2011 年臺灣國際科學展覽會

優勝作品專輯

編號：010030

作品名稱

缺一格也可以—骨牌排列順序數量之探討

得獎獎項

二等獎

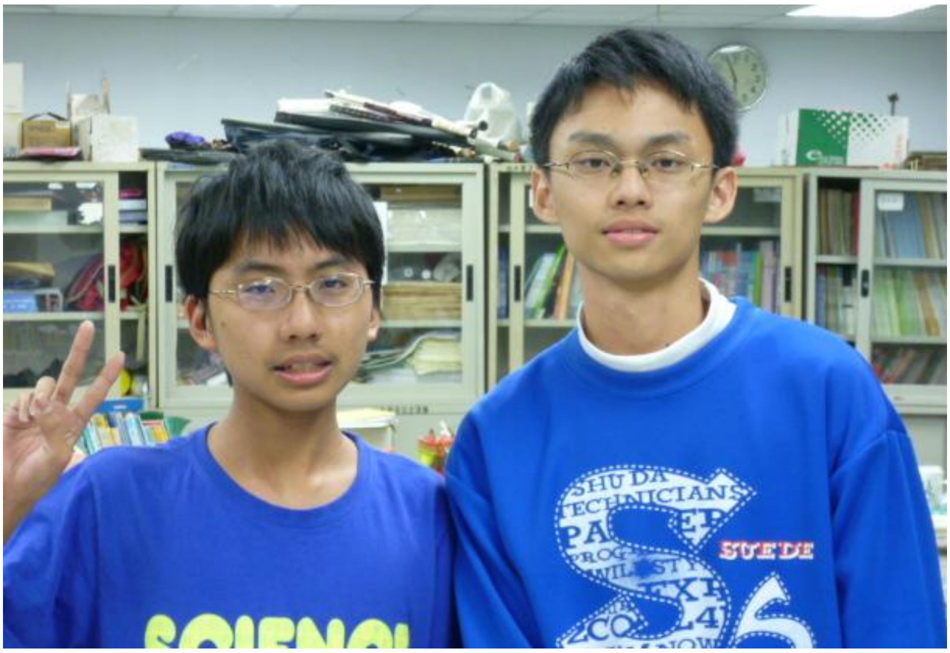
作者姓名：李宜霖、程鼎元

就讀學校：高雄市立高雄高級中學

指導教師：鍾玉才、蔡哲淵

關鍵字：**Domino Tiling、Young Tableau、Motzkin Path**

作者簡介



　　我是李宜霖(右)，來自高雄市，目前就讀高雄中學二年級。我喜歡閱讀科普書籍、科幻小說、打桌球，並思考一些數學問題。同時，這也是引起我作數學科展的最大動機，就一頭栽進這個世界中。我的「科展生涯」已經邁入第六年了，國內外的好作品，齊聚在此相互交流，我也期望能在這個難得的機會中盡全力，享受做科展的樂趣，學到更多東西。

　　　我是程鼎元(左)，就讀高雄中學二年級。平時除了做科展外，也很喜歡寫程式，思考與解決許多有趣的問題，有許多參加競賽的經驗。雖然研究過程非常辛苦，但是當發現問題的性質，進而找到巧妙而簡易的解決方法，是多麼令人喜悅又開心的事。而能夠與各地的好手進行學術交流，讓我感到十分興奮，希望能在這次的活動中讓自己認識更多朋友，也藉由這個難得的機會學習到更多平常學不到的東西。

摘要

在一次偶然的機會中，我們在科學月刊上讀到一篇探討骨牌排列 (Domino Tiling) 問題的文章。文中骨牌排列問題看似單純，但其中卻有許多有趣的性質，因此，我們想對此進行更深入的研究。

我們將1×2的骨牌填入特別的方格中，考慮填入的順序並計算其排列情形的數量。由定義的兩個模型 *SDT* (Standard Dominos Tableau) 與「缺一格」的 *GDT* (Good Dominos Tableau) 為基礎，嘗試尋找它們之間的關連性，藉由對應(bijection)的方法，得知在高度的限制為奇數的情況下，*SDT* 與 *GDT* 的數量相同，並且，兩者之間的結構有密切關聯。

除此之外，在這個研究中，針對 *SDT* 在高度不超過三排時，我們建構出了骨牌排列情形與我們架構的「廣義 Motkzin 路徑」的對應操作流程。接著，我們嘗試探討一般化的 *SDT* 與此結果的關聯性，由對應過程的想法，架構出高維度的「廣義 Motkzin 路徑」，並推測此路徑與 *SDT* 之間有緊密的相關性！

Abstract

Once, we read an article about Domino Tiling, which is the motivation of this project, on Science Monthly. The problem seemed to be simple initially; yet, in fact, there are many interesting characteristics. Hence, we started our project with this problem.

This project discusses the situations in which uses 1*2 dominos to fill into the lattices with specific rules. Based on two models, "*SDT* (Standard Domino Tableau)" and "*GDT* (Good Domino Tableau)", trying to find out their corelations. By limiting the height of SDT and GDT to odd numbers; the total number of SDT is the same as GDT with the same number of dominos. We introduce the bijection in our research.

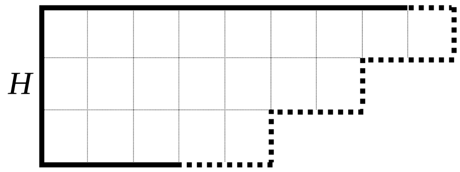
Moreover, this project considers all the conditions of SDT, when the height of SDT is not greater than 3, the bijection between SDT and "Extended Motzkin Path" is established! Then, we increase the height of SDT. On seeing the previous bijections established, we thus construct "higher Extended Motkzin Paths", claiming they will closely related to SDTs and GDTs with greater heights in this project.

壹、研究動機

我們在 2009 年十月份的科學月刊中讀到一篇關於骨牌排列(Domino Tiling)的文章[2]，文中提到了一道數學問題：利用許多1×2的骨牌拼滿$2\times n$的矩形、$3\times n$的矩形(其中$n\in\mathbb{N}$)，其方法數為何？並將問題更加一般化，擴展到利用1×2的骨牌拼滿$2m\times 2n\ \ (m,n\in\mathbb{N})$的矩形之方法數。而此問題於 1961 年已經被解決[4]，其方法數如下：

$$4^{mn}\prod_{j=1}^{m}\prod_{k=1}^{n}\left(\cos^2\frac{j\pi}{2m+1}+\cos^2\frac{k\pi}{2n+1}\right)$$

由於我們對骨牌在平面上的排列情形非常感興趣，因此提出了以下問題作為本次研究的主要方向：



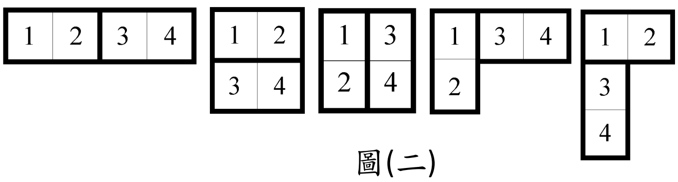
圖(一)

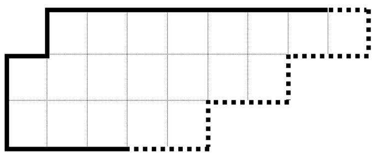
將n張骨牌填入平面中高度為$H, H\le 3$，長度不限的格子中，如圖(一)所示，但不一定要完全填滿，其填入的規則為以下 2 點：

> 1. 水平或垂直放置每張骨牌，依$(1,2),(3,4),\cdots,(2n-1,2n)$的順序在骨牌填入時標示其編號。依序填入的數字大小必須由左而右遞增，由上而下遞增。
>
> 2. 放置每塊骨牌時，均滿足排列圖形的每列的長度由上而下不得小於其下一列的長度；每列的長度由左而右不得小於其下一行的長度。且每列，每行均連續放置。

依其規則討論排列骨牌的各種情形，並計算其方法數。

以 2 張骨牌的 5 種排列方式為例，其各種排列方式如圖(二)所示。


圖(二)


圖(三)

此外，若將矩形的形狀改為如圖(三)所示，探討其骨牌排列的數量為何？

貳、研究目的

一、探討集合 SDT 與「缺一格」的 GDT 數量之間的關係

二、探討集合 SDT 在高度小於等於三的數量以及與 Motzkin Path 之間的關係

參、研究器材及設備

一、電腦

二、紙筆

三、方格紙

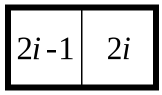
四、mingw32 – G++ Compiler 3.4.2

肆、名詞與定義

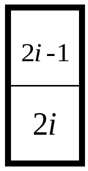
一、名詞解釋

(一) 骨牌：

邊長為1×2單位的矩形，其上標示兩相鄰整數，即$(1,2),(3,4),(5,6),\cdots,(2i-1,2i)$，可為水平放置，如圖(四)；或直立放置，如圖(五)。



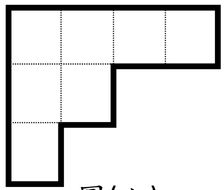
圖(四)



圖(五)

(二) Young Tableau：

一種整數分拆的表達方式，n為正整數，將n分割成k個正整數的總和，將它記為$k_1,k_2,\cdots,k_H,(k_1\ge k_2\ge k_3\ge\cdots\ge k_H),\sum_{i=1}^{H}k_i=n$；第一列由左至右連續放置$k_1$個正方形格子，第二列由左至右連續放置$k_2$個正方形格子，依此類推，第$H$列由左至右連續放置$k_H$個正方形格子，由以上規則所組成的圖形。

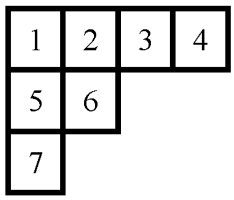


圖(六)

如圖(六)所示，考慮正整數$7=4+2+1,k_1=4,k_2=2,k_3=1$。

(三) SYT (Standard Young Tableau)：

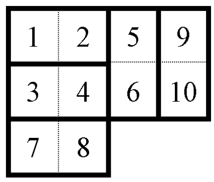
在 Young Tableau 的圖形中填入正整數$1,2,\cdots,n$，使得每行的數字依序往下遞增，每列的數字依序往右遞增，如圖(七)。



圖(七)

(四) SDT (Standard Dominos Tableau)：

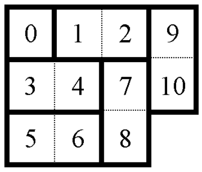
在 Young Tableau 的圖形中填入正整數$1,2,\cdots,2n$，但規定數字$1,2$必須相鄰、$3,4$必須相鄰、...、$2n-1,2n$必須相鄰；將填入數字$1,2$的兩格視為一張骨牌，$3,4$視為一張骨牌，依此類推，$2n-1,2n$視為一張骨牌，使得每行的數字依序向下遞增，每列的數字依序向右遞增，可視為 Standard Young Tableau 的特例，如圖(八)。



圖(八)

(五) GDT (Good Dominos Tableau)：

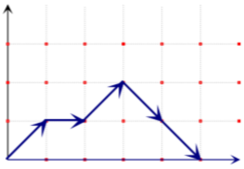
在 Young Tableau 的圖形中填入 0 及正整數 $1,2,\cdots,2n$，但規定數字 $1,2$ 必須相鄰、$3,4$ 必須相鄰、...、$2n-1,2n$ 必須相鄰；將 0 視為獨立的固定方格(即視為空格)，而填入數字 $1,2$ 的兩格視為一張骨牌，$3,4$ 視為一張骨牌，依此類推，$2n-1,2n$ 視為一張骨牌，使得每行的數字依序往下遞增，每列的數字依序往右遞增，可視為 Standard Dominos Tableau 的變形，如圖(九)。



圖(九)

(六) Motzkin Path：

從坐標原點 $(0,0)$ 走到 $(n,0)$，規定走法為依循向量 $(1,0)$、$(1,1)$ 或 $(1,-1)$，且不得通過 x 軸下方所形成的路徑，如圖(十)。


圖(十)

(七) Motzkin Number：

從坐標原點 $(0,0)$ 走到 $(n,0), n\in\mathbb{N}$，規定走法為依循向量 $(1,0)$、$(1,1)$ 或 $(1,-1)$，且不得通過 x 軸下方的方法數，前 10 項數量為：1, 2, 4, 9, 21, 51, 127, 323, 835, 2188, …

(八) Motzkin Triangle：

從坐標原點 $(0,0)$ 出發，規定走法為依循向量 $(1,0)$、$(1,1)$ 或 $(1,-1)$，且不得通過 x 軸下方，走到 (n,k) 的方法數。

(九) 廣義 Motzkin 路徑：

從坐標原點 $(0,0)$ 出發，規定走法為依循向量 $(1,0)$、$(1,1)$ 或 $(1,-1)$，且不得通過 x 軸下方，走到 (n,k) 所形成的路徑。

(十) Pascal Triangle：

由二項式係數組合成的三角形陣列，其數量等價於：從坐標原點 $(0,0)$ 出發，規定走法為依循向量 $(1,0)$ 或 $(0,1)$，走到任意點 (m,n) 的方法數。

(十一) 一路領先：

將物品依特定順序分別發給 n 個人，且在過程中符合：第一個人所得物品數≥第二個人所得物品數≥......≥第 n 個人所得物品數。

二、符號定義

(一) $S_H(n,k)$：由 n 張骨牌組成的 SDT 所形成之集合，其中共有 k 張骨牌直立放置，且其高度為 H，並將其數量記為 $|S_H(n,k)|$。($n,k \in \mathbb{N}, k \le n, H > 0$)

