中華民國第四十八屆中小學科學展覽會
作品說明書

高中組　數學科

佳作

040410

乾坤大挪移---數獨 VS 幻方

學校名稱：國立臺中第一高級中學

作者：	指導老師：
高二 林品慶 高二 黃治綱 高二 李名弘	柯建彰 林奇鋒

關鍵詞：　數獨、幻方(魔方陣)

乾坤大挪移 — 數獨 VS 幻方

壹、摘要

三年前的科學人雜誌曾看到一篇討論數獨初盤的文章—討論使一數獨有唯一解所需填進去的最小數目，結果是 17 個數字。因爲數獨是最近才竄起的數字遊戲，所以能找的資料也很有限。後來便想到跟幻方(魔方陣)結合，試著找出對應關係。

先鎖定較簡單的4×4數獨，爲了讓數獨具備特殊性質(對角線上)，我們特別創造了”G 數獨”—一組對角線上數字均不重複的數獨，使之在對角線上的和仍然與行列相同。利用幻方每行每列對 4 做模後爲完全剩餘系之性質我們順利的找出一種變換方法，即在4×4 G 數獨固定一3×3方陣的四角，將固定角之相鄰元素互換可得一共軛數獨與原數獨以$1:4$及$4:1$比例構成 X 幻方。之後找到「井字變換法」亦可構成 X 幻方。

原本想利用4×4 G 數獨的性質將9×9和16×16的情況解決，但用程式跑了一天還無法得到滿意的數據，因此我們開始懷疑9×9 G 數獨的存在性。於是我們利用 Excel 的自訂函式功能來找尋可能的變換法。最後，成功地找出一個廣義的變換法，其中仍是以兩共軛 G 數獨以$1:n^2$及$n^2:1$的比例製作 X 幻方。

貳、研究動機

近年來，每天報紙的副刊上，總是會出現幾題數獨供讀者腦力激盪。只是把9×9方陣中挖掉若干個空格，利用簡單的邏輯推理即構成一個趣味數字遊戲。如王建民在上場投球前都會先玩數獨，平常在捷運上、公車上、甚至到處都有機會看到數獨出現，這就是數獨迷人的威力。而幻方(魔方陣)也是另外一種填$1\sim n^2$的數字遊戲，雖然對數獨及對幻方的文獻已經很可觀，但將二者連結的研究卻寥寥可數，因此不禁想到數獨幻方之間是否有某種對應關係。

參、研究目的

找出數獨與幻方之間的關聯性。

一、是否能由4×4數獨經有限步驟推導出一組幻方。

二、是否能由4×4幻方經有限步驟推導出一組數獨。

三、是否能由9×9數獨經有限步驟推導出一組幻方。

四、是否能由9×9幻方經有限步驟推導出一組數獨。

五、是否能推廣到$n^2\times n^2$的情況。

六、尋找”數獨-幻方”的實際用途，與生活結合。

肆、研究方法與器材

Dev-C++、Lingo、紙、筆、Microsoft Office Excel、Visual Basic

伍、研究過程

先定義數獨方陣：

數獨方陣為一$n^2 \times n^2 (n \in N)$的正方形方陣，其中每一的正方形方陣又稱為一個區塊。同時滿足每一列、每一行、每一宮$(n \times n)$中，$1 \sim n$各只能出現 1 次，即為一數獨方陣，如下圖所示即為一數獨方陣：(以$n = 3$為例)

1	9	4	3	5	6	7	2	8
2	5	7	1	4	8	3	6	9
6	3	8	2	7	9	1	5	4
3	1	2	4	6	5	8	9	7
7	8	5	9	1	3	2	4	6
4	6	9	7	8	2	5	3	1
8	2	1	5	9	4	6	7	3
9	7	3	6	2	1	4	8	5
5	4	6	8	3	7	9	1	2

選定任一行列及九宮格，數字皆由不重複的$1 \sim 9$組成，符合數獨定義。

再定義幻方方陣(亦稱魔方陣)：

幻方為一$n \times n (n \in N)$的正方形方陣。滿足填入連續自然數$1 \sim n^2$使得每一列、每一行、每一對角線和皆相同，即為一幻方方陣，如下圖所示即為一幻方方陣：(以$n = 3$為例)

8	1	6
3	5	7
4	9	2

其中每一列、每一行、每一對角線和皆為 15。

一、4×4**數獨-幻方**探討：

因為9×9的方陣過於龐大且複雜，因此先討論較簡單的4×4的方陣。4×4數獨方陣中兩對角線元素各包含 1 至 4 且不重複，稱此種數獨為**”G 數獨**”。經由多次嘗試，我們發現”**G 數獨**”有一些特殊性質。例如：

1	4	2	3
3	2	4	1
4	1	3	2
2	3	1	4

此數獨方陣即符合”G 數獨”的定義

可以以觀察法直接產生一 G 數獨，也可以用 Lingo 此一套裝軟體產生之，前者需要強大的連續推理能力，而後者只需要輸入參數即可。由於 Lingo 是線性規劃的程式，因此給予越多限制式，運算效率越高。賦予 Lingo 我們所需要的限制式，便可以產生 G 數獨。給予任 4 個格子內容，即可得到一 G 數獨。

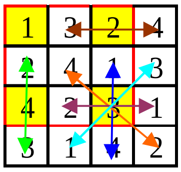
1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

G 數獨

利用此 G 數獨，固定其中一3×3的方陣，將其四角固定，我們稱此四格爲四個”G 格”，如下圖之黃格部分。

1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

以此四個 G 格對附近方格內的數以 G 格爲中心做對調，如下圖：



其中原本為 G 格”2”旁邊的 3,4 做對調、G 格”4”旁邊的 5,2 做對調、G 格”3”附近的 3,1 對調、2,1 對調、2,4 對調。

以如此程序完成所有對調，可以得到下圖：

1	4	2	3
3	2	4	1
4	1	3	2
2	3	1	4

可以發現得到另外一個 G 數獨，我們稱原 G 數獨與得到的 G 數獨爲”**共軛數獨**”。

對兩個共軛數獨加以討論，暫且把前者和後者分別命名爲M, N，第i列第j行的數分別記爲M_{ij}及N_{ij}。

1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

G 數獨 M

1	4	2	3
3	2	4	1
4	1	3	2
2	3	1	4

G 數獨 N

如果將$M_{ij}+4\times N_{ij}$(遇到M_{ij}=4 時，M_{ij}視爲 0)各自填入另外一個4×4的空方陣中，可以得到一個幻方方陣，如下圖所示：

5	3	10	16
14	12	1	7
4	6	15	9
11	13	8	2

↑這是幻方

相同的，如果將 $N_{ij}+4\times M_{ij}$(遇到 N_{ij}=4 時，N_{ij} 視爲 0)做相同程序也可以的得到一個幻方方陣：

5	16	10	3
11	2	8	13
4	9	15	6
14	7	1	12

↑這也是幻方

相同的，固定一 G 數獨其他 3×3 方陣的四個 G 格後，做對調的動作，也會有如上述相同的性質存在。

1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

可以將問題轉換爲完全剩餘系的想法(設 $n\geq1$，$a_1,a_2,...,a_n$ 關於模 n 兩兩不同餘，則稱 $a_1,a_2,...,a_n$ 爲模 n 的一個完全剩餘系)，以下證明之：

