投稿類別：數學類

篇名：解開 2048 的謎

作者：
陳思嘉。新竹市立建功高中。高二 **7** 班。
曹子涵。新竹市立建功高中。高二 **7** 班。

指導老師：
詹佩珊老師

壹●前言

一、研究動機

最近我們都很熱衷於 2048 這款遊戲，遊戲方法非常簡單，但是想要達成 2048 卻很難，每次只差臨門一腳就能達到 2048，因此都沒有成功過。於是我們就決定要來研究，用甚麼方法達到 2048 比較容易，又或者和出現方塊 2 和方塊 4 的比例是不是有關聯。

此外，當我們從網路上得知一些 2048 遊戲策略後，依著該原則進行遊戲時，果然順利達到比較高的分數，但到最後就變成單純的滑動，於是我們就想研究看看是不是有某種邏輯，或是只要有耐心地撐到最後，就能夠達到 2048 呢？因此我們決定尋找 2048 的種種奧妙。

二、研究目的

1.分析 2048 遊戲規則與並探討致勝攻略。

2.實測不同遊戲攻略並分析其優劣

3.探討方塊 2 及方塊 4 出現數量與達成 2048 快慢之關係。

三、研究架構

影響 2048 之可能因素		遊戲過程數據取得		
1. 遊戲攻略 2. 遊戲規則 3. 理論分析	→	1. 遊戲攻略比較 2. 方塊 2、4 出現數量影響	→	結論

貳●正文

一、2048 遊戲規則與致勝攻略

（一）2048 遊戲規則

2048 遊戲規則[5]是這樣，開始遊戲時會隨機出現 2 個數字，2 或 4 的方塊（以下簡稱方塊 2 及方塊 4），理論上應該有三種可能的初始狀態，包含兩個方塊 2、方塊 2 及方塊 4 各一或兩個方塊 4，如圖 1 所示。

圖 1：2048 初始畫面

接下來可以按↑←↓→四個按鍵，按↑時所有方塊會全部移到最上方，相鄰方塊數字相同時則會合併成一個方塊，該方塊的數字是為合併方塊數字和，例如兩個方塊 2 合併後會得到方塊 4。按←↓→則分別往左、下及右方集中。

圖 2 即是 2048 遊戲前三步驟的畫面擷圖，圖 2 第一個圖是啟動 2048 遊戲時得到的

2 個方塊 2，按←後原本的 2 個方塊 2 全部向左移，然後右下角出現新的方塊 2（如圖 2 中）。再按↓時原本的 3 個方塊會向下移，其中最左邊相鄰的 2 個方塊 2 結合成方塊 4，其上方則出現新的方塊 2。

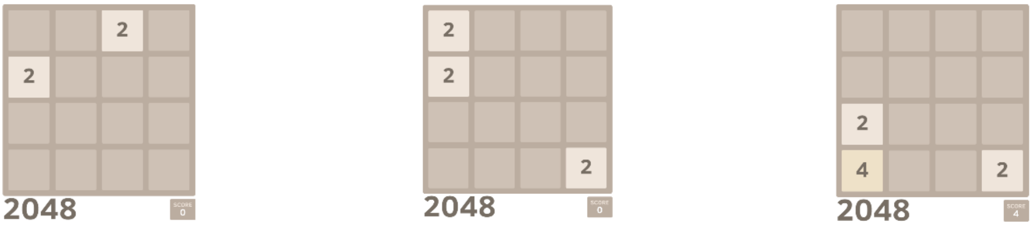

圖 2：2048 遊戲進行前三步

遊戲中是利用↑←↓→移動方塊，使相同數字的方塊能合成更大的數字，例如兩個 2 便能和成 1 個 4、2 個 256 便能和成一個 512……以此類推。而每按下一次↑←↓→鍵時，方塊移動合成結束後，會隨機出現一個新的方塊 2 或方塊 4。所以 2048 的目的就是藉由合成數字方塊，來形成一個方塊 2048，只要合成兩個方塊 1024，出現方塊 2048 的時候，便算是玩家勝利，但仍可以選擇繼續往下挑戰更高的分數。