(二) $G_H(n,k)$：由 n 張骨牌組成的 GDT 所形成之集合，其中共有 k 張骨牌直立放置，且其高度為 H，並將其數量記為 $|G_H(n,k)|$。

($n,k \in \mathbb{N}, k \le n, H > 0$)

(三) $S_H(n)$：由 n 張骨牌組成的 SDT 所形成之集合，其高度為 H，將數量記為 $|S_H(n)|$。

(四) $G_H(n)$：由 n 張骨牌組成的 GDT 所形成之集合，其高度為 H，將數量記為 $|G_H(n)|$。

(五) W_H：SDT 或 GDT 的第 H 列的長度。

(六) $M(n,k)$：從坐標原點 $(0,0)$ 出發，規定走法為依循向量 $(1,0)$、$(1,1)$ 或 $(1,-1)$，且不得通過 x 軸下方，走到 (n,k) 的方法數。$M(n,k)$ 表示其遞迴關係式：

$$M(0,0) = M(1,0) = M(0,1) = 1, M(n,0) = 1, n \ge 2$$
$$\begin{cases} M(n,k) = M(n-1,k-1) + M(n-1,k) + M(n-1,k+1), \forall k \in \mathbb{N}, k < n \\ M(n,n) = M(n-1,n-1) + M(n-1,n) \end{cases}$$

(七) D_{ij}：表示在 SDT 或是 GDT 中，該塊骨牌出現的位置，並定義第一列為 a、第二列為 b、第三列為 c...，依此類推。例如：D_{aa} 表示該塊骨牌為水平，且出現在第一列、D_{ab} 表示該塊骨牌為直立，且橫跨第一列與第二列。

伍、研究內容

一、探討集合 SDT 與「缺一格」的 GDT 數量之間的關係

(一) 探討 $S_{\le 3}(n)$ 與 $G_{\le 3}(n)$ 數量之間的關係及其對應

將 Tableau 的形狀變化為如圖(十一)所示，為方便說明，以#表示空格(即編號 0 的方格)，稱此為 GDT 。以下討論高度分別為一及二時的情形：

0	1	2	.....
3	5	.....	
4	6	.....	

圖(十一)

1. 考慮任意 $G_1(n,k), k=0$ ，唯一的排列方式為：

#	1	2	3	4	5	6	7	8	. . . . . .	2n-1	2n

圖(十二)

而 $S_1(n,k), k=0$ ，唯一的排列方式為：

1	2	3	4	5	6	7	8	. . . . . .	2n-1	2n

圖(十三)

因此，此 $\left|G_1(n)\right| = \left|S_1(n)\right| = 1$ 。

2. (1). 考慮任意 $G_{\le 2}(n,0)$ 與 $S_{\le 2}(n,0)$ ，此數量可視為兩人「一路領先」的方法數，即為 SYT 高度小於等於二的方法數。若將 $G_{\le 2}(n,0)$ 的第一排所有骨牌向左移一格，第二排之排列狀況不變，由於 $W_1 > W_2, W_1 - 1 \ge W_2$，仍符合大小關係，且其數量與 $S_{\le 2}(n,0)$ 相同，即 $\left|G_{\le 2}(n,0)\right| = \left|S_{\le 2}(n,0)\right| = C^{n}_{\left[\frac{n}{2}\right]}$ 。

將第一排所有骨牌向左移一格，方法數與其相同：

#	1	2	3	4	5	6	7	8	……………	i	j
………	………	………	………	…………	…………						

圖(十四)

1	2	3	4	5	6	7	8	……………	i	j
………	………	………	………	……………	………					

圖(十五)

(2). 不需考慮任意$G_{\le 2}(n,k), k>0$，因為該圖形不存在。

說明：對於任意$G_{\le 2}(n,k), k>0$，其W_1必為奇數，W_2必為偶數，無法在此GDT中加入一塊直立的骨牌D_{ab}，故圖形不存在，不需考慮此情況。

3. 考慮$G_{\le 3}(n)$與$S_{\le 3}(n)$數量之間的關係：

對於$G_H(n)$中的任一空格#，考慮移動該空格#之上、下、左、右的格子，與該空格之上、下、左、右的邊相鄰的骨牌分別稱為：上骨牌(U)、下骨牌(D)、左骨牌(L)、右骨牌(R)。對於所有的上、下、左、右骨牌，在步驟中定義該塊骨牌中緊鄰空格的一半為 A 部分。而該骨牌不緊鄰的另一半為 B 部分，並稱$U_A, U_B, D_A, D_B, L_A, L_B, R_A, R_B$為此次步驟的「相關方格」。以圖(十六)為例，其「相關方格」之代號如下：

L_B	U_A	U_B	
L_A	#	R_A	R_B
	D_A	D_B	

圖(十六)

(1). 列舉出$n \le 10$且高度小於等於三時，GDT與SDT的數量，發現其數量是相等的，因此我們給出一個對應方法，證明：

$\left|G_{\le 3}(n)\right| = \left|S_{\le 3}(n)\right|$ 。

(2). 對應規則及流程說明：

$\boxed{\text{(a). } G_{\le 3}(n) \Rightarrow S_{\le 3}(n)}$ ：

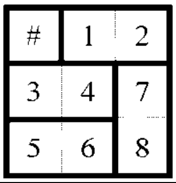
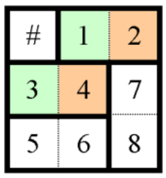
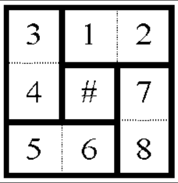
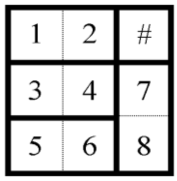
(I). 考慮與空格四邊相鄰的四張骨牌，分別將其不與空格相鄰的部份移入空格。

(II). 考慮進行操作後同一張骨牌內的兩個正整數，將該骨牌之上或左部分設為該骨牌原先兩正整數中較小者，骨牌之下或右部分設為該骨牌原先兩正整數中較大者。

(III). 這四張骨牌中，除了「回溯」外，僅有一塊在移動後仍不違反 Standard Dominos Tableau 由左至右，由上至下之遞增規則。

(以藍色表示與空格相鄰的格子；粉色表示欲移入的格子)

表(一) GDT , SDT 對應示意圖

舉例	1	2	3
			
原始圖形	分別考慮右、下骨牌可將整數 2 或 4 移入	對下骨牌進行操作 再進行操作(II) 不合遞增規則	對右骨牌進行操作 再進行操作(II) 符合遞增規則

$\boxed{\text{(b). } S_{\le 3}(n) \Rightarrow G_{\le 3}(n)}$ ：

(I). 先將填入數字 1 的小格搬離，並在與數字 2 小格相鄰的四周找尋格子作為數字 1 可能的目的地，此時 SDT 的左上角出現一個空格，並繼續進行步驟 (II).。

(II). 將在上次動作中作為目的地的位置之原先的格子搬離，並將上次動作被搬離的格子填入此處。再找尋與此步搬離之

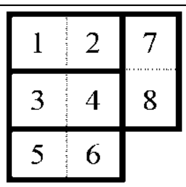
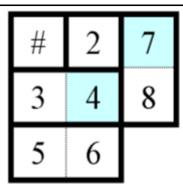
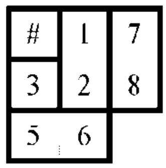
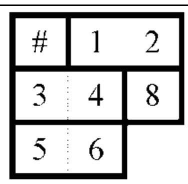
骨牌另一半格子相鄰四周的格子作為本次動作可能的目的地，依此類推。

(III). 考慮每次進行步驟 (II).後同一張骨牌內的兩個正整數，將該骨牌之上或左部分設為該骨牌原先兩正整數中較小者，骨牌之下或右部分設為該骨牌原先兩正整數中較大者。

(IV). 該骨牌四周，除了「回溯」外，僅有一塊可以作為目的地，使得操作後仍不違反 Standard Dominos Tableau 由左至右，由上至下之遞增規則。

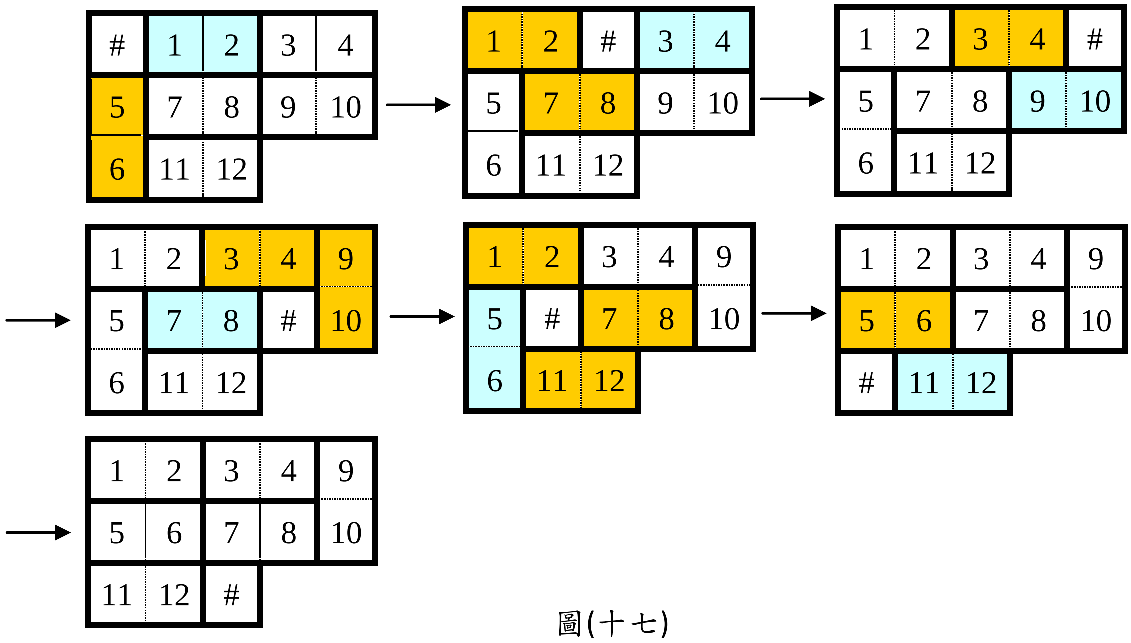
(以淺藍色表示可作為目的地的方格)

表(二) SDT ,GDT 對應示意圖

舉例	1	2	3
			
原始圖形	將 1 搬離 找尋目的地	對下骨牌進行操作 再進行操作(II) 不合遞增規則	對右骨牌進行操作 再進行操作(II) 符合遞增規則

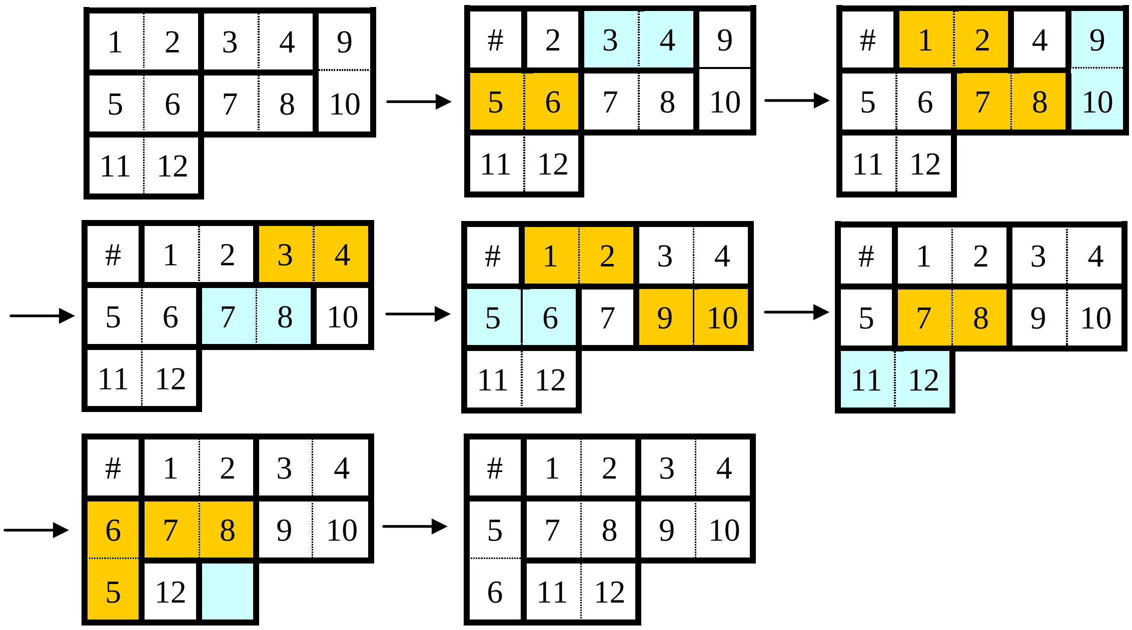
(c). 以一個六張骨牌的對應操作為例：(褐色部分表示不可移動的骨牌；淺藍色部分表示可被移動的骨牌)

(I). $G_{\leq 3}(n) \Rightarrow S_{\leq 3}(n)$：



圖(十七)

(II). $S_{\leq 3}(n) \Rightarrow G_{\leq 3}(n)$：



圖(十八)