$\forall i,j\in\{1,2,3,4\}$

$a_{ij}\in\{1,2,3,4\}$

設 M 爲一 G 數獨

其中 a_{ij} 表 G 是數獨 M 第 i 列第 j 行的值

即每一行、列、對角線、宮之元素集合均爲對 4 的完全剩餘系

固定 $a_{11},a_{31},a_{13},a_{33}$ 將 M 做前述之對換得到 N

試證：N 亦屬於 G 數獨，及 4M+N 爲一幻方

$$\begin{bmatrix} a_{11},a_{12},a_{13},a_{14} \\ a_{21},a_{22},a_{23},a_{24} \\ a_{31},a_{32},a_{33},a_{34} \\ a_{41},a_{42},a_{43},a_{44} \end{bmatrix} \qquad \begin{bmatrix} a_{11},a_{14},a_{13},a_{12} \\ a_{41},a_{44},a_{43},a_{42} \\ a_{31},a_{34},a_{33},a_{32} \\ a_{21},a_{24},a_{23},a_{22} \end{bmatrix}$$

M　　　　　　　N

pf (1) 本部份將證明$\{a_{12},a_{43},a_{34},a_{21}\}$爲對 4 的完全剩餘系

利用反證法，設$a_{12}=a_{21}$

則 $a_{14}=a_{31},a_{13}=a_{41}$或$a_{41}=a_{14},a_{31}=a_{13}$

(a)$a_{14}=a_{31},a_{13}=a_{41}$時

代入原 M 數獨可發現左下-右上之對角線不符條件

(b)$a_{41}=a_{14},a_{31}=a_{13}$時

亦不符條件

原設不合

$\therefore a_{12}\neq a_{21}$

同理，$a_{12}\neq a_{43}$ $a_{12}\neq a_{34}$ $a_{34}\neq a_{21}$ $a_{43}\neq a_{21}$ $a_{43}\neq a_{34}$

即$\{a_{12},a_{43},a_{34},a_{21}\}$中的數兩兩相異

即爲對 4 之完全剩餘系

$\therefore$得證

(2)本部份將證明每行每列均爲對 4 之完全剩餘系：

由已知：

$\because$M 爲一 G 數獨

$\therefore$每一行、每一列、每一對角線、每一宮格均爲對 4 的完全剩餘系，與 N 每行每列需爲對 4 之完全剩餘系相同

$\therefore$得証。

(3)本部份將證明宮格部份 ($\{a_{21},a_{31},a_{24},a_{34}\}\{a_{12},a_{13},a_{42},a_{43}\}$)均亦爲對 4 的完全剩餘系($\{a_{11},a_{41},a_{14},a_{44}\}\{a_{22},a_{23},a_{32},a_{33}\}$本爲對 4 之完全剩餘系)：

先証$\{a_{21},a_{31},a_{24},a_{34}\}$爲對 4 之完全剩餘系：

由(1)可知$a_{21}\neq a_{34}$，又$a_{21}\neq a_{31}$，$a_{21}\neq a_{24}$，$a_{31}\neq a_{34}$，$a_{24}\neq a_{34}$

$\therefore$只需證 $a_{31}\neq a_{24}$：

利用反證法，設$a_{31}=a_{24}$=p，則$a_{11}=a_{23}=q,a_{41}=a_{22}$=r(由(1)(2)可得)($p,q,r$ 兩兩相異)

代回 M 數獨，則由 G 數獨之性質$a_{12}=a_{43}=p$，如下圖：

$$\begin{bmatrix} q, & p, & a_{13}, & a_{14} \\ a_{21}, & r, & q, & p \\ p, & a_{32}, & a_{33}, & a_{34} \\ r, & a_{42}, & p, & a_{44} \end{bmatrix}$$

a_{33},a_{44}必有一數需爲 P 又須滿足$a_{33}\neq p$，$a_{44}\neq p$(宮之性質)，

$\therefore$矛盾，原設不合

$\therefore a_{31}\neq a_{24}$，$\{a_{21},a_{31},a_{24},a_{34}\}$爲對 4 之完全剩餘系

同理，$\{a_{12},a_{13},a_{42},a_{43}\}$亦爲對 4 之完全剩餘系

$\therefore$得證

(4)則 4M+N 會等於：

$$\begin{bmatrix} a_{11},a_{12},a_{13},a_{14} \\ a_{21},a_{22},a_{23},a_{24} \\ a_{31},a_{32},a_{33},a_{34} \\ a_{41},a_{42},a_{43},a_{44} \end{bmatrix} \qquad \begin{bmatrix} a_{11},a_{14},a_{13},a_{12} \\ a_{41},a_{44},a_{43},a_{42} \\ a_{31},a_{34},a_{33},a_{32} \\ a_{21},a_{24},a_{23},a_{22} \end{bmatrix}$$

M N

$$\begin{bmatrix} 4\times a_{11}+a_{11},4\times a_{12}+a_{14},4\times a_{13}+a_{13},4\times a_{14}+a_{12} \\ 4\times a_{21}+a_{41},4\times a_{22}+a_{44},4\times a_{23}+a_{43},4\times a_{24}+a_{42} \\ 4\times a_{31}+a_{31},4\times a_{32}+a_{34},4\times a_{33}+a_{33},4\times a_{34}+a_{32} \\ 4\times a_{41}+a_{21},4\times a_{42}+a_{24},4\times a_{43}+a_{23},4\times a_{44}+a_{22} \end{bmatrix}$$

$$4\times M+N$$

由(1)(2)(3)

每行每列分別加起來均等於 4(1+2+3+4)+(1+2+3+4)=50

滿足幻方之條件

$\therefore$ $4\times M+N$ 爲一幻方

$\therefore$ 得證

另外，如果對一 G 數獨行列變換也會有類似的性質：

以先取一過中心鉛垂線作對稱軸，取第 k 行與其對稱行第 $(n+1-k)$ 行對調，接著再取另外一中心水平軸，也對第 k 列及其對稱列第 $(n+1-k)$ 列對調，也可以得到一 G 數獨。

1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

↑這是一個 G 數獨

藍線爲鉛垂對稱軸，紅線爲水平對稱軸

1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

先以 1,4 行對稱列關於鉛垂對稱軸對調可得到：

4	3	2	1
3	4	1	2
1	2	3	4
2	1	4	3

再取 1,4 列對稱行關於水平對稱軸對調可得到：

2	1	4	3
3	4	1	2
1	2	3	4
4	3	2	1

↑這還是一個 G 數獨

也可以稱原來的 G 數獨及後來得到的 G 數獨為共軛數獨，相同的，以 $4M+N$ 及 $4N+M$ 的形式一樣可以得到一種幻方，其證明與前一種變換方法相同。

既然可以從數獨推導出幻方，那麼由幻方推出數獨在某種條件之下應該也是會成立的。因為可以由 G 數獨此種特殊數獨導出幻方，所以嘗試其逆定理。以下從幻方推數獨：

若一 4×4 幻方中符合每個宮集合中每一元素對 4 做模後的集合為對 4 之完全剩餘系，則稱此幻方為 X 幻方。

嘗試將一 X 幻方衍生出一組共軛 G 數獨，即將幻方中每個數字換為 $4M+N$ 之形式，並加以記錄。

例如：

5	3	10	16
14	12	1	7
4	6	15	9
11	13	8	2

↑先得到一幻方

$4\times1+1$	$4\times0+3$	$4\times2+2$	$4\times3+4$
$4\times3+2$	$4\times2+4$	$4\times0+1$	$4\times1+3$
$4\times0+4$	$4\times1+2$	$4\times3+3$	$4\times2+1$
$4\times2+3$	$4\times3+1$	$4\times1+4$	$4\times0+2$

↑轉換 4M+N 的形式

1	0	2	3
3	2	0	1
0	1	3	2
2	3	1	0

M’