（二）2048 的遊戲攻略

經過本研究實際嘗試 2048 的遊戲，如果沒有規則隨意滑動，通常很難 2048，最多只能合成到 512，而 1024 也是很少出現，但是也有玩家可以到 2048 甚至更大的數字。於是我們就上網查詢有關 2048 的遊戲攻略[1][4]，其規則都是盡可能往某一個方向集中，而在「停不下來的 2048」科展報告中[3]，將 2048 遊戲攻略進一步分為三種，並以理論分析其致勝原理。

這三種 2048 遊戲攻略分別是「L 形堆疊法」(以下簡稱「L 型法」)、「垂直掉落堆疊法」(以下簡稱「垂直法」)、「一行角落堆疊法」(以下簡稱「一行法」)，本研究將其攻略原則整理如下表。

表 1：2048 遊戲攻略（整理「停不下來的 2048」[3]）

L 型法	1.集中在 4x4 遊戲區右下角，只能往右或往下滑 2.將最大數字方塊放在最角落。 3.依序將數字，由下至上，由大到小依序排於 L 型的角落。
垂直法	1.集中於 4x4 遊戲區的最下方，只能向左、向右或向下滑。 2.將最大的數字疊在最下層，最下層由左到右或由右到左疊滿後，再依序向上堆疊 3.當左右都無法滑動，也無法向下滑動時，可以向上滑一次。
一行法	1.集中於介面的最左邊的直行，並且使最左邊直行穩定不移動 2.依序由下到上，由大到小排列。 3.當往左滑及往上滑都無法滑動時，才考慮向下滑，萬不得已時才向右滑。

綜合以上三種遊戲攻略，均強調減少一至二個滑動方向，例如 L 型法盡可能不向左滑亦不向右滑，而垂直法盡可能不向上滑，一行法盡可能不向右及向下滑。最主要原因是要讓方塊 2 及方塊 4 的位置能集中在某個區域，已合成之 8 以上的數字方塊集中於另一區，以利相同數字方塊的合成，也能減少合成較大數字方塊的步數，因為步數一旦增多，即表示會出現方塊 2 或方塊 4，形成相同數字方塊之間的障礙。

（三）合成 2^n 需要的方塊 2 及 4 的數量估計

2048 的遊戲運用的數學原理很簡單，一開始出現方塊 2 或方塊 4，然後相鄰之相同數字的方塊可以加總成為新的方塊，從排列組合的角度來分析[2]，假設合成方塊 2^n 需要 x 個 2 及 y 個 n，則 $2x+4y=2^n$，其中 x 及 y 均為 0 或正整數。

以合成一個方塊 8 為例，要尋找 $2x+4y=8$ 的 0 或正整數解，其可能解為(4,0)、(2,1)及(0,2)，也就是可以是四個 2 合成一個 8，或是二個 2 及一個 4，又或者是二個 4，共有 3 種組合方式，如圖 3 所示。圖 4 則是方塊 16 有 5 種可能的組合方式，依此類推：

2^n 至少需要 $2^{(n-2)}$個方塊 4，最多需要 $2^{(n-1)}$個方塊 2

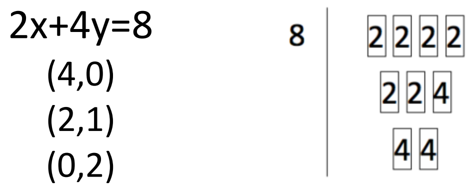

圖 3：合成方塊 8 可能之方塊組合

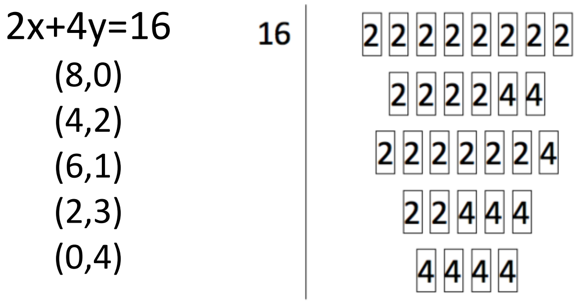

圖 4：合成方塊 16 可能之方塊組合

但事實上並不是如此，如果所出現的兩個方塊 4 是在同欄或同列，且中間並沒有其他的數字方塊，那麼一步即可合成方塊 8；但是如果不在同欄或同行，那就會多於 1 步，所以通常實際上合成某個數字方塊時，通常會多於預估方塊 2 的數量。