(3). 證明 $G_{\le 3}(n)$ 與 $S_{\le 3}(n)$ 的對應

無論在何種情形下，均恰有一塊骨牌之移動為「合法操作」，並且證明此操作可逆，即 $G_{\le 3}(n) \xrightleftharpoons{bijection} S_{\le 3}(n)$ 。

證明一： $G_{\le 3}(n) \Rightarrow S_{\le 3}(n)$ 的操作唯一

將空格#在 GDT 中的位置分成幾種情況討論，為簡化其過程，討論並列出以下可能的情況：

(a). 該空格位於該 Tableau 之左上角，此狀態為 GDT 的原始狀態，分別考慮將 R_B 、 D_B 移入空格#中，將其分為下列兩種情形：

表(三)

圖形	# R_A D_A	R_B R_A D_A	D_B R_A D_A
操作	原始狀態	操作 1-1	操作 1-2
情形 1-1	$R_A > D_A$	不合法	合法
情形 1-2	$D_A > R_A$	合法	不合法

說明：考慮以下各種情形：

(I). 若原始狀態符合情形 1-1，則 $R_B > D_A$ ， $R_A > D_B$ ，即操作 1-1 不合法，操作 1-2 合法。

(II). 若原始狀態符合情形 1-2，則 $D_A > R_B$ ， $D_B > R_A$ ，即操作 1-1 合法，操作 1-2 不合法。

(III). 若骨牌 R 或骨牌 D 其中一塊不存在，則另外一塊必可合法移動。

(b). 該空格位於該 Tableau 之最右下角，即骨牌 R 、 D 不存在，此情形對應結束，所成的圖形為與 GDT 所對應之 SDT 。同理可

證，該空格周圍的骨牌L、D不存在以及骨牌U 、R不存在的情形皆只有一步「合法操作」。

(c). 空格周圍的骨牌U 、D不存在。

表(四)

圖形	L_A # R_A	L_A L_B R_A	L_A R_B R_A
操作	原始狀態	操作 3-1	操作 3-2
情形 3-1	$L_A < R_A$	合法	合法

說明：原始狀態符合情形 3-1，恰有兩步操作合法，因此有兩種「合法操作」，可知其中一步為「回溯」。同理可證，該空格周圍的骨牌L、R不存在時的情形，皆有兩種「合法操作」，但可知其中一步為「回溯」。

(d). 空格周圍的骨牌U 不存在，即該空格位於 Tableau 的上方，已知$L_A < D_A$ 。

表(五)

圖形	L_A # R_A D_A	L_A L_B R_A D_A	L_A D_B R_A D_A	L_A R_B R_A D_A
操作	原始狀態	操作 4-1	操作 4-2	操作 4-3
情形 4-1	$D_A < L_A < R_A$			
情形 4-2	$L_A < D_A < R_A$	合法	合法	不合法
情形 4-3	$L_A < R_A < D_A$	合法	不合法	合法

說明：考慮以下各種情形：

(I). 若原始狀態符合情形 4-1，不存在此情形，因為與已知 $L_A < D_A$ 矛盾。

(II). 若原始狀態符合情形 4-2

(i). 對於操作 4-1，因為 $L_B < R_A, L_B < D_A$，此操作合法。

(ii). 對於操作 4-2，因為 $L_A < D_B < R_A$，此操作合法。

(iii). 對於操作 4-3，因為 $R_B < R_A, R_B < D_A$，此操作不合法。

因此有兩種「合法操作」，但可知其中一步必為「回溯」。

(III). 若原始狀態符合情形 4-3

(i). 對於操作 4-1，因為 $L_B < R_A, L_B < D_A$，此操作合法。

(ii). 對於操作 4-2，因為 $L_A < D_B < R_A$，此操作不合法。

(iii). 對於操作 4-3，因為 $R_B < R_A, R_B < D_A$，此操作合法。

因此有兩種「合法操作」，但可知其中一步必為「回溯」。

同理可證，當該空格周圍的骨牌 L 不存在、骨牌 D 不存在以及骨牌 R 不存在時的情形，皆有兩種「合法操作」，但可知其中一步必為「回溯」。

(e). 當空格位於 Tableau 的中間時，即其上、下、左、右均有骨牌的一般情形。

表(六)

圖形	$\begin{array}{\|c\|c\|c\|}\hline & U_A & \\ \hline L_A & \# & R_A \\ \hline & D_A & \\ \hline\end{array}$	$\begin{array}{\|c\|c\|c\|}\hline & U_A & \\ \hline L_A & R_B & R_A \\ \hline & D_A & \\ \hline\end{array}$	$\begin{array}{\|c\|c\|c\|}\hline & U_A & \\ \hline L_A & U_B & R_A \\ \hline & D_A & \\ \hline\end{array}$
操作	原始狀態	操作 5-1	操作 5-2
情形 5-1	$U_A < L_A < R_A < D_A$	不合法	不合法
情形 5-2	$U_A < L_A < D_A < R_A$	不合法	合法
情形 5-3	$L_A < U_A < D_A < R_A$	合法	合法
情形 5-4	$L_A < U_A < R_A < D_A$	合法	不合法

圖形	$\begin{array}{\|c\|c\|c\|}\hline & U_A & \\ \hline L_A & \# & R_A \\ \hline & D_A & \\ \hline\end{array}$	$\begin{array}{\|c\|c\|c\|}\hline & U_A & \\ \hline L_A & L_B & R_A \\ \hline & D_A & \\ \hline\end{array}$	$\begin{array}{\|c\|c\|c\|}\hline & U_A & \\ \hline L_A & D_B & R_A \\ \hline & D_A & \\ \hline\end{array}$
操作	原始狀態	操作 5-3	操作 5-4
情形 5-1	$U_A < L_A < R_A < D_A$	合法	合法
情形 5-2	$U_A < L_A < D_A < R_A$	合法	不合法
情形 5-3	$L_A < U_A < D_A < R_A$	不合法	不合法
情形 5-4	$L_A < U_A < R_A < D_A$	不合法	合法

說明：原先為一個合法的 Tableau，即 $U_A < T < D_A, L_A < T < R_A$，由遞移律得知：$L_A < D_A, U_A < R_A$。

故可將 U_A, D_A, R_A, L_A 四者之大小關係，分為以下四種：

(I). $U_A < L_A < R_A < D_A$：操作 5-1 及 5-2 不合法；操作 5-3 及

5-4 合法，但其中一步為「回溯」。

(II). $U_A < L_A < D_A < R_A$：操作 5-1 及 5-4 不合法；操作 5-2 及 5-3 合法，但其中一步為「回溯」。

(III). $L_A < U_A < D_A < R_A$：操作 5-3 及 5-4 不合法；操作 5-1 及 5-2 合法，但其中一步為「回溯」。

(IV). $L_A < U_A < R_A < D_A$：操作 5-2 及 5-3 不合法；操作 5-1 及 5-4 合法，但其中一步為「回溯」。

無論為何種情形，均恰有兩種可能的合法移動過程，但其中一步為「回溯」。

(f). 討論所有骨牌相對位置，得知以下結果：

(I). 考慮空格位於最左上角，即空格之上方或左方並沒有任何方格存在，則此為 *GDT* 的初始狀態，恰有一步為「合法操作」。

(II). 考慮空格位於最右下角，即空格之下方或右方並沒有任何方格存在，則此為終止狀態，對應已經完成。

(III). 考慮其他狀況，討論各種方格分佈之可能，證明在任意狀態中，恰有兩步操作為「合法操作」，但有一步操作會回到上一狀態，因此不考慮此情形。故對於任一狀態，其「合法操作」是唯一的。

證明二：$S_{\le 3}(n,k) \Rightarrow G_{\le 3}(n,k)$ 的操作唯一

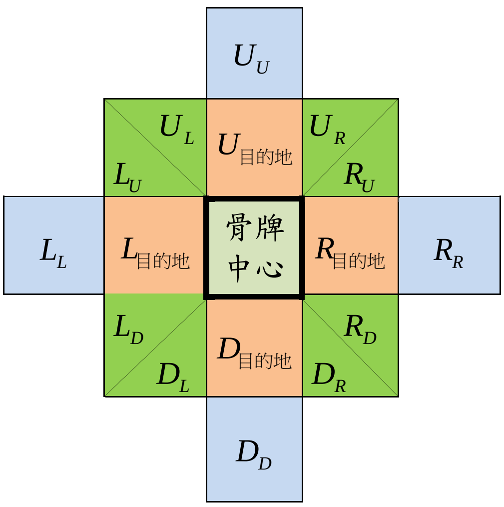
考慮移動目的地方格四周的骨牌，可能被移動操作而影響大小關係，由於必須符合每行，每列的遞增性質，要討論周圍與其接觸的骨牌，並且仍符合 Young Tableau 規則。因此，對骨牌周圍的排列情形的相對位置進行討論，藉此證明每一步操作均合法且唯一，最後形成與 *SDT* 相對應的圖形。

(a). 名詞符號定義：

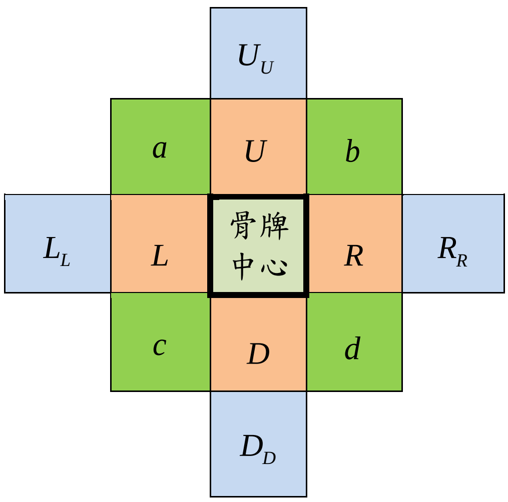
(I). 骨牌中心：一塊骨牌中與欲被搬離的格子相鄰但不移動的方格。

(II). 目的地方格：「骨牌中心」周圍上、下、左、右四塊方格，分別為$U_{目的地}$、$L_{目的地}$、$D_{目的地}$、$R_{目的地}$，而與此四塊方格周圍上、下、左、右四塊方格再分別稱為U_U、L_L、R_R、D_D、U_R、R_U、U_L、L_U、L_D、D_L、D_R、R_D、R_U、U_R，如圖(十九)。

將方格U_L、L_U稱為方格a；方格U_R、R_U稱為方格b；方格L_D、D_L稱為方格c；方格R_D、D_R稱為方格d，如圖(二十)，搬離後的數字為x。對於任何情況，將操作方式分為以下四種，再逐一分類討論移動的合法性。



圖(十九)



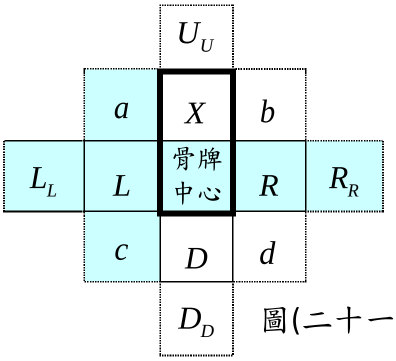
圖(二十)

(b). 各種操作情形：

(I). 操作一：移入的格子為$U_{目的地}$，其中以$X_{\min}, X_{\max}$表示移入後該骨牌上的數字。如圖(二十一)，須滿足大小關係：

(i). $U_U < X_{\min} < X_{\max} < D < D_D$

(ii). $a < X < b$

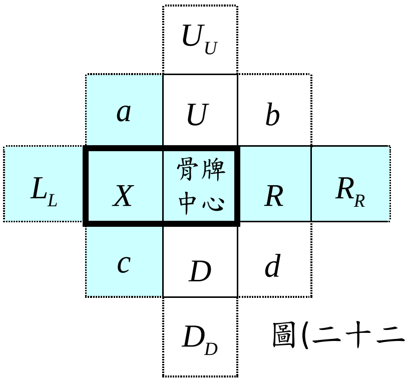


圖(二十一)

(II). 操作二：移入的格子為$L_{目的地}$，其中以$X_{\min}, X_{\max}$表示移入後該骨牌上的數字。如圖(二十二)，須滿足大小關係：

(i). $L_L < L < X_{\min} < X_{\max} < R_R$

(ii). $a < X < c$

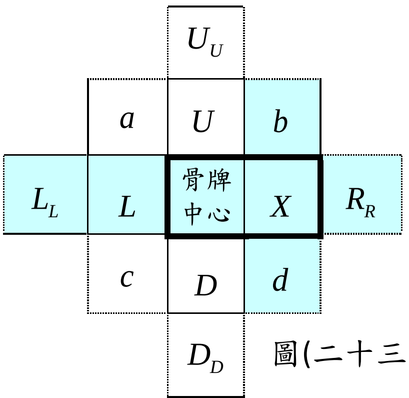


圖(二十二)

(III). 操作三：移入的格子為$R_{目的地}$，其中以$X_{\min}, X_{\max}$表示移入後該骨牌上的數字。如圖(二十三)，須滿足大小關係：

(i). $L_L < L < X_{\min} < X_{\max} < R_R$

(ii). $b < X < d$

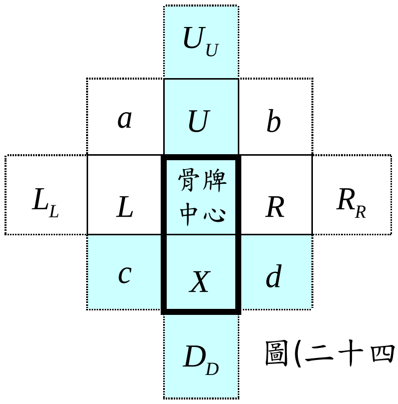


圖(二十三)