1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

N

將 0 修正為 4

1	4	2	3
3	2	4	1
4	1	3	2
2	3	1	4

↑G 數獨 M

1	3	2	4
2	4	1	3
4	2	3	1
3	1	4	2

↑G 數獨 N

因此能知道數獨推導幻方的逆定理為真。

利用 C++，發現4×4的 G 數獨共有 48 組，但不是任兩 G 數獨皆可以組成 X 幻方。於是我們大膽猜想或許每一組 G 數獨均能和其他N組 G 數獨產生 X 幻方。經由檢驗後(詳見附錄檢驗程式)，得到每一 G 數獨都能與其他 24 組 G 數獨構成 X 幻方，所以共有$\frac{48\times24}{2}=576$組4×4 X 幻方。

二、9×9**數獨-幻方**探討：

(一)條件式：

因爲在9×9數獨-幻方中，本想藉由結果來找尋變換法，但我們所寫的程式無法跑出所需要的結果，所以我們便反過來想，先定義一種變換法以推得結果，最後加以檢驗證明。我們找到了一些規律性。因爲9×9的數獨有九個宮格，便想到可以先固定一個宮格裡的九個值再推出整個數獨，九個值的填法如下：

A1	C2	
C1	A2	B2
	B1	A3

其中需滿足　A1+A2+A3

=A1+B1+B2

=C1+C2+A3

=C1+A2+B2

=C2+A2+B1

且 9 格內的值需互異，否則不滿足數獨定義。

可以使用各種工具(觀察法,C++,…)得到符合條件的單位宮格，此單位宮格 H 即滿足填法：

3	6	1
2	5	8
9	4	7

(以下皆以此單位宮格 H 作範例)

將單位宮格 H 內的順序加以調整，第一行平移至最後一行，稱爲”左平移”；類似的，第一列平移至最後一列，稱爲”上平移”。以滿足此填法的單位宮格 H 填入9×9方陣的左上宮格，每往下一九宮格就執行一次”左平移”後填入，每往右一宮格就執行一次”上平移”後填入，可得到一9×9數獨方陣，實例如下:

3	6	1	2	5	8	9	4	7
2	5	8	9	4	7	3	6	1
9	4	7	3	6	1	2	5	8
6	1	3	5	8	2	4	7	9
5	8	2	4	7	9	6	1	3
4	7	9	6	1	3	5	8	2
1	3	6	8	2	5	7	9	4
8	2	5	7	9	4	1	3	6
7	9	4	1	3	6	8	2	5

稱此爲”初始 G 數獨”

探討如何將此初始 G 數獨轉換爲幻方方陣：

首先將初始數獨內的數字全部減 1，得到一由 0~8 組成的 9×9 數獨 M，實例如下：

2	5	0	1	4	7	8	3	6
1	4	7	8	3	6	2	5	0
8	3	6	2	5	0	1	4	7
5	0	2	4	7	1	3	6	8
4	7	1	3	6	8	5	0	2
3	6	8	5	0	2	4	7	1
0	2	5	7	1	4	6	8	3
7	1	4	6	8	3	0	2	5
6	8	3	0	2	5	7	1	4

G 數獨 M

然後將此數獨以有限步驟變換如下：

1.取出左上角的單位宮格

2	5	0
1	4	7
8	3	6

2.將此單位宮格逆時針旋轉 90 度。

0	7	6
5	4	3
2	1	8

3.將此單位宮格執行一次右平移。

6	0	7
3	5	4
8	2	1

單位宮格 S

4.由上述步驟即可得到單位宮格 S。

利用此單位宮格 S 再次進行填製初始 G 數獨的方法作成9×9 G 數獨，單位宮格內的順序加以調整，每往下一九宮格就執行一次”右平移”和”下平移”後填入，每往右一宮格就執行一次”上平移”和”右平移”後填入含 0~8，可得到一 G 數獨 N，將實例完整填入後如下：

6	0	7	8	2	1	3	5	4
3	5	4	6	0	7	8	2	1
8	2	1	3	5	4	6	0	7
7	6	0	1	8	2	4	3	5
4	3	5	7	6	0	1	8	2
1	8	2	4	3	5	7	6	0
0	7	6	2	1	8	5	4	3
5	4	3	0	7	6	2	1	8
2	1	8	5	4	3	0	7	6

G 數獨 N

將 G 數獨 M 與 G 數獨 N 內的值以 9:1 相加之後得到一9×9 方陣如下：

2	5	0	1	4	7	8	3	6
1	4	7	8	3	6	2	5	0
8	3	6	2	5	0	1	4	7
5	0	2	4	7	1	3	6	8
4	7	1	3	6	8	5	0	2
3	6	8	5	0	2	4	7	1
0	2	5	7	1	4	6	8	3
7	1	4	6	8	3	0	2	5
6	8	3	0	2	5	7	1	4

G 數獨 M

6	0	7	8	2	1	3	5	4
3	5	4	6	0	7	8	2	1
8	2	1	3	5	4	6	0	7
7	6	0	1	8	2	4	3	5
4	3	5	7	6	0	1	8	2
1	8	2	4	3	5	7	6	0
0	7	6	2	1	8	5	4	3
5	4	3	0	7	6	2	1	8
2	1	8	5	4	3	0	7	6

G 數獨 N

24	45	7	17	38	64	75	32	58
12	41	67	78	27	61	26	47	1
80	29	55	21	50	4	15	36	70
52	6	18	37	71	11	31	57	77
40	66	14	34	60	72	46	8	20
28	62	74	49	3	23	43	69	9
0	25	51	65	10	44	59	76	30
68	13	39	54	79	33	2	19	53
56	73	35	5	22	48	63	16	42

爲符合幻方方陣定義，再將此方陣內的每個值各加上 1，即可得到：

$9 \times M + 1 \times N$ ：

25	46	8	18	39	65	76	33	59
13	42	68	79	28	62	27	48	2
81	30	56	22	51	5	16	37	71
53	7	19	38	72	12	32	58	78
41	67	15	35	61	73	47	9	21
29	63	75	50	4	24	44	70	10
1	26	52	66	11	45	60	77	31
69	14	40	55	80	34	3	20	54
57	74	36	6	23	49	64	17	43

X 幻方

(其中每行每列對角線相加後值皆爲 369)

可以發現以不同方法仍可得到 X 幻方。

相同的，若 G 數獨 M 與 G 數獨 N 內每一格以 1:9 的比例相加，也可以得到另外一組 X 幻方如下：

$1 \times M + 9 \times N$ ：

57	6	64	74	23	17	36	49	43
29	50	44	63	4	70	75	24	10
81	22	16	30	51	37	56	5	71
69	55	3	14	80	20	40	34	54
41	35	47	67	61	9	15	73	21
13	79	27	42	28	48	68	62	2
1	66	60	26	11	77	52	45	31
53	38	32	7	72	58	19	12	78
25	18	76	46	39	33	8	65	59

亦爲 X 幻方

因此我們稱 G 數獨 M 與 G 數獨 N 有共軛關係。

(二)推廣：

經過試誤法，我們再度找到了一些規律性。因爲9×9的數獨有九個宮格，便想到可以先固定一個宮格裡的九個值再推出整個數獨，作法如下：

1	2	3
5	4	6
9	8	7

(以下皆以此單位宮格 H 作範例)

將此單位宮格 H 填入9×9方陣的左上宮格，每往下一九宮格就執行一次”左平移”和”上平移”後填入，每往右一宮格就執行一次”左平移”和”下平移”後填入，可得到一9×9數獨方陣，實例如下:

1	2	3	8	7	9	6	5	4
5	4	6	2	3	1	7	9	8
9	8	7	4	6	5	3	1	2
4	6	5	3	1	2	9	8	7
8	7	9	6	5	4	1	2	3
2	3	1	7	9	8	5	4	6
7	9	8	5	4	6	2	3	1
3	1	2	9	8	7	4	6	5
6	5	4	1	2	3	8	7	9