理論上一個方塊 2^n，至少需要 $2^{(n-1)}$個方塊 2，最多需要 $2^{(n-2)}$個方塊 4，而 2048 遊戲過程中，方塊 2 及方塊 4 是隨機出現，所以合成一個方塊 2^n，其所需要步數應介於 $2^{(n-2)}$與 $2^{(n-1)}$之間（包含初始會有 2 個方塊 2 或方塊 4），但是通常合成某個數字方塊時，4x4 方格中通常會留有部份未合成的其他數字方塊，致使步數會多於實際合成一個數字方塊所需之方塊數。

二、三種遊戲攻略是否真的容易 2048 之比較

依 2048 遊戲規則來看，第一點中所提之「L 型法」、「一行法」及「垂直法」三種遊戲攻略，理論上都是採取依數字大小有系統的排列，並且盡量減少往某一個方向滑動，以防止新產生的方塊 2 及方塊 4 可能成為合成較大數字的阻礙，但是究竟何種方法會有較佳效率，我們將在這個實驗中以實際的遊戲過程數據作探討。

（一）撰寫能記錄遊戲過程之 2048 程式

曾經嘗試利用截圖方式記錄過程，但是有時會不小心漏掉某個畫面，而事後又要以人工方式進行比對，花費時太多無法取得大量數據，而網路上也找不到可以記錄過程的 2048 程式，，所以我們決定運用在高一的程式語言課程[6]中所學之 Visual BASIC 2010 自行撰寫程式。

1. 首先依 2048 遊戲規則，以 Visual Basic 的主控台模式撰寫程式，因主控台模式不需要處理圖形化介面，因此本研究是以簡單的數字介面配合普通文字檔(.txt）儲存遊戲過程每次滑動方向、方塊 2 或方塊 4 出現的位置，以方便日後研究分析。
2. 分別以「L 型法」、「垂直法」及「一行法」等三種遊戲攻略進行遊戲過程之記錄，取得能合成方塊 1024 之至少 10 次之遊戲過程記錄。
3. 再從前述遊戲記錄中，選取 10 次可合成方塊 1024 以上的數據，分別計算平均步數和方塊 2、方塊 4 出現的數量，並探討數據間的關係

（二）三種攻略合成 32、64、128、256、512、1024 及 2048 步數與估計時間

本研究分別以「L 型法」、「垂直法」及「一行法」等三種遊戲攻略各進行 41 次遊戲過程的記錄，其中「L 型法」有 32 筆數據達到 512，12 筆數據達到 1024，1 筆數據達到 2048。「垂直法」有 29 筆數據達到 512，10 筆數據達到 1024，而只有第 8 筆數據有達到 2048。「一行法」有 34 筆數據達到 512，10 筆數據達到 1024，而只有第 7 筆數據達到 2048。

接著將「L 型法」、「垂直法」及「一行法」三種方法各 41 次遊戲結束時，方塊最大值的次數整理於表 2，各欄位數據分別表示遊戲結束時方塊最大值的次數及比例。

表 2：三種方法合成方塊 32、64、128、256、512、1024 及 2048 的比例

	合成 32	合成 64	合成 128	合成 256	合成 512	合成 1024	合成 2048
L 型法	1/2%	0/0%	3/7%	5/12%	**20/49%**	11/27%	1/2%
垂直法	0/0%	0/0%	1/2%	10/24%	**20/48%**	9/22%	1/2%
一行法	0/0%	0/0%	3/7%	4/10%	**24/59%**	9/22%	1/2%

從表 2 可以看出「L 型法」、「垂直法」及「一行法」三種方法遊戲結束時方塊最大值以 512 所佔比例最高，也就是說若運用遊戲攻略，多半能順利合成至 512，但若要 1024 甚至 2048，需要更多的耐力與智力。

再計算「L 型法」、「垂直法」及「一行法」三種方法合成 32、64、128、256、512、1024、2048 時之平均步數，並以滑動一步費時 3 秒計算，將數據整理於表 3，其中各欄位分別記錄兩筆數據，第一筆指的是合成該數字時平均花費的步數，第二筆則是以一步花費 3 秒估計花費的分鐘數。