(IV). 操作四：移入的格子為$D_{目的地}$，其中以$X_{\min}, X_{\max}$表示移入後該骨牌上的數字。如圖(二十四)，須滿足大小關係：

(i). $U_U < U < X_{\min} < X_{\max} < D_D$

(ii). $c < X < d$

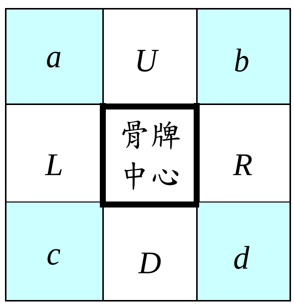


圖(二十四)

由上述各種操作方式得知，無論其操作動作為何，只要原狀態合法，其各操作過程之(i). 項條件不須考慮，因此，我們只列出可能影響其操作合法性的格子，分別討論。(c). 比較「骨牌中心」四周的八個方格以及被搬移的格子$X, (X_{\min}, X_{\max})$，　共有九類情形，將其列出如下，一一討論。

(I). 情形一：

探討a, b, c, d以及被搬離的格子X共六種大小順序關係。在此情形中d, D, R不影響移動的關係。分別討論發現以上四種可能的操作方式只有兩種符合Young Tableau規則，但必有一種為「回溯」操作，



圖(二十五)

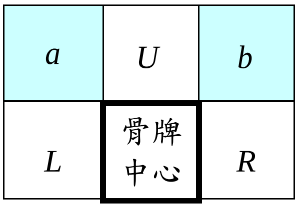
列為下表(七)。

表(七)

順序關係	操作一	操作二	操作三	操作四
1. $a<X_{\min}<X_{\max}<b<c<d$	合法	合法	不合法	不合法
2. $a<b<X_{\min}<X_{\max}<c<d$	不合法	合法	合法	不合法
3. $a<b<c<X_{\min}<X_{\max}<d$	不合法	不合法	合法	合法
4. $a<c<b<X_{\min}<X_{\max}<d$	不合法	不合法	合法	合法
5. $a<c<X_{\min}<X_{\max}<b<d$	合法	不合法	不合法	合法
6. $a<X_{\min}<X_{\max}<c<b<d$	合法	合法	不合法	不合法

(II). 情形二：

探討a,b以及被搬離的格子X共兩種大小順序關係，其下方無格子，但不可以作為目的地，在此情形中L,R不影響移動的關係。



圖(二十六)

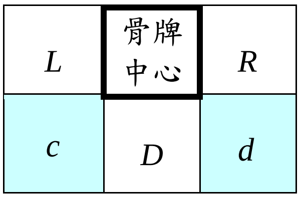
仿照情形一的方式分別討論四種可能的操作方式，但只有兩種符合 Young Tableau 規則，且其中一種必為「回溯」。

表(八)

順序關係	操作一	操作二	操作三	操作四
1. $a<X_{\min}<X_{\max}<b$	合法	合法	不合法	
2. $a<b<X_{\min}<X_{\max}$	不合法	合法	合法	

(III). 情形三：

探討c,d以及被搬離的格子X兩種大小順序關係，其上方無格子且為SDT的邊界，不可為目的地。仿照相同方式，分別討論可能的操作方式，但只有兩種符合 Young Tableau 規則，



圖(二十七)

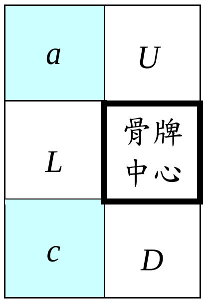
且其中一種必為「回溯」。

表(九)

順序關係	操作一	操作二	操作三	操作四
1. $c<X_{\min}<X_{\max}<d$		不合法	合法	合法
2. $X_{\min}<X_{\max}<c<d$		合法	合法	不合法

(IV). 情形四：

探討 a,c 以及被搬離的格子 X 共兩種大小順序關係，其右方無格子，但不可作為目的地，在此情形中 D 不影響移動的關係。



圖(二十八)

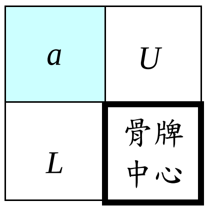
仿照相同方式，分別討論，發現可能的操作方式只有兩種符合 Young Tableau 規則，但其中一種必為「回溯」。

表(十)

順序關係	操作一	操作二	操作三	操作四
1. $a<X_{\min}<X_{\max}<c$	合法	合法		不合法
2. $X_{\min}<X_{\max}<a<c$	不合法	合法		合法

(V). 情形五：

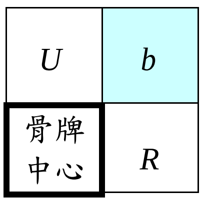
因為 $a<X_{\min}<X_{\max}$，且其右、下方無格子，表示對應操作結束。



圖(二十九)

(VI). 情形六：

其左方不可作為目的地，但下方可以，在此情形中 R 不影響移動的關係。



圖(三十)

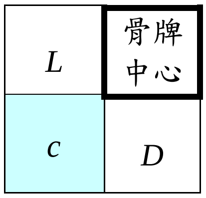
仿照前述的方式，發現以上四種可能的操作方式只有兩種符合 Young Tableau 規則，但其中一種必為「回溯」。

表(十一)

順序關係	操作一	操作二	操作三	操作四
1. $X_{\min} < X_{\max} < b$	合法		合法	不合法
2. $b < X_{\min} < X_{\max}$	不合法		合法	合法

(VII). 情形七：

其上方不可作為目的地，但右方可以，在此情形中 D 不影響移動的關係。仿照前述的方式，發現以上四種可能的操作方式只有兩種符合 Young Tableau 規則，但其中一種必為「回溯」。



圖(三十一)

表(十二)

順序關係	操作一	操作二	操作三	操作四
1. $X_{\min} < X_{\max} < c$		合法	合法	不合法
2. $c < X_{\min} < X_{\max}$		不合法	合法	合法

(VIII). 情形八：

其左、右方不可作為目的地，只有兩種符合 Young Tableau 規則，但其中一種必為「回溯」。

圖(三十二)

(IX). 情形九：

其左、右方不可作為目的地，只有兩種符合 Young Tableau 規則，但其中一種必為「回溯」。

圖(三十三)

(d). 以上的大小順序關係，分別討論這八種情形、四種可能的操作方式，其中只有兩種符合 Young Tableau 規則，但其中一種必為「回溯」。 同時證明了 SDT 對應到 GDT 的操作過程唯一。

4. 考慮 $G_{\le H}(n)$ 與 $S_{\le H}(n)$, $H>3$ 時，數量之間的關係：

當高度只要限制在小於等於奇數的情形下，SDT 以及「缺一格」的 GDT 數量會**相等**。意即：$|S_{\le H}(n)| = |G_{\le H}(n)|, \forall H = 2i-1, \mathrm{i} \in \mathbb{N}$。藉由以上的對應方式可證明此現象。

關於 $G_{\le H}(n)$ 以及 $S_{\le H}(n)$ 之間的關係，列出如下：

$$SDT_{\le H}(n,k)\begin{cases} 若 H \in 2i, i \in \mathbb{N} \begin{cases} k = 0 \Leftrightarrow GDT_{\le H}(n,k) \\ k \neq 0 \Leftrightarrow GDT_{\le H+1}(n,k) \end{cases} \\ 若 H \in 2i-1, i \in \mathbb{N} \Leftrightarrow GDT_{\le H}(n,k) \end{cases}$$

$$GDT_{\le H}(n,k)\begin{cases} 若 H \in 2i-1, i \in \mathbb{N} \begin{cases} SDT_{\le H}(n,k) \\ SDT_{\le H-1}(n,k) \end{cases} \\ 若 H \in 2i, i \in \mathbb{N} \Leftrightarrow SDT_{\le H}(n,k) \end{cases}$$

二、探討集合 *SDT* 在高度小於等於三的數量以及與 Motzkin Path 之間的關係

(一) 探討 $S_{\le 3}(n,k)$ 限制特殊規則的數量與路徑之間的關係及對應

1. 對於 $S_{\le 3}(n,k)$，探討骨牌僅以水平方式出現時，其數量與路徑之間的關係

當骨牌僅以水平方式出現，即 D_{aa},D_{bb},D_{cc}，將其分成數種狀況來探討：

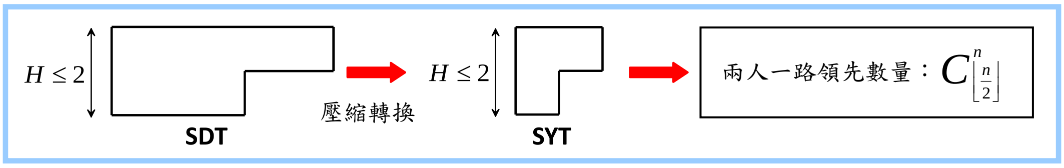
(1). 當 $S_1(n,0)$，骨牌僅以 D_{aa} 形式出現時，其數量僅有一種，如圖(三十四)。

1	2	3	4	5	6	…	…	$2n$-1	$2n$

圖(三十四)

(2). 當 $S_{\le 2}(n,0)$，骨牌僅以 D_{aa},D_{bb} 形式出現時，其數量為何？

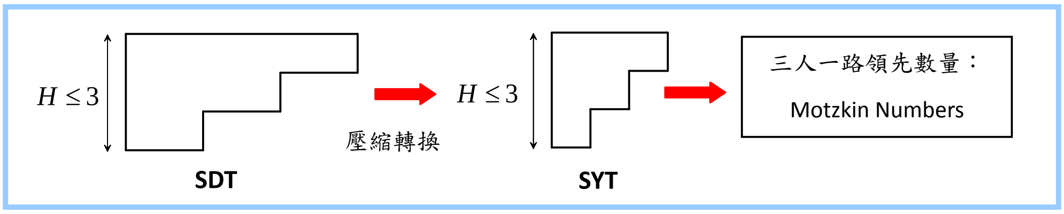
發現若將每塊水平的骨牌壓縮為一格，其數量等價於高度小於等於 2 的 *SYT*，即為兩人「一路領先」的情形，數量為 $C^{n}_{\left\lfloor \frac{n}{2} \right\rfloor}$。



圖(三十五)

(3). 當 $S_{\le 3}(n,0)$，骨牌以 D_{aa},D_{bb},D_{cc} 形式出現時，其數量為何？

同(2)，先將每塊骨牌壓縮為一格，其數量等價於高度小於等於 3 的 *SYT*，即為三人「一路領先」的情形，此數量為 Motzkin Numbers。在文獻[2]及[3]中已證出其數量與高度小於等於 3 的 *SYT* 相等，並找出其路徑之對應。



圖(三十六)

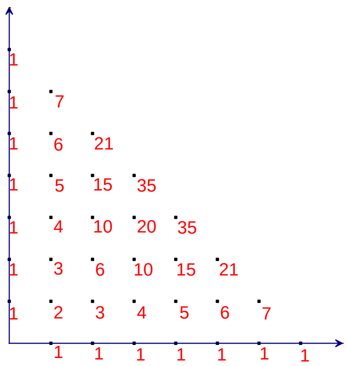
此外，對於$S_{\le H}(n,0), H>3$的情況，皆可同樣經由將每張骨牌的原先兩格壓縮為一格的方法，轉換成高度為H的SYT，即為H人「一路領先」的情形，此部分的結果在先前已被完成[1](2010)。

2. 對於$S_{\le 3}(n,k)$，探討限制直立骨牌出現的數量與路徑之間的關係及對應

(1). 在$S_{\le 2}(n,k)$，骨牌以D_{aa}, D_{ab}, D_{bb}形式出現時，其數量及與路徑之間的關係為何？

我們知道對於任意一種骨牌填入順序的情形，若$W_1 - W_2 = 0$，即當下第一列和第二列長度相同，因此有兩種骨牌D_{aa}, D_{ab}可以填加；$W_1 - W_2 \ge 2$，即當下第一列長度大於第二列長度，因此亦有兩種骨牌D_{aa}, D_{bb}可以填加。根據上述，可以知道其方法數為2^n。

又2^n可以拆成由二項式係數組合成的三角陣列，即$2^n = \sum_{i=0}^{n} C_i^n$，如圖(三十七)所示。其數量等價於：從坐標原點出發，規定走法為依循向量$(1,0)$或$(0,1)$，在第一象限內，走到任意點(m,n)的方法數，其數量可表示為：C_n^{m+n}。



圖(三十七)

(a). $\boxed{\text{對應規則：} SDT \xrightarrow{bijection} Path}$：

(I). 若 W_1=W_2，下一步在 SDT 中新增 D_{aa}，考慮當下路徑中 $[\rightarrow],[\uparrow]$數量的關係，分成下列兩種情況：

(i) 若$[\rightarrow]<[\uparrow]$，在路徑中新增$[\uparrow]$。

(ii) 若$[\rightarrow]\geq[\uparrow]$，在路徑中新增$[\rightarrow]$。

(II). 若 W_1=W_2，下一步在 SDT 中新增 D_{ab}，考慮當下路徑中 $[\rightarrow],[\uparrow]$數量的關係，分成下列兩種情況：