稱此爲”初始 G 數獨”

探討如何將此初始 G 數獨轉換爲幻方方陣：

首先將初始數獨內的數字全部減 1，得到一由 0~8 組成的9×9數獨 M，實例如下：

0	1	2	7	6	8	5	4	3
4	3	5	1	2	0	6	8	7
8	7	6	3	5	4	2	0	1
3	5	4	2	0	1	8	7	6
7	6	8	5	4	3	0	1	2
1	2	0	6	8	7	4	3	5
6	8	7	4	3	5	1	2	0
2	0	1	8	7	6	3	5	4
5	4	3	0	1	2	7	6	8

G 數獨 M

然後將此數獨以有限步驟變換如下：

1.取出左上角的單位宮格

0	1	2
4	3	5
8	7	6

2.將此單位宮格逆時針旋轉 90 度。

2	5	6
1	3	7
0	4	8

3.將此單位宮格執行一次右平移。

6	2	5
7	1	3
8	0	4

單位宮格 S

4.由上述步驟即可得到單位宮格 S。

利用此單位宮格 S 再次進行填製初始 G 數獨的方法作成 9×9 G 數獨，單位宮格內的順序加以調整，每往下一九宮格就執行一次”右平移”和”下平移”後填入，每往右一宮格就執行一次”上平移”和”右平移”後填入含 0~8，可得到一 G 數獨 N，將實例完整填入後如下：

6	2	5	3	7	1	0	4	8
7	1	3	4	8	0	2	5	6
8	0	4	5	6	2	1	3	7
4	8	0	2	5	6	7	1	3
5	6	2	1	3	7	8	0	4
3	7	1	0	4	8	6	2	5
1	3	7	8	0	4	5	6	2
0	4	8	6	2	5	3	7	1
2	5	6	7	1	3	4	8	0

G 數獨 N

將 G 數獨 M 與 G 數獨 N 內的值以 1:9 相加之後得到一 9×9 方陣如下：

0	1	2	7	6	8	5	4	3
4	3	5	1	2	0	6	8	7
8	7	6	3	5	4	2	0	1
3	5	4	2	0	1	8	7	6
7	6	8	5	4	3	0	1	2
1	2	0	6	8	7	4	3	5
6	8	7	4	3	5	1	2	0
2	0	1	8	7	6	3	5	4
5	4	3	0	1	2	7	6	8

G 數獨 M

6	2	5	3	7	1	0	4	8
7	1	3	4	8	0	2	5	6
8	0	4	5	6	2	1	3	7
4	8	0	2	5	6	7	1	3
5	6	2	1	3	7	8	0	4
3	7	1	0	4	8	6	2	5
1	3	7	8	0	4	5	6	2
0	4	8	6	2	5	3	7	1
2	5	6	7	1	3	4	8	0

G 數獨 N

54	19	47	34	69	17	5	40	75
67	12	32	37	74	0	24	53	61
80	7	42	48	59	22	11	27	64
39	77	4	20	45	55	71	16	33
52	60	26	14	31	66	72	1	38
28	65	9	6	44	79	58	21	50
15	35	70	76	3	41	46	56	18
2	36	73	62	25	51	30	68	13
23	49	57	63	10	29	43	78	8

爲符合幻方方陣定義，再將此方陣內的每個值各加上 1，即可得到：

$1 \times M + 9 \times N$ ：

55	20	48	35	70	18	6	41	76
68	13	33	38	75	1	25	54	62
81	8	43	49	60	23	12	28	65
40	78	5	21	46	56	72	17	34
53	61	27	15	32	67	73	2	39
29	66	10	7	45	80	59	22	51
16	36	71	77	4	42	47	57	19
3	37	74	63	26	52	31	69	14
24	50	58	64	11	30	44	79	9

X 幻方

(其中每行每列對角線相加後值皆爲 369)

相同的，若 G 數獨 M 與 G 數獨 N 內每一格以 9:1 的比例相加，也可以得到另外一組 X 幻方如下：

$9\times M+1\times N$ ：

7	12	24	67	62	74	46	41	36
44	29	49	14	27	1	57	78	70
81	64	59	33	52	39	20	4	17
32	54	37	21	6	16	80	65	58
69	61	75	47	40	35	9	10	23
13	26	2	55	77	72	43	30	51
56	76	71	45	28	50	15	25	3
19	5	18	79	66	60	31	53	38
48	42	34	8	11	22	68	63	73

亦爲 X 幻方

以下證明之：

A 矩陣爲一3×3的單位宮格

$$A=\begin{bmatrix} A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \end{bmatrix}\text{，其中 } A_{i,j}=0\sim 8,\quad i,j=1,2,3$$

定義：

$$I=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\text{(單位矩陣)}$$

$$I^{M}=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}=\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}\text{，}I^{R^1}=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}=\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$I^{R^2}=\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}=\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\text{，}I^{L^1}=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}=\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$I^{L^2}=\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}=\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}\text{，}I^{U^1}=\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}=\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$I^{U^2}=\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}=\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}，I^{D^1}=\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}=\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}$$

$$I^{D^2}=\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}=\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}$$

$$A^T=\begin{bmatrix}A_{1,1}&A_{1,2}&A_{1,3}\\A_{2,1}&A_{2,2}&A_{2,3}\\A_{3,1}&A_{3,2}&A_{3,3}\end{bmatrix}，A^M=\begin{bmatrix}A_{1,1}&A_{2,1}&A_{3,1}\\A_{1,2}&A_{2,2}&A_{3,2}\\A_{1,3}&A_{2,3}&A_{3,3}\end{bmatrix}\begin{bmatrix}0&0&1\\0&1&0\\1&0&0\end{bmatrix}=\begin{bmatrix}A_{3,1}&A_{2,1}&A_{1,1}\\A_{3,2}&A_{2,2}&A_{1,2}\\A_{3,3}&A_{2,3}&A_{1,3}\end{bmatrix}$$

$$A^{L^1}=\begin{bmatrix}A_{1,1}&A_{2,1}&A_{3,1}\\A_{1,2}&A_{2,2}&A_{3,2}\\A_{1,3}&A_{2,3}&A_{3,3}\end{bmatrix}\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}=\begin{bmatrix}A_{2,1}&A_{3,1}&A_{1,1}\\A_{2,2}&A_{3,2}&A_{1,2}\\A_{2,3}&A_{3,3}&A_{1,3}\end{bmatrix}$$

$$A^{L^2}=\begin{bmatrix}A_{1,1}&A_{2,1}&A_{3,1}\\A_{1,2}&A_{2,2}&A_{3,2}\\A_{1,3}&A_{2,3}&A_{3,3}\end{bmatrix}\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}$$

$$=\begin{bmatrix}A_{1,1}&A_{2,1}&A_{3,1}\\A_{1,2}&A_{2,2}&A_{3,2}\\A_{1,3}&A_{2,3}&A_{3,3}\end{bmatrix}\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}=\begin{bmatrix}A_{3,1}&A_{1,1}&A_{2,1}\\A_{3,2}&A_{1,2}&A_{2,2}\\A_{3,3}&A_{1,3}&A_{2,3}\end{bmatrix}$$

$$A^{L^3}=A$$

$$A^{R^1}=\begin{bmatrix}A_{1,1}&A_{2,1}&A_{3,1}\\A_{1,2}&A_{2,2}&A_{3,2}\\A_{1,3}&A_{2,3}&A_{3,3}\end{bmatrix}\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}=\begin{bmatrix}A_{3,1}&A_{1,1}&A_{2,1}\\A_{3,2}&A_{1,2}&A_{2,2}\\A_{3,3}&A_{1,3}&A_{2,3}\end{bmatrix}$$