表 3：合成方塊 32、64、128、256、512、1024、2048 時之平均步數與估計時間

	合成 32	合成 64	合成 128	合成 256	合成 512	合成 1024	合成 2048
L 型法	18.0/0.9	34.2/1.7	62.1/3.1	117.6/5.9	**210.5/10.5**	393.6/19.7	714/35.7
垂直法	19.4/1.0	35.9/1.8	64.5/3.2	119.1/6.0	**216.8/11.0**	403.4/20.2	793/39.7
一行法	18.1/0.9	33.5/1.7	61.5/3.1	110.3/5.5	**210.4/10.4**	387.3/19.4	692/34.6
方塊 2	2^4=16	2^5=32	2^6=64	2^7=128	2^8=256	2^9=512	2^{10}=1024
方塊 4	2^3=8	2^4=16	2^5=32	2^6=64	2^7=128	2^8=256	2^9=512

欲合成 512，「L 型法」的是 210.5 步，估計需要 10.5 分鐘、「垂直法」是 216.8 步，估計需要 11.0 分鐘、「一行法」是 210.4 步，估計需要 10.4 分鐘，無論是那一種方法，合成 512 大約需 10 分鐘。若要合成 1024，「L 型法」的是 393.6 步，估計需要 19.7 分鐘、「垂直法」是 403.4 步，估計需要 20.2 分鐘、「一行法」是 387.3 步，估計需要 19.4 分鐘，無論是那一種方法，合成 1024 大約需要 20 分鐘。

數據中有三次合成 2048，其所花費的步數及估計時間分別是「L 型法」為 714 步，估計需要 35.7 分鐘，「垂直法」為 793 步，估計需要 39.7 分鐘，以及「一行法」為 692 步，估計需要 34.6 分鐘。

從合成 1024 的平均步數來看，以「一行法」所需步數最少，同時「一行法」合成 512 比例為 59%亦是三種方法中最高。

從「一行法」的攻略來分析，由於將較大數字集中一行，因此可以空出 4x3 製造區，再者往左上堆疊，集中一角落，依照數字大小順序排列，新出現的數字也不容易影響到大數字，因而大部分都能以較少步數達到 1024，估計所花費的時間是最少的。

「垂直法」則是比較冒險的玩法，先預留一些數字，以便在無法滑動時，還有拯救的機會，也因為一直往下堆疊，有時候會同步滑動合成，步數也沒有想像中的大，但較不容易合成較大數字，或者合成之數字方塊較分散（不相鄰）致使無法合併。

「L 型法」有 3x3 的製造區，但是這種方法是一直往右下角集中，是不停往右合成的，所用的步數雖不多，但是也因為不方便依數字大小排列，缺少同步合成（一次合成 2 個以上的方塊）的機會，所以需要較多步數才能合成較大數字，有如表格中的平均值，到達 1024 的步數，是排名第二的。

再比較同一種攻略的合成 32、64、128、256、512、1024、2048 時之平均步數與估計時間發現，合成 1024 所花費的步數是合成 512 的 2 倍左右，因為當合成 512 時，必需要設法再合成另一個 512 才能合成 1024，因此所花費步數應呈現 2 倍成長，我們將三種方法合成 32、64、128、256、512、1024、2048 的平均步數，並與合成 2^n 所需之方塊 2 數量及方塊 4 數量以折線圖繪製如圖 5。