(i) 若$[\rightarrow]<[\uparrow]$，在路徑中新增$[\rightarrow]$。

(ii) 若$[\rightarrow]\geq[\uparrow]$，在路徑中新增$[\uparrow]$。

(III). 若 W_1>W_2，下一步在 SDT 中新增 D_{aa}，考慮當下路徑中 $[\rightarrow],[\uparrow]$數量的關係，分成下列兩種情況：

(i) 若$[\rightarrow]>[\uparrow]$，在路徑中新增$[\rightarrow]$。

(ii) 若$[\rightarrow]<[\uparrow]$，在路徑中新增$[\uparrow]$。

(IV). 若 W_1>W_2，下一步在 SDT 中新增 D_{ab}，考慮當下路徑中 $[\rightarrow],[\uparrow]$數量的關係，分成下列兩種情況：

(i) 若$[\rightarrow]>[\uparrow]$，在路徑中新增$[\uparrow]$。

(ii) 若$[\rightarrow]<[\uparrow]$，在路徑中新增$[\rightarrow]$。

(b). $\boxed{\text{對應規則：} Path \xrightarrow{bijection} SDT}$：

考慮目前$\rightarrow,\uparrow$方向的路徑數量：

(I). 若$[\rightarrow]=[\uparrow]$，下一步新增$[\rightarrow]$，相當於在 SDT 中新增 D_{aa}；下一步新增$[\uparrow]$，相當於在 SDT 中新增 D_{ab}。

(II). 若$\left[\rightarrow\right]>\left[\uparrow\right]$，下一步新增$\left[\rightarrow\right]$，相當於在 SDT 中新增 D_{aa}；下一步新增$\left[\uparrow\right]$，考慮當下 SDT 中 W_1,W_2 的關係，分成下列兩種情況：

(i) 若 W_1-$W_2=0$，在 SDT 中新增 D_{ab}。

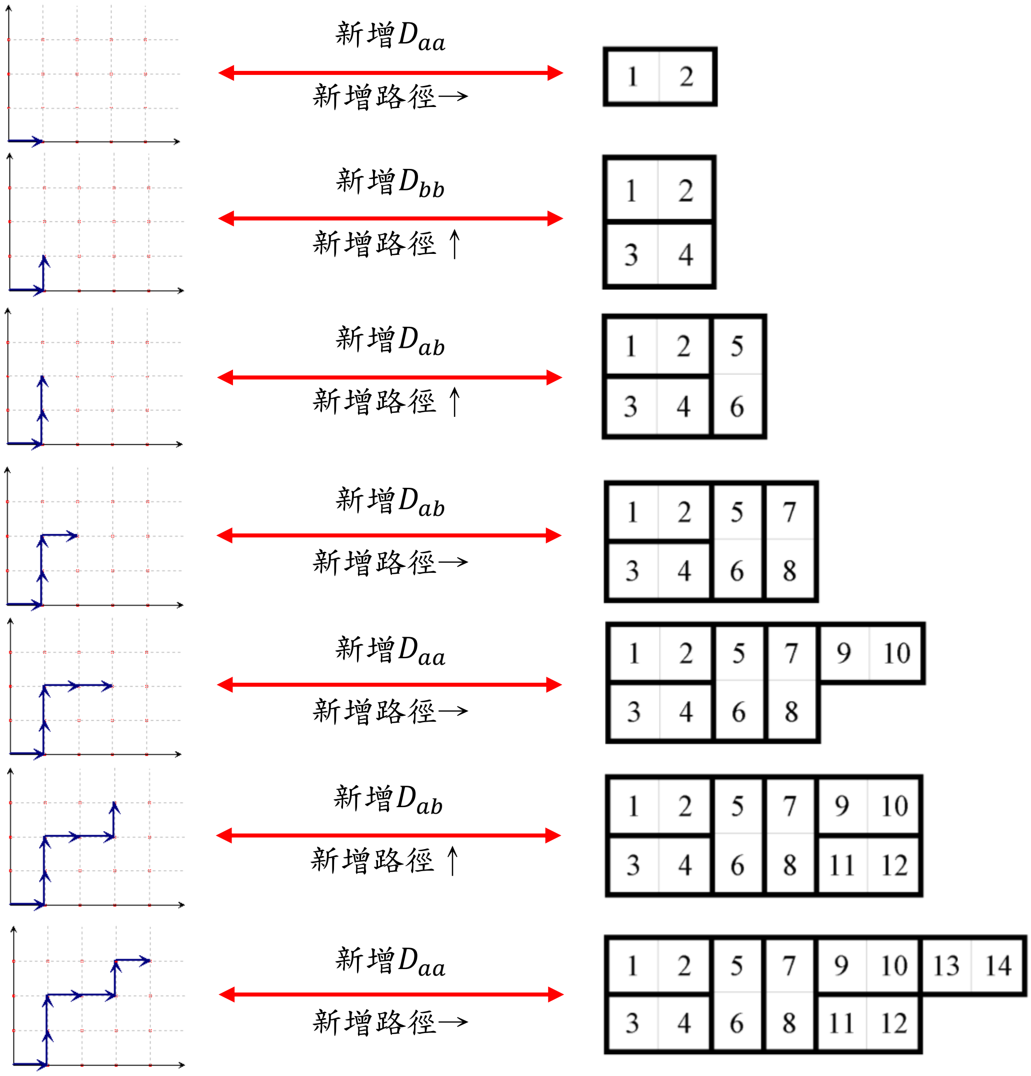
(ii) 若 W_1-$W_2\neq 0$，在 SDT 中新增 D_{bb}。

(III). 若$\left[\rightarrow\right]<\left[\uparrow\right]$，下一步新增$\left[\uparrow\right]$，相當於新增 D_{aa}；下一步新增$\left[\uparrow\right]$，考慮當下 SDT 中 W_1,W_2 的關係，分成下列兩種情況：

(i) 若 W_1-$W_2=0$，在 SDT 中新增 D_{ab}。

(ii) 若 W_1-$W_2\neq 0$，在 SDT 中新增 D_{bb}。

(c). 以下由其中一種7張骨牌時的對應為例：



圖(三十八)

(2). 當 $S_{\le 3}(n,k)$，骨牌缺少 D_{bc} 時，其數量與路徑之間的關係為何？

此問題等價於上述(1). 之問題再加入骨牌 D_{cc} 的排列情形，利用前幾個數量，利用「線上整數數列百科全書」查詢是否有已知的結果，發現該數列與一種特別的路徑方法數前幾項吻合，於是思考與骨牌排列情形的關係：

1, 2, 4, 10, 26, 66, 178, 488, 1320, 3674, ... (OEIS：A151278)

其數列代表的意義為：在第一象限中，從原點 $(0,0)$ 出發，走法依循向量 $(1,0),(0,1),(-1,-1)$，走到任意點的方法數，故我們

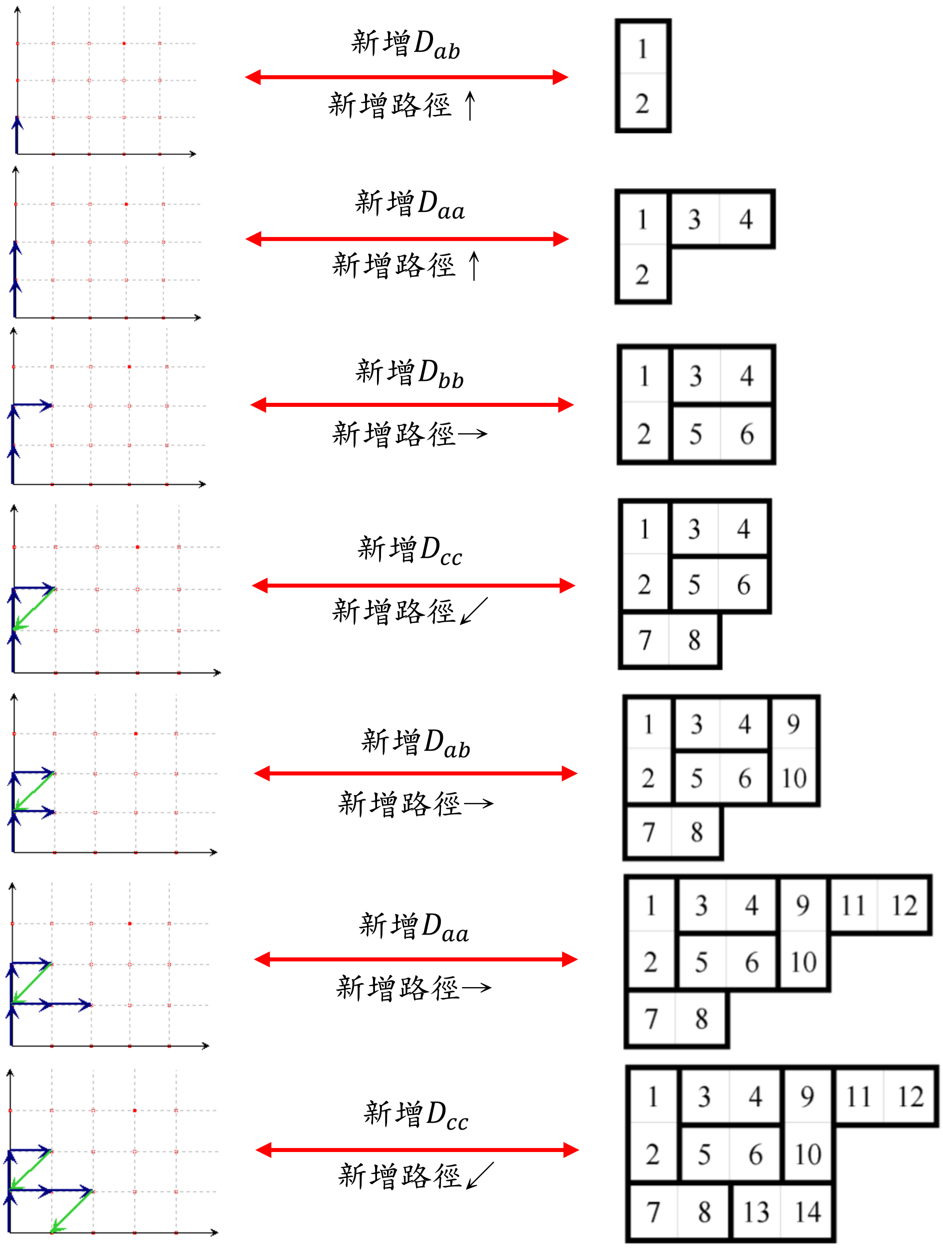
試著找出這兩者之間的對應關係。

(a). 對應規則：承 P. 17 的對應方式再加入以下兩條規則：

(I). 若在 *SDT* 中新增 D_{cc}，則路徑增加[↙]。

(II). 若新增路徑[↙]，則在 *SDT* 中增加 D_{cc}。

(b). 以下由其中一種7張骨牌時的對應為例：



圖(三十九)

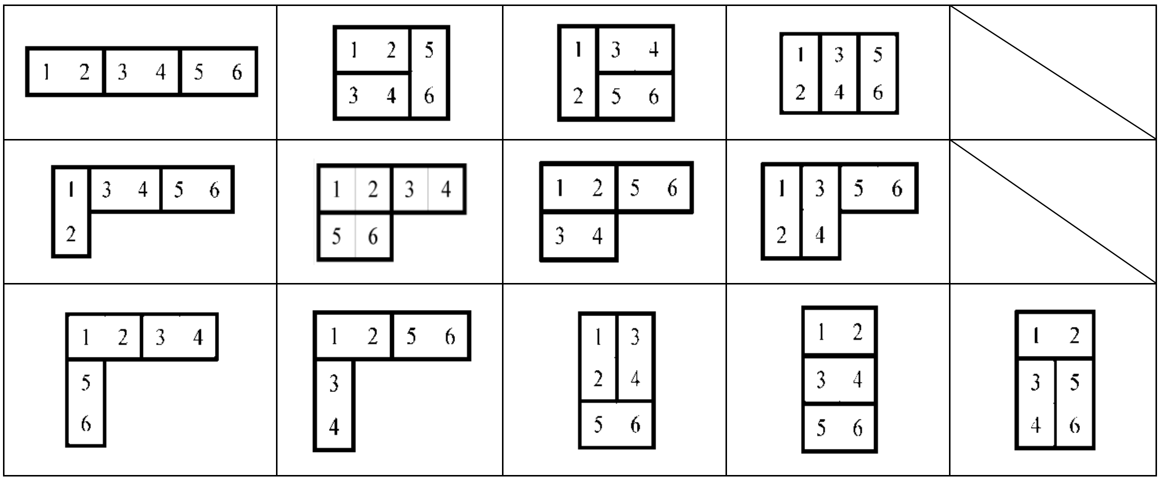
(二) 探討 $S_{\le 3}(n,k)$ 各種不同排列順序的數量與 Motzkin Triangle 之間的關係

1. 探討 $S_{\le 3}(n,k)$ 不同排列順序的數量

已知當 $|S_{\le 3}(1)|=2$，$|S_{\le 3}(2)|=5$，列舉出三塊骨牌且高度小於等於三的情形，並計算其數量。之後，將其數量按照骨牌直立的個數，分類整理如下表所示：