$$A^{R^2}=\begin{bmatrix}A_{1,1}&A_{2,1}&A_{3,1}\\A_{1,2}&A_{2,2}&A_{3,2}\\A_{1,3}&A_{2,3}&A_{3,3}\end{bmatrix}\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}$$

$$=\begin{bmatrix}A_{1,1}&A_{2,1}&A_{3,1}\\A_{1,2}&A_{2,2}&A_{3,2}\\A_{1,3}&A_{2,3}&A_{3,3}\end{bmatrix}\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}=\begin{bmatrix}A_{2,1}&A_{3,1}&A_{1,1}\\A_{2,2}&A_{3,2}&A_{1,2}\\A_{2,3}&A_{3,3}&A_{1,3}\end{bmatrix}$$

$$A^{R^3}=A$$

$$A^{U^1} = \begin{bmatrix} A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \\ A_{1,1} & A_{2,1} & A_{3,1} \end{bmatrix}$$

$$A^{U^2} = \begin{bmatrix} A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \\ A_{1,1} & A_{2,1} & A_{3,1} \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} A_{1,3} & A_{2,3} & A_{3,3} \\ A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \end{bmatrix}$$

$$A^{U^3} = A$$

$$A^{U^1 L^1} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \\ A_{1,1} & A_{2,1} & A_{3,1} \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} A_{2,2} & A_{3,2} & A_{1,2} \\ A_{2,3} & A_{3,3} & A_{1,3} \\ A_{2,1} & A_{3,1} & A_{1,1} \end{bmatrix}$$

$$A^{U^2 L^2} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} A_{2,2} & A_{3,2} & A_{1,2} \\ A_{2,3} & A_{3,3} & A_{1,3} \\ A_{2,1} & A_{3,1} & A_{1,1} \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} A_{3,3} & A_{1,3} & A_{2,3} \\ A_{3,1} & A_{1,1} & A_{2,1} \\ A_{3,2} & A_{1,2} & A_{2,2} \end{bmatrix}$$

$$A^{D^1} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \end{bmatrix} = \begin{bmatrix} A_{1,3} & A_{2,3} & A_{3,3} \\ A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \end{bmatrix}$$

$$A^{D^2} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} A_{1,3} & A_{2,3} & A_{3,3} \\ A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \end{bmatrix} \begin{bmatrix} A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \\ A_{1,1} & A_{2,1} & A_{3,1} \end{bmatrix}$$

$$A^{D^3} = A$$

$$A^{D^1L^1}=\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}\begin{bmatrix}A_{1,1}&A_{2,1}&A_{3,1}\\A_{1,2}&A_{2,2}&A_{3,2}\\A_{1,3}&A_{2,3}&A_{3,3}\end{bmatrix}\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}$$

$$=\begin{bmatrix}A_{1,3}&A_{2,3}&A_{3,3}\\A_{1,1}&A_{2,1}&A_{3,1}\\A_{1,2}&A_{2,2}&A_{3,2}\end{bmatrix}\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}=\begin{bmatrix}A_{2,3}&A_{3,3}&A_{1,3}\\A_{2,1}&A_{3,1}&A_{1,1}\\A_{2,2}&A_{3,2}&A_{1,2}\end{bmatrix}$$

$$A^{D^2L^2}=\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}\begin{bmatrix}A_{2,3}&A_{3,3}&A_{1,3}\\A_{2,1}&A_{3,1}&A_{1,1}\\A_{2,2}&A_{3,2}&A_{1,2}\end{bmatrix}\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}=\begin{bmatrix}A_{3,2}&A_{1,2}&A_{2,2}\\A_{3,3}&A_{1,3}&A_{2,3}\\A_{3,1}&A_{1,1}&A_{2,1}\end{bmatrix}$$

若B 代表一數獨矩陣，此B 矩陣爲一9×9魔方矩陣G 除以9 之後的餘數矩陣，而Q 代表一數獨矩陣，此Q 矩陣爲一9×9魔方矩陣G 除以9 之後的商數矩陣。則

$$B=\begin{bmatrix}A&A^{D^1L^1}&A^{D^2L^2}\\A^{U^1L^1}&A^{R^1}&A^{D^1}\\A^{U^2L^2}&A^{U^1}&A^{R^2}\end{bmatrix}=\begin{bmatrix}A_{1,1}&A_{2,1}&A_{3,1}&A_{2,3}&A_{3,3}&A_{1,3}&A_{3,2}&A_{1,2}&A_{2,2}\\A_{1,2}&A_{2,2}&A_{3,2}&A_{2,1}&A_{3,1}&A_{1,1}&A_{3,3}&A_{1,3}&A_{2,3}\\A_{1,3}&A_{2,3}&A_{3,3}&A_{2,2}&A_{3,2}&A_{1,2}&A_{3,1}&A_{1,1}&A_{2,1}\\A_{2,2}&A_{3,2}&A_{1,2}&A_{3,1}&A_{1,1}&A_{2,1}&A_{1,3}&A_{2,3}&A_{3,3}\\A_{2,3}&A_{3,3}&A_{1,3}&A_{3,2}&A_{1,2}&A_{2,2}&A_{1,1}&A_{2,1}&A_{3,1}\\A_{2,1}&A_{3,1}&A_{1,1}&A_{3,3}&A_{1,3}&A_{2,3}&A_{1,2}&A_{2,2}&A_{3,2}\\A_{3,3}&A_{1,3}&A_{2,3}&A_{1,2}&A_{2,2}&A_{3,2}&A_{2,1}&A_{3,1}&A_{1,1}\\A_{3,1}&A_{1,1}&A_{2,1}&A_{1,3}&A_{2,3}&A_{3,3}&A_{2,2}&A_{3,2}&A_{1,2}\\A_{3,2}&A_{1,2}&A_{2,2}&A_{1,1}&A_{2,1}&A_{3,1}&A_{2,3}&A_{3,3}&A_{1,3}\end{bmatrix}$$

其中水平、鉛直與對角線方向的元素皆不會重覆。

$$Q=\begin{bmatrix} P & P^{U^1R^1} & P^{U^2R^2} \\ P^{D^1R^1} & P^{L^1} & P^{U^1} \\ P^{D^2R^2} & P^{D^1} & P^{L^2} \end{bmatrix}=\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} & A_{2,2} & A_{2,3} & A_{2,1} & A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,3} & A_{2,1} & A_{2,2} & A_{1,2} & A_{1,3} & A_{1,1} & A_{3,1} & A_{3,2} & A_{3,3} \\ A_{1,3} & A_{1,1} & A_{1,2} & A_{3,2} & A_{3,3} & A_{3,1} & A_{2,1} & A_{2,2} & A_{2,3} \\ A_{1,2} & A_{1,3} & A_{1,1} & A_{3,1} & A_{3,2} & A_{3,3} & A_{2,3} & A_{2,1} & A_{2,2} \\ A_{3,2} & A_{2,3} & A_{3,1} & A_{2,1} & A_{2,2} & A_{2,3} & A_{1,3} & A_{1,1} & A_{1,2} \\ A_{2,2} & A_{3,3} & A_{2,1} & A_{1,1} & A_{1,2} & A_{1,3} & A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,1} & A_{2,2} & A_{2,3} & A_{1,3} & A_{1,1} & A_{1,2} & A_{3,2} & A_{3,3} & A_{3,1} \\ A_{1,1} & A_{1,2} & A_{1,3} & A_{3,3} & A_{3,1} & A_{3,2} & A_{2,2} & A_{2,3} & A_{2,1} \\ A_{3,1} & A_{3,2} & A_{3,3} & A_{2,3} & A_{2,1} & A_{2,2} & A_{1,2} & A_{1,3} & A_{1,1} \end{bmatrix}$$