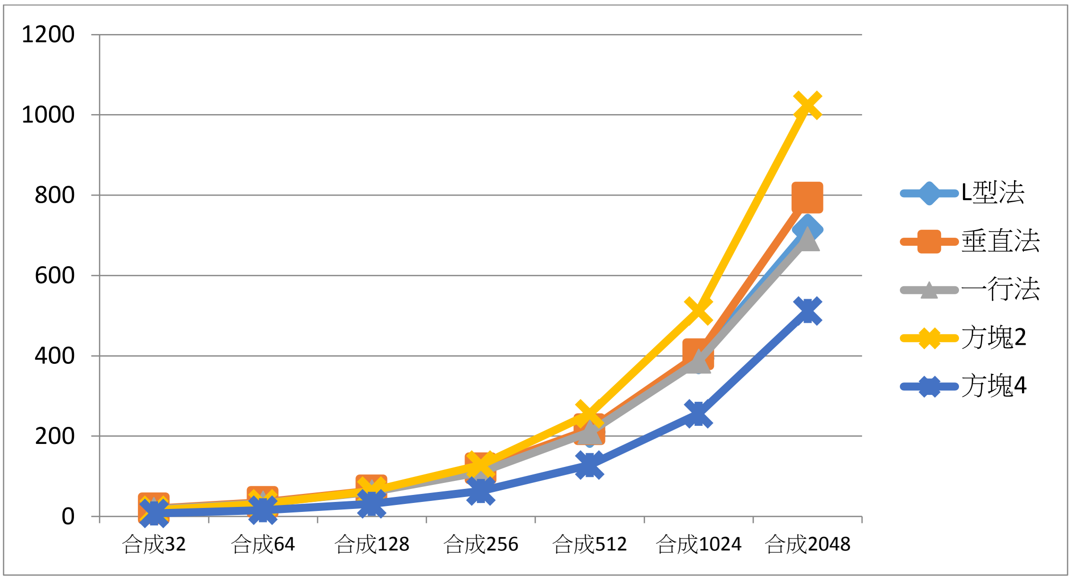

圖 5：三種方法與方塊 2 及方塊 4 需求數折線圖

從圖 5 可以看出「L 型法」、「垂直法」及「一行法」合成 32、64、128、256、512、1024、2048 所需步數幾乎差不多，其圖形呈現 2 的指數成長，但都少於理論上需求之方塊 2 數量，多於理論上需求之方塊 4 數量，主要是因為合成方塊 2n 時，通常 4x4 的方格內仍會有其他未合成的數字方塊。其折線圖在合成 256 時，其所花費步數與組合 256 理論上最多只需 128 個 2 接近，而略高於理論上需求最少的方塊 4 數量（因為遊戲中不會只有方塊 4），但是合成 512 及 1024 時則開始明顯低於 256 及 512 個 2，同時也明顯高於

方塊 4 的數量。

（三）「L 型法」、「垂直法」及「一行法」三種遊戲攻略優劣分析

在「L 型法」、「垂直法」及「一行法」三種遊戲攻略中，以「一行法」合成 1024 時所需平均步數最少，但以「L 型法」較容易 1024。其原因與三種攻略的製造區大小有關，說明如下：

圖 3 是分別就「L 型法」、「垂直法」及「一行法」三種攻略在遊戲結束時，其所合成之最大方塊為 512、1024 及 2048 時之最後一個畫面，其中紅色方框圈出來的部分為製造區，而藍色框框圈出來的部分為堆疊區。

	L 型法	垂直法	一行法
512	2 4 2 8 4 8 4 16 2 16 256 32 4 8 64 512	2 8 2 4 8 16 4 8 32 128 32 16 256 16 512 128	512 2 16 8 2 32 128 16 8 16 2 4 2 4 8 2
1024	4 8 2 16 2 4 16 2 4 16 256 4 2 8 32 1024	4 2 4 2 16 8 16 4 32 128 64 16 64 512 1024 256	2 64 8 4 64 128 16 8 128 64 32 4 1024 32 8 2
2048	2 4 64 256 4 16 256 1024 2 32 128 2048 4 8 16 64	2 4 8 2 8 32 16 8 2 64 128 32 128 512 2048 256	32 4 32 4 2048 64 8 2 128 32 4 8 8 4 2 4

圖 6:三種方法達到 512、1024、2048 之遊戲結束畫面

從圖 6 可以發現，「L 型法」的製造區只有 3x3，相對較小，但因為製造區部分排列由大到小，有順序的排列造成合成方塊容易，由表 6 也可看出，以最大值 512 結束遊戲的比例是三種方法最少的，代表當「L 型法」達到較大值時，後面合成容易達到 2048，但平均步數排序第二，所花費時間多。

「垂直法」的製造區有 3x4，是較大的製造區，不過製造區的部分排列沒有按照次序由大到小，會造成製造區一分為二，如圖中圈出的黑色框框，除了要合成右邊的方塊也要合成左邊的方塊，由於相同的方塊距離遙遠，所以不易合成。（如圖 7）

4	2	4	2
16	8	16	4
32	128	64	16
64	512	1024	256

圖 7: 「垂直法」第一筆數據的結束畫面

「一行法」的製造區和「垂直法」的製造區同為 3x4，但因製造區的數字方塊很有順序地由左到右、由大到小排序（如圖 8），雖然表 2 中以最大值 512 結束遊戲的比例占了 59%，比其他方法都高，代表「一行法」容易合成 512，而且在三個方法之中，平均步數又是最小的，花費時間最少。