(1). 列舉 $S_{\le 3}(3,k)$：其可能排列之情形共有 13 種，如下表(十三)。

表(十三) 高度 $H \le 3, n=3$ 的所有情形



(2). 依骨牌直立個數分類：其中 n 為骨牌的塊數，k 為骨牌直立的個數。

表(十四) 高度 $H \le 3, n \le 10$ 的所有數量情形

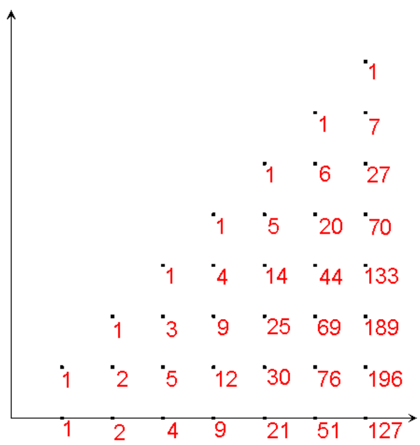
k \ n	1	2	3	4	5	6	7	8	9	10
0	1	2	4	9	21	51	127	323	835	2188
1	1	2	5	12	30	76	196	512	1353	3610
2		1	3	9	25	69	189	518	1422	3915
3			1	4	14	44	133	392	1140	3288
4				1	5	20	70	230	726	2235
5					1	6	27	104	369	1242
6						1	7	35	147	560
7							1	8	44	200
8								1	9	54
9									1	10
10										1

如表(十四)所示，經由觀察，我們發現這些數量之間存在一種關係：每一項都是前一排三個相鄰的數字總和，如：$20=14+5+1$。因此可以寫出它們之間的遞迴關係式：

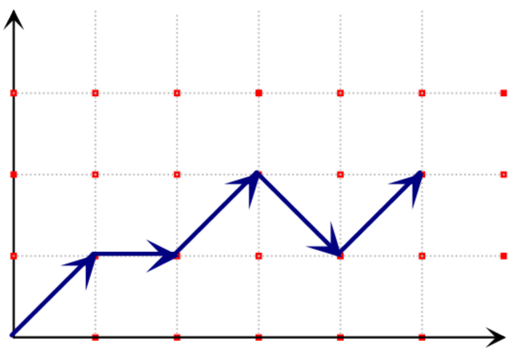
$$M(0,0)=M(1,0)=M(0,1)=1, M(n,0)=1, n\geq 2$$
$$\begin{cases} M(n,k)=M(n-1,k-1)+M(n-1,k)+M(n-1,k+1), \forall k\in\mathbb{N}, k<n \\ M(n,n)=M(n-1,n-1)+M(n-1,n) \end{cases}$$

(3). 由於這個雙變數的遞迴關係式之前幾項數值恰與 Motzkin Triangle 相同！如圖(四十)。發現它可視為從坐標原點 $(0,0)$ 出發，規定走法為依循向量 $(1,0)$、$(1,1)$ 或 $(1,\text{-}1)$，且不得通過 x 軸下方，走到 (n,k) 的方法數，定義此路徑為「廣義 Motzkin 路徑」。

於是我們開始思考這些骨牌排列順序的數量與「廣義 Motzkin 路徑」之間的關係。



圖(四十) Motzkin Triangle


圖(四十一) 廣義 Motzkin 路徑

2. $S_{\leq 3}(n,k)$ 與 Motzkin Triangle 之間的對應關係：

由於發現 SDT 的數量若依照直立的骨牌數量分類，與 Motzkin Triangle 的數量有密切的相關性，並利用電腦程式對其數量較大時的情況進行模擬與計算，發現該數量與 Motzkin Triangle 之遞迴關係式皆符合，因此我們猜測對於 SDT 在 $H\leq 3$ 時各種圖形均有一個對應方式可

以與「廣義 Motzkin 路徑」對應。

對應規則：我們先從塊數較少的骨牌中嘗試找尋規律，並思考其規則的合理性及一致性；但在擴展至更多塊骨牌時，發現先前的對應方式發生矛盾，不同的骨牌會對應到相同的路徑。於是重新思考規則，經過多次修正，並使用電腦驗證此對應規則，找出一套完整且合理的對應方式，將其列出如下，並舉出三個例子作為說明：

(1). 符號表示：

(a). 將 SDT 中 D_{aa} 以 a 表示、 D_{bb} 以 b 表示、 D_{cc} 以 c 表示、 D_{ab} 以$\nearrow_{ab}$表示、 D_{bc} 以$\nearrow_{bc}$表示，依照骨牌填入的順序以 $a, b, c, \nearrow_{ab}, \nearrow_{bc}$表示。

(b). 向量$\nearrow$表示向量 $(1,1)$ 、向量 $\rightarrow$ 表示向量 $(1,0)$ 、向量$\searrow$表示向量 $(1,-1)$ 。

(2). 配組操作：(備註：標記已配組的每一符號，在置換後則取消所有標記)

(a). 由左至右依序找到各個 c 。

(b). 承(a). 將每個 c 嘗試往左尋找兩個未被標記的$\nearrow_{ab}$，且其途中不存在未被標記的 b 。

(I). 若存在兩個未被標記的$\nearrow_{ab}$

(i). 由此 c 向左找最鄰近且未被標記之$\nearrow_{ab}$。

(ii). 承(i). 將 c 與此$\nearrow_{ab}$設為「特殊配組」，並將此$\nearrow_{ab}$標記之。

(II). 若不存在兩個未被標記的$\nearrow_{ab}$

(i). 向左找最鄰近且未被標記的 b 。

(ii). 承(i). 繼續向左找未被標記的 a 。

(iii). 承(i). (ii). 將 (a,b) 與 (b,c) 配組，並將 a,b 標記。

(c). 由左至右依序找到各個未被標記的 b

(I). 繼續向左找最近未被標記的 a 。

(II). 承(I). 將 (a,b) 配組，並將 a 標記。

(3). 置換操作：

(a). 考慮各配組之區間 $[p,q]$ $(p<q)$ ，將 q 由右往左進行考慮各區間：

(I). 若此配組為「特殊配組」，將本配組進行互相置換。

(II). 若不為「特殊配組」，則由 $q-1$ 往左直至 $p+1$，考慮此區間範圍：

(i). 若找到$\nearrow_{ab}$或$\nearrow_{bc}$，考慮以下狀況：

①. 若此配組為 (a,b) 之配組，將$\nearrow_{ab}$或$\nearrow_{bc}$與 b 置換，取消所有標記，由(2).重新配組。

②. 若此配組為 (b,c) 之配組，將$\nearrow_{ab}$或$\nearrow_{bc}$與 c 置換，並將 c 設為 $\rightarrow$。此後，設為 $\rightarrow$ 的 c 不再參與任何配組。取消所有標記，由(2).重新配組。

(ii). 若找不到$\nearrow_{ab}$或$\nearrow_{bc}$，則繼續往左考慮下一區間。

(b). 先將所有$\nearrow_{ab}$或$\nearrow_{bc}$設為向量$\nearrow$

(I). 由左至右依序找到各個 c ，設為向量$\searrow$。

(II). 承(I). 每一個 c 向左找最近的 b ，設為向量$\rightarrow$。

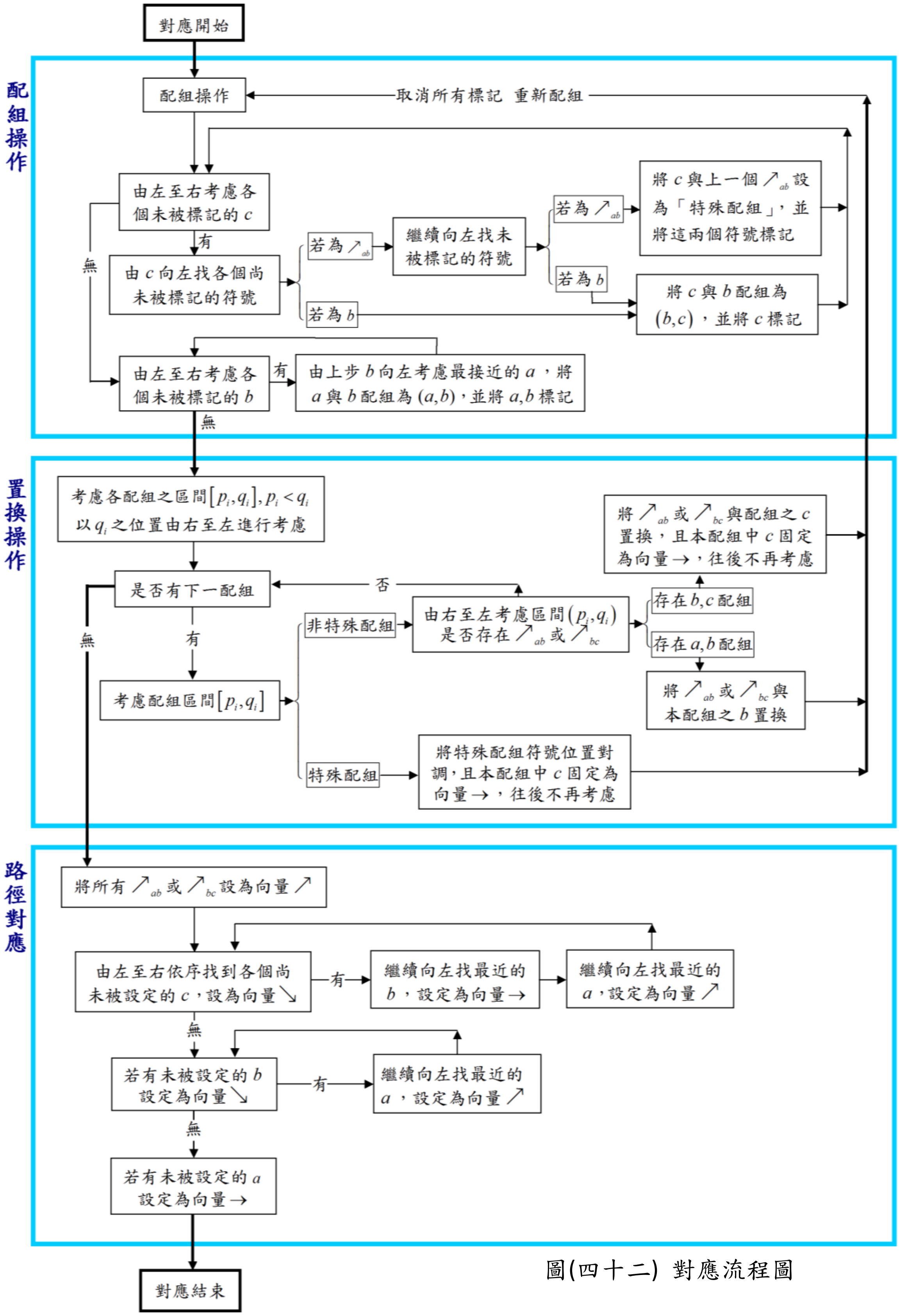
(III). 承(II). 每一個 b 向左找最近的 a ，設為向量$\nearrow$。

(IV). 承(I). (II). (III). 若有未被設定向量的 b ，設為向量$\searrow$。。

(V). 承(IV). 每一個 b 向左找最近的 a ，設為向量$\nearrow$。

(VI). 承(IV). (V). 若有未被設定向量的 a ，設為向量$\rightarrow$。

(4). 對應規則流程圖：



圖(四十二) 對應流程圖

(5). 舉例說明：

先將原始的符號列出，其中針對每個配組以相同顏色標示，若有需要置換的配組則以紅色表示，在此舉出以下三個例子：

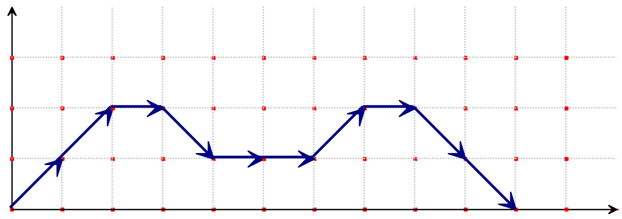
(a). 例一：($k=0$) 此情形不需配組或置換，直接進行路徑對應。

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

↕ 壓縮轉換為 *SYT*

1	2	6	7
3	5	8	
4	9	10	

↓



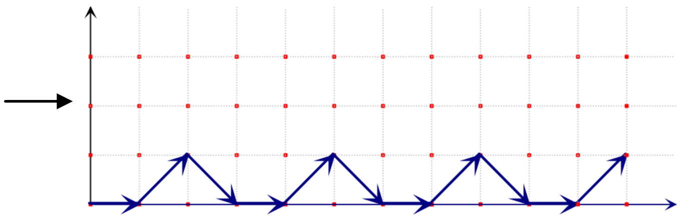
圖(四十三) $k=0$ 對應示意圖

表(十五)

原始	a	a	b	c	b	a	a	b	c	c
路徑	a	a	b	c	b	a	a	b	c	c
路徑	a	a	b	↘	b	a	a	b	↘	↘
路徑	a	a	→	↘	→	a	a	→	↘	↘
路徑	↗	↗	→	↘	→	a	↗	→	↘	↘
路徑	↗	↗	→	↘	→	→	↗	→	↘	↘

(b). 例二：($k=1$)

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



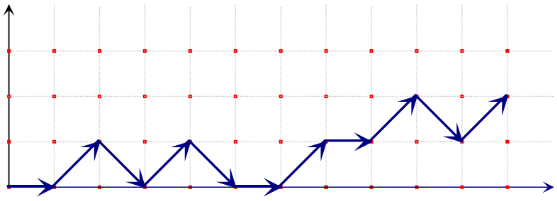
圖(四十四)

表(十六)