其中水平、鉛直與對角線方向的元素皆不會重覆。

而

$$P=A^{D^1MT}=\begin{bmatrix} A_{1,3} & A_{2,3} & A_{3,3} \\ A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \end{bmatrix}^{MT}=\left\{ \begin{bmatrix} A_{1,3} & A_{2,3} & A_{3,3} \\ A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \right\}^T$$

$$=\begin{bmatrix} A_{3,3} & A_{2,3} & A_{1,3} \\ A_{3,1} & A_{2,1} & A_{1,1} \\ A_{3,2} & A_{2,2} & A_{1,2} \end{bmatrix}^T=\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \end{bmatrix}$$

$$P^{U^1R^1}=\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \end{bmatrix}\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$=\begin{bmatrix} A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \\ A_{3,3} & A_{3,1} & A_{3,2} \end{bmatrix}\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}=\begin{bmatrix} A_{2,2} & A_{2,3} & A_{2,1} \\ A_{1,2} & A_{1,3} & A_{1,1} \\ A_{3,2} & A_{3,3} & A_{3,1} \end{bmatrix}$$

$$P^{U^2R^2}=\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$=\begin{bmatrix} A_{1,3} & A_{1,1} & A_{1,2} \\ A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}=\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \end{bmatrix}$$

$$P^{D^1R^1}=\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \end{bmatrix}\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$=\begin{bmatrix} A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \\ A_{3,3} & A_{3,1} & A_{3,2} \end{bmatrix}\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}=\begin{bmatrix} A_{1,2} & A_{1,3} & A_{1,1} \\ A_{3,2} & A_{3,3} & A_{3,1} \\ A_{2,2} & A_{2,3} & A_{2,1} \end{bmatrix}$$

$$P^{D^2R^2}=\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$=\begin{bmatrix} A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \\ A_{3,3} & A_{3,1} & A_{3,2} \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}=\begin{bmatrix} A_{2,1} & A_{2,2} & A_{2,3} \\ A_{1,1} & A_{1,2} & A_{1,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \end{bmatrix}$$

$$P^{L^1}=\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}=\begin{bmatrix} A_{3,1} & A_{3,2} & A_{3,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ A_{1,1} & A_{1,2} & A_{1,3} \end{bmatrix}$$

$$P^{L^2}=\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \end{bmatrix}\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}=\begin{bmatrix} A_{3,2} & A_{3,3} & A_{3,1} \\ A_{2,2} & A_{2,3} & A_{2,1} \\ A_{1,2} & A_{1,3} & A_{1,1} \end{bmatrix}$$

$$P^{D^1}=\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \end{bmatrix}=\begin{bmatrix} A_{1,3} & A_{1,1} & A_{1,2} \\ A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \end{bmatrix}$$

$$P^{U^1}=\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} A_{3,3} & A_{3,1} & A_{3,2} \\ A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \end{bmatrix}=\begin{bmatrix} A_{2,3} & A_{2,1} & A_{2,2} \\ A_{1,3} & A_{1,1} & A_{1,2} \\ A_{3,3} & A_{3,1} & A_{3,2} \end{bmatrix}$$

因此

$G = 9Q + B + C$　　(C 爲一 9×9 階矩陣,且每個元素皆爲 1)

$$
= 9\begin{bmatrix}
A_{3,3} & A_{3,1} & A_{3,2} & A_{2,2} & A_{2,3} & A_{2,1} & A_{1,1} & A_{1,2} & A_{1,3} \\
A_{2,3} & A_{2,1} & A_{2,2} & A_{1,2} & A_{1,3} & A_{1,1} & A_{3,1} & A_{3,2} & A_{3,3} \\
A_{1,3} & A_{1,1} & A_{1,2} & A_{3,2} & A_{3,3} & A_{3,1} & A_{2,1} & A_{2,2} & A_{2,3} \\
A_{1,2} & A_{1,3} & A_{1,1} & A_{3,1} & A_{3,2} & A_{3,3} & A_{2,3} & A_{2,1} & A_{2,2} \\
A_{3,2} & A_{2,3} & A_{3,1} & A_{2,1} & A_{2,2} & A_{2,3} & A_{1,3} & A_{1,1} & A_{1,2} \\
A_{2,2} & A_{3,3} & A_{2,1} & A_{1,1} & A_{1,2} & A_{1,3} & A_{3,3} & A_{3,1} & A_{3,2} \\
A_{2,1} & A_{2,2} & A_{2,3} & A_{1,3} & A_{1,1} & A_{1,2} & A_{3,2} & A_{3,3} & A_{3,1} \\
A_{1,1} & A_{1,2} & A_{1,3} & A_{3,3} & A_{3,1} & A_{3,2} & A_{2,2} & A_{2,3} & A_{2,1} \\
A_{3,1} & A_{3,2} & A_{3,3} & A_{2,3} & A_{2,1} & A_{2,2} & A_{1,2} & A_{1,3} & A_{1,1}
\end{bmatrix} +
$$

$$
\begin{bmatrix}
A_{1,1} & A_{2,1} & A_{3,1} & A_{2,3} & A_{3,3} & A_{1,3} & A_{3,2} & A_{1,2} & A_{2,2} \\
A_{1,2} & A_{2,2} & A_{3,2} & A_{2,1} & A_{3,1} & A_{1,1} & A_{3,3} & A_{1,3} & A_{2,3} \\
A_{1,3} & A_{2,3} & A_{3,3} & A_{2,2} & A_{3,2} & A_{1,2} & A_{3,1} & A_{1,1} & A_{2,1} \\
A_{2,2} & A_{3,2} & A_{1,2} & A_{3,1} & A_{1,1} & A_{2,1} & A_{1,3} & A_{2,3} & A_{3,3} \\
A_{2,3} & A_{3,3} & A_{1,3} & A_{3,2} & A_{1,2} & A_{2,2} & A_{1,1} & A_{2,1} & A_{3,1} \\
A_{2,1} & A_{3,1} & A_{1,1} & A_{3,3} & A_{1,3} & A_{2,3} & A_{1,2} & A_{2,2} & A_{3,2} \\
A_{3,3} & A_{1,3} & A_{2,3} & A_{1,2} & A_{2,2} & A_{3,2} & A_{2,1} & A_{3,1} & A_{1,1} \\
A_{3,1} & A_{1,1} & A_{2,1} & A_{1,3} & A_{2,3} & A_{3,3} & A_{2,2} & A_{3,2} & A_{1,2} \\
A_{3,2} & A_{1,2} & A_{2,2} & A_{1,1} & A_{2,1} & A_{3,1} & A_{2,3} & A_{3,3} & A_{1,3}
\end{bmatrix}
$$

其中在幻方矩陣 G 內的元素，共有 81 個數學式，且每個數學式爲皆爲唯一，而每個元素值介於 1～81，因此可得到 G 內的每個元素不會重覆的唯一性。

(1) 存在性：

$$
A=\begin{bmatrix}
A_{1,1} & A_{2,1} & A_{3,1} \\
A_{1,2} & A_{2,2} & A_{3,2} \\
A_{1,3} & A_{2,3} & A_{3,3}
\end{bmatrix}
$$