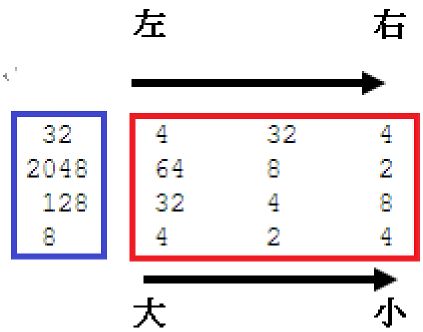

圖 8：「一行法」排列順序說明圖

（四）過程中 2 及 4 出現的數量對合成 1024 之影響

2048 遊戲規則是每滑動一步會出現一個方塊 2 或方塊 4，過程中兩種方塊出現的比例是否亦會影響合成步數，因此我們從三種方法各 10 次達成 1024 的數據取得過程中方塊 2 及方塊 4 出現的數量整理於表 4。

表 4：三種方法合成方塊 1024 期間所出現 2、4 的數量及平均值

		1	2	3	4	5	6	7	8	9	10	平均	比例
L 型法	2	173	197	180	226	183	213	219	190	198	223	200.2	0.50
	4	179	185	203	185	190	220	228	198	184	212	198.4	0.49
垂直法	2	170	216	222	209	172	192	209	234	193	195	201.2	0.49
	4	202	248	225	184	183	203	207	215	184	191	204.2	0.50
一行法	2	193	193	198	226	170	219	187	206	190	186	196.8	0.50
	4	220	190	196	155	194	209	191	188	189	193	192.5	0.49

從表 9 的結果發現各次數據方塊 2 與方塊 4 出現之比例相當，各有 50%左右，無法看出方塊 2 或方塊 4 的數量是否影響合成步數。

三、方塊 2 或方塊 4 出現比例是否比較容易 2048

從表 4 發現，方塊 2 及方塊 4 數量差不多，因此若遊戲中只出現方塊 2，或者只出現方塊 4，是不是比較容易達成 2048，於是我們設計了兩個程式，一個程式只會出現方塊 2（以下以 olny2 簡稱），而另外一個則是只會出現方塊 4（以下以 olny4 簡稱），並探討方塊 2 及方塊 4 對合成 2048 有所影響。「一行法」較其他兩種攻略能以較少步數合成較大數字，所以在此將全部採取「一行法」遊戲攻略進行實驗數據的取得。

（一）修改 2048 程式為只出現方塊 2、方塊 4

1. 修改程式成 only2（只出現方塊 2 的程式）和 only4（只出現方塊 4 的程式），並以普通文字檔（.txt）記錄遊戲過程。

2. 以「一行法」攻略分別進行 only2 及 only4 各 20 次的遊戲，並記錄其過程。

3. 記錄遊戲進行中合成 8、16、32、64、128、512、1024、2048 時所花費的步數。

（二）only 2 或 only4 何者較容易 2048

再以「一行法」分別使用 only2 及 only4 程式進行 20 次的遊戲過程記錄，其中 only2 的 20 次記錄中，最高能合成到 4096（有 1 次），有 10 次可達 2048，而 only4 最高能合成到 4096（有 6 次），有 10 次可達 2048。本研究將 only2 及 only4 各 20 次遊戲結束時方塊最大值的次數整理於表 5，各欄位數據分別表示遊戲結束時方塊最大值的次數及比例。

表 5：only2、only4 合成方塊 32、64、128、256、512、1024 及 2048、4096 的比例

	合成 32	合成 64	合成 128	合成 256	合成 512	合成 1024	**合成 2048**	合成 4096
only2（20 次）	0/0%	0/0%	0/0%	0/0%	1/5%	9/45%	**9/45%**	1/5%
only4（20 次）	0/0%	0/0%	0/0%	0/0%	0/0%	4/20%	**10/50%**	6/30%

從表 5 可以看出 only2、only4 遊戲結束時方塊最大值以 2048 所佔比例最高。此時 only2 平均步數為 1215.4、only4 平均步數為 636.2。換言之，只出現方塊 2 或只出現方塊 4 非常容易合成 2048。