原始	a	a	↗	b	a	c	b	a	c	b	c
配組	a	a	↗	b	a	c	b	a	c	b	c
置換	a	a	b	↗	a	c	b	a	c	b	c
配組	a	a	b	↗	a	c	b	a	c	b	c
置換	a	a	b	→	a	↗	b	a	c	b	c
配組	a	a	b	→	a	↗	b	a	c	b	c
置換	a	a	b	→	a	b	↗	a	c	b	c
配組	a	a	b	→	a	b	↗	a	c	b	c
置換	a	a	b	→	a	b	→	a	↗	b	c
配組	a	a	b	→	a	b	→	a	↗	b	c
置換	a	a	b	→	a	b	→	a	b	↗	c
配組	a	a	b	→	a	b	→	a	b	↗	c
置換	a	a	b	→	a	b	→	a	b	→	↗
配組	a	a	b	→	a	b	→	a	b	→	↗
路徑	→	↗	↘	→	↗	↘	→	↗	↘	→	↗

(c). 例三：($k=2$)

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



圖(四十五)

表(十七)

原始	a	a	↗	a	↗	b	c	b	a	c	b
配組	a	a	↗	a	↗	b	c	b	a	c	b
置換	a	a	↗	a	b	b	c	↗	a	c	b
配組	a	a	↗	a	b	b	c	↗	a	c	b
置換	a	a	↗	a	b	b	c	→	a	↗	b
配組	a	a	↗	a	b	b	c	→	a	↗	b
置換	a	a	↗	a	b	b	c	→	a	b	↗
配組	a	a	↗	a	b	b	c	→	a	b	↗
置換	a	a	b	a	↗	b	c	→	a	b	↗
配組	a	a	b	a	↗	b	c	→	a	b	↗
置換	a	a	b	a	b	↗	c	→	a	b	↗
配組	a	a	b	a	b	↗	c	→	a	b	↗
置換	a	a	b	a	b	→	↗	→	a	b	↗
配組	a	a	b	a	b	→	↗	→	a	b	↗
路徑	→	↗	↘	↗	↘	→	↗	→	↗	↘	↗

(6). 「廣義 Motzkin 路徑」與 $S_{\le 3}(n,k)$ 之間的對應關係：

我們將「廣義 Motzkin 路徑」對應至集合 SDT 的操作過程列出如下，以確認此對應為 1-1 且 onto。

(a). 配組：由左至右，考慮每個向量↗，往左找最鄰近且未被標記的向量↘：

(I). 若向量↗與向量↘之間存在未被標記的向量→，考慮區間[向量↗位置,向量↘位置]中最接近向量↗的向量→，則將上一步的向量↗、向量→以及向量↘，配組為 (a,b,c)，並將此三符號標記。

(II). 若整體均已無(I). 之情況，開始考慮若向量↘與向量↗之間不存在未被標記的向量→，將向量↗以及向量↘，配組為 (a,b)，並將此兩符號標記。

(b). 由左至右，考慮未被標記的向量↗，進行以下操作：

(I). 考慮由最左邊至上步的向量↗區間範圍的目前狀態，考慮被配組為 a 的位置視為 D_{aa}，配組為 b 的位置視為 D_{bb}，配組為 c 的位置視為 D_{cc}，由上述骨牌所組成的 SDT：

(i). 若 $W_1 = W_2$，將向量↗定義為$↗_{ab}$，繼續向右考慮未被標記的向量↗，並重複步驟(b).。

(ii). 若 $W_1 > W_2 = W_3$，將向量↗定義為$↗_{bc}$，繼續向右考慮未被標記的向量↗，並重複步驟(b).。

(II). 若不滿足(i). (ii).，將向量↗進行位置置換操作：

(i). 在由最左邊至上步的向量↗的區間範圍中，由向量↗往左找尋配組 (a,b) 的過程中是否有未被標記的向量→。

①. 若無，則將配組 (a,b) 中的 b 與向量↗位置置換，並重新步驟(b).。

②. 若有，考慮其可能為符號 c，則將 c 與向量→置換，將(i)中的配組 (a,b) 更改為配組 (a,b,c)，並重新步驟(b).。

③. 承②. 若其不為符號 c，則進行①. 之操作，並重新步驟(b).。

(ii). 若(i). 中無法找到未被使用的配組 (a,b)，且由最左邊至上步的向量↗的區間範圍中尚有未被標記的向量→，且該向量前仍有被定義為↗$_{ab}$的向量，則考慮與向量↗較接近且未被標記的向量 →，定義為「特殊配對」; 並將向量↗與向量→進行置換，並將此兩符號標記，重新步驟(b).。

(c). 考慮所有未被標記的向量→，定義為骨牌 D_{aa}；配組 (a,b) 定義為骨牌 D_{aa} 與骨牌 D_{bb}；配組 (a,b,c) 定義為骨牌 D_{aa},D_{bb},D_{cc}。對於其他未被配組的向量↗，考慮由最左邊至向量↗的區間範圍所形成的 SDT，進行以下操作：

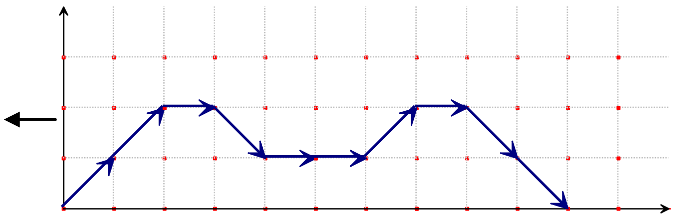
(I). 若 $W_1 = W_2$，將向量↗定義為骨牌 D_{ab}

(II). 若 $W_1 > W_2 = W_3$，將向量↗定義為骨牌 D_{bc}

(d). 對應完成。

(7). 我們將先前「例一」與「例二」的最終「廣義 Motzkin 路徑」與原先之 $S_{\le 3}(n,k)$ 間的操作過程列出如下：

(a). 例一 (由於沒有向量↗，不需置換，僅需配組。)



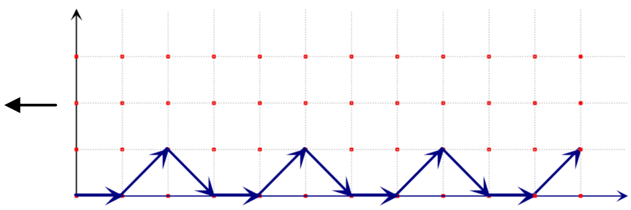
圖(四十六)

表(十八)

原始	↗	↗	→	↘	→	→	↗	→	↘	↘
配組	↗	a	b	c	→	→	↗	→	↘	↘
配組	↗	a	b	c	→	→	a	b	↘	c
配組	a	a	b	c	b	→	a	b	c	c
路徑	a	a	b	c	b	a	a	b	c	c
路徑	D_{aa}	D_{aa}	D_{bb}	D_{cc}	D_{bb}	D_{aa}	D_{aa}	D_{bb}	D_{cc}	D_{cc}

(b). 例二

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



圖(四十七)

表(十九)

路徑	$\rightarrow$	$\nearrow$	$\searrow$	$\rightarrow$	$\nearrow$	$\searrow$	$\rightarrow$	$\nearrow$	$\searrow$	$\rightarrow$	$\nearrow$
配組	$\rightarrow$	a	b	$\rightarrow$	$\nearrow$	$\searrow$	$\rightarrow$	$\nearrow$	$\searrow$	$\rightarrow$	$\nearrow$
配組	$\rightarrow$	a	b	$\rightarrow$	a	b	$\rightarrow$	$\nearrow$	$\searrow$	$\rightarrow$	$\nearrow$
配組	$\rightarrow$	a	b	$\rightarrow$	a	b	$\rightarrow$	a	b	$\rightarrow$	$\nearrow$
考慮	$\rightarrow$	a	b	$\rightarrow$	a	b	$\rightarrow$	a	b	$\rightarrow$	$\nearrow$
置換	$\rightarrow$	a	b	$\rightarrow$	a	b	$\rightarrow$	a	b	$\nearrow$	c
置換	$\rightarrow$	a	b	$\rightarrow$	a	b	$\rightarrow$	a	$\nearrow$	b	c
置換	$\rightarrow$	a	b	$\rightarrow$	a	b	$\nearrow$	a	c	b	c
置換	$\rightarrow$	a	b	$\rightarrow$	a	b	$\nearrow$	a	c	b	c
置換	$\rightarrow$	a	b	$\rightarrow$	a	$\nearrow$	b	a	c	b	c
置換	$\rightarrow$	a	b	$\nearrow$	a	c	b	a	c	b	c
置換	$\rightarrow$	a	$\nearrow$	b	a	c	b	a	c	b	c
骨牌	D_{aa}	D_{aa}	D_{bc}	D_{bb}	D_{aa}	D_{cc}	D_{bb}	D_{aa}	D_{cc}	D_{bb}	D_{cc}

(8). $S_{\leq 3}(n)$ 數量之計算：

由此對應可知：SDT 共 n 塊骨牌的數量可對應到「廣義 Motzkin 路徑」，只需計算路徑的方法數為何，即可得知 $|S_{\leq 3}(n)|$ 。

若不考慮「廣義 Motzkin 路徑」的水平部份，則單純上升、下降的路徑總數可以視為兩人「一路領先」方法數。

兩人「一路領先」方法數為：$C^{n}_{\left\lfloor \frac{n}{2} \right\rfloor}$ 。

以加總的方式可呈現「廣義 Motzkin 路徑」的總方法數為：$\sum_{i=0}^{n} C^{i}_{\left\lfloor \frac{i}{2} \right\rfloor} C^{n}_{i}$ 。

陸、結論

一、對於集合 *SDT* 及集合 *GDT* 的圖形，可找到一個對應方式證明這兩集合的數量相同，意即$|S_{\le H}(n,k)| = |G_{\le H}(n,k)|, \forall H = 2i-1, i \in \mathbb{N}$。

二、對於有限制骨牌種類的 *SDT* ：

(一) $\left|S_{\le 2}(n)\right| = 2^n$，由於 $2^n = \sum_{i=0}^{n} C_i^n$，意即對應到在平面坐標從原點$(0,0)$出發，並以向量$(1,0)$及$(0,1)$為路徑方向，在第一象限中(包含 x, y 軸正向)，走到任一點的方法數。

(二) 在 $S_{\le 3}(n)$ 中，限制不得有骨牌 D_{bc} 出現，則其方法數恰對應到：在平面坐標上從原點出發以向量$(1,0)$、$(0,1)$及$(-1,-1)$為路徑方向，在第一象限中(包含 x, y 軸正向)，走到任一點的方法數。

三、在 $S_{\le 3}(n)$ 中，依照骨牌直立的個數加以分類，其方法數恰為 Motzkin Triangle，且與在坐標平面上，從原點走到(n,k)的「廣義 Motzkin 路徑」一一對應。

對於 $n = 1, 2, ..., 9, ...$，列舉出$\left|S_{\le 3}(n)\right|$的值：2, 5, 13, 35, 96, 267, 750, 2123, 6046, ...

方法數為：$\sum_{i=0}^{n} C_i^n C_{\left\lfloor \frac{i}{2} \right\rfloor}^i$。

柒、討論

一、上述研究中皆以考慮骨牌填入的順序去探討其方法數。若不考慮骨牌填入順序的方法數，初步得出以下結果：

(一) 不考慮骨牌放置順序且限制高度小於等於三排，其數列前 8 項為：

2, 5, 11, 21, 40, 76, 134, 239, ...

(二) 不考慮骨牌放置順序以及高度限制，其數列前 7 項為：

2, 6, 16, 42, 106, 268, 650, ...

關於以上兩數列，在現有的數列資料中[7]尚未找尋到相關資料，希望在未來能夠對其進行深入研究，並嘗試找出其遞迴關係式或一般式以計算其數量與應用。

二、希望能在三維空間中，將骨牌以立體堆疊的方式探討其方法數以及各種性質。

三、若將填入的骨牌由1×2改成1×3、$1\times4,\cdots$，甚至是$m\times n$，探討其方法數以及各種性質。

捌、參考資料

[1] 侯宗誠、許德瑋 (2010)。由蟲子問題衍生一路領先與 Motzkin 路徑之對應及推廣。 2010 台灣國際科學展覽會優勝作品專輯。

[2] 游森棚 (2009)。游理數，數裡游－骨牌拼圖。科學月刊。第四十卷第十期。頁740-741。

[3] Eric S. Egge, Enumerating *rc*-Invariant Permutations with No Long Decreasing Subsequences, *Annals of Combinatorics*, (2010).

[4] P.W. Kasteleyn, The statistics of dimers on a lattice, I: The number of dimer arrangements on a quadratic lattice, Physica 27, (1961), p 1209-1225.

[5] R. Donaghey and L.W. Shapiro, Motzkin Numbers, *Journal of Combinatorial Theory, Series A* 23, 291-301, (1977).

[6] S.-P. Eu, Skew-standard tableaux with three rows, *Advances in Applied Mathematics*, (2010).

