：開始的時候你手中握有一些資訊，也就是題目中跑者的能力和順序。你必須「逐步執行程式」，這些「步驟」就是題目中的海狸會遇到的各種路況如：上坡路、石頭路、下坡路、石頭路；你必須依序觀察執行每一步的結果，才能得到程式的「輸出」，也就是題目中最後的排名結果。



### 關鍵字

程式設計、偵錯









2015年國際運算思維能力測驗工作小組

指導單位 教育部

執行單位 國立臺灣師範大學 資訊工程學系

工作團隊 李忠謀、柯佳伶、林育慈、蔣宗哲、

林于立、朱德清、汪藝誼