再計算 only2、only4 合成 32、64、128、256、512、1024、2048、4096 時之平均步數，並以滑動一步費時 3 秒計算，將所得數據整理於表 6，其中各欄位分別記錄兩筆數據，第一筆指的是合成該數字時平均花費的步數，第二筆則是以一步花費 3 秒估計花費的分鐘數。

表 6：合成方塊 32、64、128、256、512、1024、2048、4096 時之平均步數與估計時間

	合成 32	合成 64	合成 128	合成 256	合成 512	合成 1024	合成 2048	合成 4096
only2	26.9/1.4	45.8/2.3	82.5/4.1	171/8.6	289/14.5	577/28.9	**1213.8/60.7**	2079/104.0
only4	13/0.7	26/1.3	50/2.5	83.4/4.2	162/8.1	295.4/14.8	**625.6/31.3**	1095.5/54.8
方塊 2	2^4=16	2^5=32	2^6=64	2^7=128	2^8=256	2^9=512	2^{10}=1024	2^{11}=2048
方塊 4	2^3=8	2^4=16	2^5=32	2^6=64	2^7=128	2^8=256	2^9=512	2^{10}=1024

我們將 only2、only4 合成 32、64、128、256、512、1024、2048、4096 的平均步數與方塊 2、方塊 4 的指數圖形繪製如圖 9。

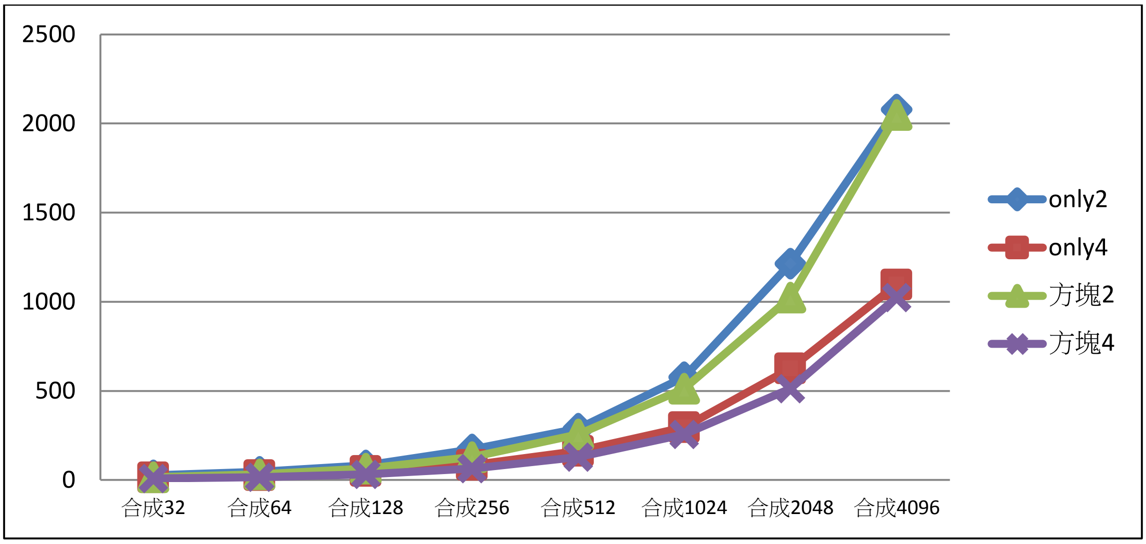

圖 9：only2、only4 與方塊 2、方塊 4 的指數圖形折線圖

從圖 9 可以看出 only2 合成 32、64、128、256、512、1024 的圖形皆在 2 的指數圖形折線上，而 only4 則在 2 的指數圖形折線下，明顯看出 only2 所花費步數最大，而 only4 則是最小。而 only2 所花費的步數幾乎是 only4 的 2 倍。若以 2048 來看的話，only2 約需 1200 步左右達到，與合成 2048 理論上需求 1024 個方塊 2 接近；而 only4 約需 600 步左右達到，與合成 2048 理論上需求 512 個方塊 4 接近。

接下來我們從數據挑選所有達 2048 的數據，將其所花費的步數並計算其平均整理於表 7。

表 7：only2、only4 合成 2048 的數據

2048	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	平均
only2	1199	1042	1035	1214	1039	1058	1131	1078	1126	1187	/	/	/	/	/	/	1110.9
only4	740	725	635	689	647	763	523	629	528	645	553	578	538	574	550	692	625.56