利用排列組合觀念，可知 A 矩陣共有 9!=362880 組合情形，所以滿足 9×9 的魔方矩陣(水平、垂直與對角線方向和相等)，至少有 362880 組解答。

(2)唯一性：

證明魔方矩陣 G 內的元素皆不重覆，則每一種幻方矩陣 G 的答案組合也是唯一。

可將幻方中每個元素表爲 $9(A-1)+B$ ，A,B∈ {1,2,3,4,5,6,7,8,9}

設存在兩個相同的元素值 $9(A_1-1)+B_1=9(A_2-1)+B2$

其中 $(A_1,B_1)\neq(A_2,B_2)$

移項後 $9(A_1-A_2)=B_2-B_1$

$9|(B_2-B_1)$ 　　又 $0\leq(B_2-B_1)\leq 8$

所以 $B_2-B_1=0\Rightarrow B_2=B_1$

$A_1\neq A_2 \quad \Rightarrow (A_1-A_2)\neq 0$ (不合)

矛盾

原設錯誤

$\therefore$ 兩個相同的元素值 $9(A_1-1)+B_1=9(A_2-1)+B2$ 不存在

$\therefore$ 得證

接下來再以幻方回推數獨：

只要是 X 幻方，依然可以執行逆推。從 X 幻方以 9:1 或 1:9 的比例分開爲兩個共軛數獨，方法與 4×4 X 幻方相同，於此不再贅述。

三、 25×25 **數獨-幻方**探討：

九個值的填法如下：

A1		C1		
	A2	C2		D2
B1	B2	A3	B4	B5
		C4	A4	
	D1	C5		A5

其中需滿足　A1+A2+A3+A4+A5

=B1+B2+A3+B4+B5

=C1+C2+A3+C4+C5

=D1+C4+B4+D2+A1

此單位宮格 H 即滿足填法：

1	15	22	18	9
23	19	6	5	12
10	2	13	24	16
14	21	20	7	3
17	8	4	11	25

(以下皆以此單位宮格作範例)

將單位宮格內的順序加以調整，第一行平移至最後一行，稱爲”左平移”；類似的，第一列平移至最後一列，稱爲”上平移”。以滿足此填法的九宮格填入 9×9 方陣的左上宮格，每往下一九宮格就執行一次”左平移”後填入，每往右一宮格就執行一次”上平移”後填入後，得到”初始數獨”。爾後將每格內的值減 1 可得到一組由 0~24 組成的 25×25 數獨 M。將數獨 M 與數獨 N 內的值以 25:1 相加之後得到一 25×25 方陣再將每一格內的值加上 1 即可得到 25×25 的 X 幻方：

8	371	527	444	215	567	464	135	123	276	250	28	316	587	384	336	507	499	155	68	404	195	88	256	622
554	470	138	106	297	233	46	302	594	390	342	514	485	173	51	425	178	91	262	609	11	357	549	430	218
236	32	324	580	393	329	520	488	156	72	408	196	77	269	615	17	364	535	448	201	575	453	141	112	284
350	503	491	162	59	411	182	99	255	618	4	370	538	431	222	558	471	127	119	290	242	39	310	598	376
417	189	85	273	601	25	353	541	437	209	561	457	149	105	293	229	45	313	581	397	333	521	477	169	65
365	533	446	202	19	451	142	114	285	573	34	325	578	391	237	518	486	157	74	330	197	79	270	613	406
472	129	120	288	556	40	308	596	377	244	501	492	164	60	348	184	100	253	616	412	368	536	432	224	5
43	311	582	399	230	522	479	170	63	331	190	83	271	602	419	351	542	439	210	23	459	150	103	291	562
509	500	153	66	337	193	86	257	624	405	372	529	445	213	6	465	133	121	277	569	26	317	589	385	248
176	92	264	610	423	359	550	428	216	12	468	136	107	299	555	47	304	595	388	231	515	483	171	52	344
544	440	208	21	352	148	101	292	564	460	312	584	400	228	41	480	168	61	332	524	81	272	604	420	188
131	122	279	570	463	319	590	383	246	27	498	151	67	339	510	87	259	625	403	191	530	443	211	7	374
305	593	386	232	49	481	172	54	345	513	94	265	608	421	177	548	426	217	14	360	137	109	300	553	466
487	159	75	328	516	80	268	611	407	199	531	447	204	20	363	144	115	283	571	452	323	576	392	239	35
98	251	617	414	185	537	434	225	3	366	130	118	286	557	474	306	597	379	245	38	494	165	58	346	502
427	219	15	358	546	110	298	551	467	139	591	387	234	50	303	174	55	343	511	482	263	606	422	179	95
113	281	572	454	145	577	394	240	33	321	160	73	326	517	489	266	612	409	200	78	449	205	18	361	532
599	380	243	36	307	163	56	347	504	495	252	619	415	183	96	435	223	1	367	539	116	287	559	475	128
166	62	334	525	478	274	605	418	186	82	438	206	22	354	545	102	294	565	458	146	585	398	226	42	314
260	623	401	192	89	441	212	9	375	528	124	280	568	461	132	588	381	247	29	320	152	69	340	508	496
221	2	369	540	433	289	560	473	126	117	378	241	37	309	600	57	349	505	493	161	620	413	181	97	254
295	563	456	147	104	396	227	44	315	583	64	335	523	476	167	603	416	187	84	275	207	24	355	543	436
382	249	30	318	586	70	338	506	497	154	621	402	194	90	258	214	10	373	526	442	278	566	462	134	125
53	341	512	484	175	607	424	180	93	261	220	13	356	547	429	296	552	469	140	108	389	235	48	301	592
614	410	198	76	267	203	16	362	534	450	282	574	455	143	111	395	238	31	322	579	71	327	519	490	158

(其中每行每列對角線相加後直皆爲 7825)

相同的，若數獨 M 與數獨 N 內每一格以 1:25 的比例相加，也可以得到另外一組幻方方陣。我們再次稱數獨 M 與數獨 N 有共軛關係。

證明與9×9的証明方式相同。

回推數獨與9×9的方法相同。

陸、結果討論：

當初在三年前的科學人雜誌曾看到其中一篇討論數獨初盤的文章。裡面討論使一數獨有唯一解所需填進去的最小數目，當時看到的結果是 17 個數字。剛開始覺得這個討論很有趣，便想利用做科展這個機會「參一腳」。畢竟這算是很新的題目，所以能找的資料也很有限，於是後來便想到跟另外一種方陣題目(遊戲)結合，試著找出之間的關係，這樣不論是在數獨或者是幻方都能多一種不同的填製法。

一開始先鎖定較簡單的4×4數獨，為了讓數獨具備特殊性質(對角線上)，我們特別創造了”G 數獨”---一組對角線上數字均不重複的數獨，使之在對角線上的和仍然與行列相同。利用幻方每行每列對 4 做模後為完全剩餘系之性質我們順利的找出一種變換方法，即在4×4 G 數獨固定一3×3方陣的四角，將固定角之相鄰元素互換可得一共軛數獨與原數獨以$1:4$及$4:1$比例構成 X 幻方(由於找不到有關此變換法的資料，所以應是原創)。之後又找到「井字變換法」亦可構成 X 幻方。

但一切似乎完全停滯在4×4的情況中，原本想利用 G 數獨的性質一股作氣將9×9和16×16的情況一一解決，但用 C 語言程式跑了整整一天還無法製造任何一組符合9×9的 G 數獨，因此我們開始懷疑9×9 G 數獨的存在性，當然也可能只是程式設計效率不好。於是我們改變方法利用 Excel 的自訂函式功能來找尋可能的變換法。也成功地找出一個較廣義的變換法，其中仍是以兩共軛 G 數獨以$1:n^2$及$n^2:1$的比例製作 X 幻方。

其實這樣的一組”數獨-幻方”，可以當作機密文件的密碼鎖，只要給予其中幾格的值，經過已編寫好的密碼轉換程式(其實就是執行”數獨-幻方”變換的方法)，儼然成為加密系統。可想而知，階數越高的”數獨-幻方”會更加複雜，因此”數獨-幻方”將會與之有實際上的應用。

柒、未來展望

一、密碼鎖在應用程式上的應用。

二、數獨-幻方與棋盤問題類似，應有機會與之連結。

三、儘可能找出所有的變幻法及其規律。

捌、參考資料

尤怪，尤怪之家，http://www.shes.hcc.edu.tw/~oddest/index.htm

狄拉賀（Jean-Paul Delahaye），數獨樂樂無窮，2006 年 7 月科學人雜誌

捌、附錄

---------4×4 G 數獨 C++程式內容-----------

```
#include <fstream>#include <algorithm>#include <iostream>
using namespace std;
struct FreqList{
	int num; int count;
	bool operator<(const FreqList& another) const{
		return (count > another.count);}};
const int boardSize = 4;
int gameBoard[boardSize][boardSize] = {{0,0}};
int columnInRow[boardSize + 1][boardSize];
bool isInColumn[boardSize + 1][boardSize];
bool isFixed[boardSize + 1][boardSize];
FreqList processList[boardSize];
float TOTAL_MAGIC=0;   //幻方組數
float TOTAL_COUNT=0;     //數獨組數
void occupy(const int row, const int col, const int num){
	gameBoard[row][col] = num;
	columnInRow[num][row] = col;
	isInColumn[num][col] = true;}
void free(const int row, const int col){
	int &num = gameBoard[row][col];
	columnInRow[num][row] = -1;
	isInColumn[num][col] = false;
	num = 0;}
void initialize(){       // 初始化數字位置的矩陣
	for (int num = 0; num <= boardSize; ++num){
		for (int index = 0; index < boardSize; ++index){
			columnInRow[num][index] = -1;
			isInColumn[num][index] = false;
```

```
                isFixed[num][index] = false;}}//  初始化程序表
        for (int num = 1; num <= boardSize; ++num){
            processList[num - 1].num = num;
            processList[num - 1].count = 0;}//  輸入初始數據
        ifstream dataFile("mcnp.in");
        for (int rowIndex = 0; rowIndex < boardSize; ++rowIndex){
            for (int columnIndex = 0; columnIndex < boardSize; ++columnIndex){
                int temp;
                dataFile >> temp;
                if (temp){
                    occupy(rowIndex, columnIndex, temp);
                    isFixed[temp][rowIndex] = true;
                    ++processList[temp - 1].count;}}}
        dataFile.close();
        //處理程序表列行排序
        sort(processList, processList + boardSize);
void print(ostream& resFile,ostream& MagicFile);
bool place(const int, const bool);
void process(){
        //G 數獨輸出檔
        ofstream MagicFile("magic.out");
        bool isPlaced[boardSize] = {false};
        int stackTop = 0;
        while (stackTop > -1){
            if (stackTop == boardSize){
                print(resFile,MagicFile);
                --stackTop; }
            else{
                isPlaced[stackTop] =
place(processList[stackTop].num,isPlaced[stackTop]);
                if (isPlaced[stackTop])
                    ++stackTop;
                else
                    --stackTop;}}
        //最後輸出組數至檔案
        resFile << "共有" << TOTAL_COUNT << "組數獨答案";
        resFile.close();
        MagicFile << "共有" << TOTAL_MAGIC << "組幻方答案";
        MagicFile.close();}
bool place(const int num, const bool hasBeenPlacedBefore){
```

```
        int currentRow;
        if (!hasBeenPlacedBefore){
                currentRow = 0;
                while ((currentRow < boardSize) && isFixed[num][currentRow])
++currentRow;}
        else{
                currentRow = boardSize - 1;
                while ((currentRow >= 0) && isFixed[num][currentRow]) --currentRow;}
        while (currentRow > -1){
                if (currentRow == boardSize) return true;
                else{
                        int& prevColumn = columnInRow[num][currentRow];
                        int currentColumn = prevColumn + 1;

                        while (currentColumn < boardSize
                                        && (isInColumn[num][currentColumn]
                                            || gameBoard[currentRow][currentColumn] != 0))
                                ++currentColumn;
                        //  找不到下一合適的列
                        if (currentColumn == boardSize){
                                //找不到，釋放本行數據，回溯到上一可以修改的行
                                if (prevColumn > -1)
                                        free(currentRow, prevColumn);
                                --currentRow;
                                while ((currentRow >= 0) && isFixed[num][currentRow])
--currentRow;}
                        else{
                                //找到了，填充新位置，選擇下一可以修改的行
                                if (prevColumn > -1)
                                        free(currentRow, prevColumn);
                                occupy(currentRow, currentColumn, num);
                                ++currentRow;
                                while ((currentRow < boardSize) && isFixed[num][currentRow])
++currentRow;}}}        return false;}
void print(ostream& resFile,ostream& MagicFile)
{        //檢查是否為 G 數獨
        int i=0,sum=1,mul_s=1;
        int MAGIC[boardSize][boardSize];
        bool Is_MAGIC=true;
        //先將目前的答案儲存至 MAGIC 陣列
```

```
        for (int rowIndex = 0; rowIndex < boardSize; ++rowIndex)
        {           for (int columnIndex = 0; columnIndex < boardSize; ++columnIndex)
{ MAGIC[columnIndex][rowIndex]=gameBoard[rowIndex][columnIndex];}}
        // 1*2*3....*(boardSize-1)
        for (i=0;i<boardSize;i++)
        {mul_s=(i+1)*mul_s;}
        //左上至右下
        i=0;
        while (i<boardSize)
        { sum=sum*MAGIC[i][i];
           i++; }
        if (mul_s!=sum)    //不是 G 數獨,則不印出
              Is_MAGIC=false;
        // reset
        sum=1;
        //右上至左下
        i=boardSize-1;
        while (i>-1)
        {sum=sum*MAGIC[i][i];
           i--;}
        if (mul_s!=sum) //不是 G 數獨,則不印出
              Is_MAGIC=false;
        for (int rowIndex = 0; rowIndex < boardSize; ++rowIndex)
        {for (int columnIndex = 0; columnIndex < boardSize; ++columnIndex)
                { resFile << gameBoard[rowIndex][columnIndex] << " ";
                        //如果是 G 數獨,輸出
                        if (Is_MAGIC==true)
                                MagicFile << gameBoard[rowIndex][columnIndex] << " ";}
                resFile << endl;
                //如果是 G 數獨,輸出換行
                if (Is_MAGIC==true)
                                MagicFile << endl;}
        resFile << endl;
        TOTAL_COUNT=TOTAL_COUNT+1;
        //如果是 G 數獨,輸出換行
        if (Is_MAGIC==true){
                      MagicFile << endl;
                      TOTAL_MAGIC=TOTAL_MAGIC+1;}}
int main(){
        initialize();
```

```
process();
cout << "總共有" << TOTAL_MAGIC << "組 G 數獨答案!" << endl;
system("PAUSE");
return 0;}
```

-------------程式結束-----------

-------------mcnp.in 內容-----------

```
0 0 0 0
0 0 0 0
0 0 0 0
0 0 0 0
```

--------------內容結束---------------

--------------mcnp.out 內容---------------

………(G 數獨略)

共有 48 組幻方答案

----------------內容結束------------------

【評語】040410

1) 科展重視實驗過程，說明如何由失敗過程中摸索出成功的線索。本作品以膾炙人口的遊戲爲研究對象以趣味數學的方法作探討，應用計算機的蠻力獲得了一些靈感。然而本作品的成果與理想的目標尚有距離。

2) 請注意別字：應該是「變換法」，而不是「變幻法」(作品說明書 P.25)。

3) 本作品學習到電腦應用的寶貴經驗：人們往往透過試算表 Excel 獲得比 C 語言程式更豐富的數學靈感！