合成 2048 情況下，only2 平均步數為 1110.9，only4 平均步數為 625.56，從步數就能知道 only4 能運用比 only2 少的時間達成 2048，所以如果是方塊 2、4 比例的話，如果都是 only4 能夠花費時間少也能達到 2048。

而從表 7 也可明顯看出，only4 在 20 筆數據中，就有 16 筆數據達到 2048，而 only2 卻只有 10 次，代表 only4 非常容易達到 2048。

四、方塊 2 及方塊 4 出現比例對遊戲難易度之影響

只出現方塊 4 最容易 2048，只出現方塊 2 次之，前者花費時間較少；而方塊 2 及方塊 4 交錯出現之遊戲則不容易 2048。從 2048 程式實測中，2048 遊戲的方塊 2、方塊 4 之比例幾乎是 1:1，遊戲啟動時更少會出現兩個方塊 4，運用攻略可有較多機會合成方塊 512 或方塊 1024，但要進一步合成方塊 2048，則一定的難度。

2048 的遊戲規則很簡單，只使用加法，但是方塊 2 及方塊 4 交錯出現讓遊戲變得有點難度。在第三點中可以明顯看出，只出現方塊 2 或只出現方塊 4，兩者皆容易達到 2048，甚至能進一步達至 4096，其過程到最後進行機械式的滑動，容易令人厭煩，只是比耐力而已，相對來說方塊 2、4 比例大概 1:1 則較有挑戰性，需要謹慎思考每一步。

參●結論

看似簡單的 2048 程式，仍需要遊戲攻略才容易合成 2048。應用遊戲攻略「L 型法」、「垂直法」及「一行法」容易合成 512（約有一半的機會），但三者仍不容易 2048。若真要從三者挑選一種較佳遊戲攻略，本研究認為一行法是三種滑動方法最容易達成 2048 的，原因是從數據中看出達到 512 的比率最高，且滑動步數最少，耗時也短，容易達成也不減玩家興致。

若以滑動一次需時 3 秒來估計，方塊 2 與方塊 4 交錯出現的狀況下，三種攻略要合成方塊 32 約需 1 分鐘、合成方塊 64 約需 2 分鐘、合成方塊 128 約需 3 分鐘、合成方塊 256 約需 6 分鐘、合成方塊 512 約需 10 分鐘、合成方塊 1024 約需 20 分鐘，若要達 2048 則初步估計約需 30 分鐘，隨著數字越來越大，開始考驗著玩家耐力。

當遊戲設定只出現方塊 2（only 2）或只出現方塊 4（only 4）時，以「一行法」遊戲攻略進行遊戲，合成 2048 的比例過半，甚至可以達成 4096，兩者只差在滑動步數，only4 滑動步數與估計耗時均較 only 2。在 only 2 及 only 4 遊戲過程中也發現，當掌握或熟悉遊戲攻略後，2048 變成是耐力大於智力的考驗。

綜合言之，若 2048 只出現方塊 2 或只出現方塊 4，非常容易致勝（達成 2048），也就是遊戲變簡單了，而原本設定的版本是方塊 2 與方塊 4 交錯出現，讓遊戲變得難度而挑起玩家挑戰的慾望。

肆●引述資料

[1]小昭（2014）。2048 過關技巧心得攻略。2014/11/1 日，取自 http://shouyou.com.tw/how/2014/0605/185323.html#.VGIzYIfffKU

[2]高淑珍（譯）(2003)。圖解數學基礎入門。台北縣：世茂出版有限公司。

[3]許馨云、孫瑋彤、翁子恒（2014）。停不下來的 2048。2014/11/11，取自 http://163.25.121.156/registration/103/score（第 54 屆金門地區中小學科學展覽會）。

[4]星詠（2014）。數字和空間的無窮魅力——停不下來的 2048。2014/11/10，取自 http://www.taiwanfansclub.com/article-171989-1.html

[5]維基百科（2014）。2048。2014/10/26，取自 http://zh.wikipedia.org/wiki/2048

[6]旗立研究室（2012）。Visual Basic 2010 程式語言。台北市：旗立資訊股份有限公司。