[7] 線上整數數列百科全書 (The On-Line Encyclopedia of Integer Sequences) http://oeis.org/

[8] Motzkin Number. (OEIS：A001006) http://oeis.org/A001006

[9] Number of directed animals of size n. (OEIS：A005773) http://oeis.org/A005773

[10] Motzkin Triangle. (OEIS：A026300) http://oeis.org/A026300

[11] Number of walks: Starting at $(0,0)$ and consisting of n steps taken from $(1,0),(0,1),(-1,-1)$ (OEIS：A151278) http://oeis.org/A151278

[12] Motzkin Number-Wolfram MathWorld http://mathworld.wolfram.com/MotzkinNumber.html

[13] Domino Tiling-Wolfram MathWorld http://mathworld.wolfram.com/DominoTiling.html

[14] Young Tableau-Wolfram MathWorld http://mathworld.wolfram.com/YoungTableau.html

[15] Pascal Triangle-Wolfram MathWorld http://mathworld.wolfram.com/PascalsTriangle.html

附錄一 集合 SDT 與集合 GDT（$n=1,2,3,4$）之間對應的每一步轉換過程(由左而右)：

$GDT_{\le 3}(1)$ 對應至 $SDT_{\le 3}(1)$ 的情況：

符號定義：$(D;\, p,q,r)$

D：表示水平骨牌的數量。

(p,q,r)：分別表示 SDT 圖形第一列、第二列以及第三列的長度。

(**1**; 3)	(**1**; 2)
$\begin{array}{\|c\|c\|c\|}\hline \# & 1 & 2 \\ \hline\end{array}$	$\begin{array}{\|c\|c\|}\hline 1 & 2 \\ \hline\end{array}$
(**0**; 1,1,1)	(**0**; 1,1)
$\begin{array}{\|c\|}\hline \# \\ \hline 1 \\ \hline 2 \\ \hline\end{array}$	$\begin{array}{\|c\|}\hline 1 \\ \hline 2 \\ \hline\end{array}$

$GDT_{\le 3}(2)$ 對應至 $SDT_{\le 3}(2)$ 的情況：

(**2**; 5)	(**2**; 5)	(**2**; 4)
$\begin{array}{\|c\|c\|c\|c\|c\|}\hline \# & 1 & 2 & 3 & 4 \\ \hline\end{array}$	$\begin{array}{\|c\|c\|c\|c\|c\|}\hline 1 & 2 & \# & 3 & 4 \\ \hline\end{array}$	$\begin{array}{\|c\|c\|c\|c\|}\hline 1 & 2 & 3 & 4 \\ \hline\end{array}$
(**2**; 3,2)	(**2**; 2,2)	
$\begin{array}{\|c\|c\|c\|}\hline \# & 1 & 2 \\ \hline 3 & 4 \\ \cline{1-2}\end{array}$	$\begin{array}{\|c\|c\|}\hline 1 & 2 \\ \hline 3 & 4 \\ \hline\end{array}$	
(**1**; 3,1,1)	(**1**; 2,1,1)	
$\begin{array}{\|c\|c\|c\|}\hline \# & 1 & 2 \\ \hline 3 \\ \cline{1-1} 4 \\ \cline{1-1}\end{array}$	$\begin{array}{\|c\|c\|}\hline 1 & 2 \\ \hline 3 \\ \cline{1-1} 4 \\ \cline{1-1}\end{array}$	
(**1**; 3,1,1)	(**1**; 3,1)	
$\begin{array}{\|c\|c\|c\|}\hline \# & 3 & 4 \\ \hline 1 \\ \cline{1-1} 2 \\ \cline{1-1}\end{array}$	$\begin{array}{\|c\|c\|c\|}\hline 1 & 3 & 4 \\ \hline 2 \\ \cline{1-1}\end{array}$	
(**0**; 2,2,1)	(**0**; 2,2)	
$\begin{array}{\|c\|c\|}\hline \# & 3 \\ \hline 1 & 4 \\ \hline 2 \\ \cline{1-1}\end{array}$	$\begin{array}{\|c\|c\|}\hline 1 & 3 \\ \hline 2 & 4 \\ \hline\end{array}$	

$GDT_{\le 3}(3)$ 對應至 $SDT_{\le 3}(3)$ 的情況：

(**3**; 7)	(**3**; 7)	(**3**; 7)	(**3**; 6)
$\begin{array}{\|c\|c\|c\|c\|c\|c\|c\|}\hline \# & 1 & 2 & 3 & 4 & 5 & 6 \\ \hline\end{array}$	$\begin{array}{\|c\|c\|c\|c\|c\|c\|c\|}\hline 1 & 2 & \# & 3 & 4 & 5 & 6 \\ \hline\end{array}$	$\begin{array}{\|c\|c\|c\|c\|c\|c\|c\|}\hline 1 & 2 & 3 & 4 & \# & 5 & 6 \\ \hline\end{array}$	$\begin{array}{\|c\|c\|c\|c\|c\|c\|}\hline 1 & 2 & 3 & 4 & 5 & 6 \\ \hline\end{array}$

(**3;** 5,2)

#	1	2	3	4
5	6			

(**3;** 5,2)

1	2	#	3	4
5	6			

(**3;** 4,2)

1	2	3	4
5	6		

(**3;** 5,2)

#	1	2	5	6
3	4			

(**3;** 5,2)

1	2	#	5	6
3	4			

(**3;** 4,2)

1	2	5	6
3	4		

(**3;** 3,2,2)

#	1	2
3	4	
5	6	

(**3;** 2,2,2)

1	2
3	4
5	6

(**2;** 5,1,1)

#	1	2	3	4
5				
6				

(**2;** 5,1,1)

1	2	#	3	4
5				
6				

(**2;** 4,1,1)

1	2	3	4
5			
6			

(**2;** 5,1,1)

#	1	2	5	6
3				
4				

(**2;** 5,1,1)

1	2	#	5	6
3				
4				

(**2;** 4,1,1)

1	2	5	6
3			
4			

(**2;** 5,1,1)

#	3	4	5	6
1				
2				

(**2;** 5,1)

1	3	4	5	6
2				

(**2;** 3,3,1)

#	1	2
3	5	6
4		

(**2;** 3,3,1)

1	2	#
3	5	6
4		

(**2;** 3,3,1)

1	2	5
3	#	6
4		

(**2;** 3,3)

1	2	5
3	4	6

(**2;** 3,3,1)

#	3	4
1	5	6
2		

(**2;** 3,3)

1	3	4
2	5	6

(**1;** 4,2,1)

#	3	5	6
1	4		
2			

(**1;** 4,2)

1	3	5	6
2	4		

(**1;** 3,2,2)

#	1	2
3	5	
4	6	

(**1;** 2,2,2)

1	2
3	5
4	6

(**1;** 3,2,2)

#	3	4
1	5	
2	6	

(**1;** 3,2,2)

1	3	4
2	5	
#	6	

(**1;** 3,2,2)

1	3	4
2	#	
5	6	

(**1;** 2,2,2)

1	3
2	4
5	6

(**0;** 3,3,1)

#	3	5
1	4	6
2		

(**0;** 3,3)

1	3	5
2	4	6

$GDT_{\le 3}(4)$ 對應至 $SDT_{\le 3}(4)$ 的情況：

(**4;** 9)

#	1	2	3	4	5	6	7	8

(**4;** 9)

1	2	#	3	4	5	6	7	8

(**4;** 9)

1	2	3	4	#	5	6	7	8

(**4;** 9)

1	2	3	4	5	6	#	7	8

(**4;** 8)

1	2	3	4	5	6	7	8

(**4;** 7,2)

#	1	2	3	4	5	6
7	8					

(**4;** 7,2)

1	2	#	3	4	5	6
7	8					

(**4;** 7,2)

1	2	3	4	#	5	6
7	8					

(**4;** 6,2)

1	2	3	4	5	6
7	8				

(**4;** 7,2)

#	1	2	3	4	7	8
5	6					

(**4;** 7,2)

1	2	#	3	4	7	8
5	6					

(**4;** 7,2)

1	2	3	4	#	7	8
5	6					

(**4;** 6,2)

1	2	3	4	7	8
5	6				

(**4;** 7,2)

#	1	2	5	6	7	8
3	4					

(**4;** 7,2)

1	2	#	5	6	7	8
3	4					

(**4;** 7,2)

1	2	5	6	#	7	8
3	4					

(**4;** 6,2)

1	2	5	6	7	8
3	4				

(**4;** 5,4)

#	1	2	3	4
5	6	7	8	

(**4;** 5,4)

1	2	#	3	4
5	6	7	8	

(**4;** 4,4)

1	2	3	4
5	6	7	8

(**4;** 5,4)

#	1	2	5	6
3	4	7	8	

(**4;** 5,4)

1	2	#	5	6
3	4	7	8	

(**4;** 4,4)

1	2	5	6
3	4	7	8

(**4;** 5,2,2)

#	1	2	3	4
5	6			
7	8			

(**4;** 5,2,2)

1	2	#	3	4
5	6			
7	8			

(**4;** 4,2,2)

1	2	3	4
5	6		
7	8		

(**4;** 5,2,2)

#	1	2	5	6
3	4			
7	8			

(**4;** 5,2,2)

1	2	#	5	6
3	4			
7	8			

(**4;** 4,2,2)

1	2	5	6
3	4		
7	8		

(**4;** 5,2,2)

#	1	2	7	8
3	4			
5	6			

(**4;** 5,2,2)

1	2	#	7	8
3	4			
5	6			

(**4;** 4,2,2)

1	2	7	8
3	4		
5	6		

(**3;** 7,1,1)

#	1	2	3	4	5	6
7						
8						

(**3;** 7,1,1)

1	2	#	3	4	5	6
7						
8						

(**3;** 7,1,1)

1	2	3	4	#	5	6
7						
8						

(**3;** 6,1,1)

1	2	3	4	5	6
7					
8					

(**3;** 7,1,1)

#	1	2	3	4	7	8
5						
6						

(**3;** 7,1,1)

1	2	#	3	4	7	8
5						
6						

(**3;** 7,1,1)

1	2	3	4	#	7	8
5						
6						

(**3;** 6,1,1)

1	2	3	4	7	8
5					
6					

(**3;** 7,1,1)

#	1	2	5	6	7	8
3						
4						

(**3;** 7,1,1)

1	2	#	5	6	7	8
3						
4						

(**3;** 7,1,1)

1	2	5	6	#	7	8
3						
4						

(**3;** 6,1,1)

1	2	5	6	7	8
3					
4					

(**3;** 7,1,1)

#	3	4	5	6	7	8
1						
2						

(**3;** 7,1)

1	3	4	5	6	7	8
2						

(**3;** 5,3,1)

#	1	2	3	4
5	7	8		
6				

(**3;** 5,3,1)

1	2	#	3	4
5	7	8		
6				

(**3;** 4,3,1)

1	2	3	4
5	7	8	
6			

(**3;** 5,3,1)

#	1	2	5	6
3	7	8		
4				

(**3;** 5,3,1)

1	2	#	5	6
3	7	8		
4				

(**3;** 4,3,1)

1	2	5	6
3	7	8	
4			

(**3;** 5,3,1)

#	1	2	7	8
3	5	6		
4				

(**3;** 5,3,1)

1	2	#	7	8
3	5	6		
4				

(**3;** 5,3,1)

1	2	5	7	8
3	#	6		
4				

(**3;** 5,3)

1	2	5	7	8
3	4	6		

(**3;** 5,3,1)

#	3	4	5	6
1	7	8		
2				

(**3;** 5,3)

1	3	4	5	6
2	7	8		

(**3;** 5,3,1)

#	3	4	7	8
1	5	6		
2				

(**3;** 5,3)

1	3	4	7	8
2	5	6		

(**3;** 3,3,3)

#	1	2
3	4	7
5	6	8

(**3;** 3,3,3)

1	2	#
3	4	7
5	6	8

(**3;** 3,3,2)

1	2	7
3	4	8
5	6	

(**3;** 3,3,3)

#	1	2
3	5	6
4	7	8

(**3;** 3,3,3)

1	2	#
3	5	6
4	7	8

(**3;** 3,3,3)

1	2	5
3	#	6
4	7	8

(**3;** 3,3,3)

1	2	5
3	4	6
#	7	8

(**3;** 3,3,2)

1	2	5
3	4	6
7	8	

(**3;** 3,3,3)

#	3	4
1	5	6
2	7	8

(**3;** 3,3,3)

1	3	4
2	5	6
#	7	8

(**3;** 3,3,2)

1	3	4
2	5	6
7	8	

(**2;** 6,2,1)

#	3	5	6	7	8
1	4				
2					

(**2;** 6,2)

1	3	5	6	7	8
2	4				

(**2;** 5,2,2)

#	1	2	3	4
5	7			
6	8			

(**2;** 5,2,2)

1	2	#	3	4
5	7			
6	8			

(**2;** 4,2,2)

1	2	3	4
5	7		
6	8		

(**2;** 5,2,2)

#	1	2	5	6
3	7			
4	8			

(**2;** 5,2,2)

1	2	#	5	6
3	7			
4	8			

(**2;** 4,2,2)

1	2	5	6
3	7		
4	8		

(**2;** 5,2,2)

#	1	2	7	8
3	5			
4	6			

(**2;** 5,2,2)

1	2	#	7	8
3	5			
4	6			

(**2;** 4,2,2)

1	2	7	8
3	5		
4	6		

(**2;** 5,2,2)

#	3	4	5	6
1	7			
2	8			

(**2;** 5,2,2)

1	3	4	5	6
2	7			
#	8			

(**2;** 5,2,2)

1	3	4	5	6
2	#			
7	8			

(**2;** 5,2,2)

1	3	#	5	6
2	4			
7	8			

(**2;** 4,2,2)

1	3	5	6
2	4		
7	8		

(**2;** 5,2,2)

#	3	4	7	8
1	5			
2	6			

(**2;** 5,2,2)

1	3	4	7	8
2	5			
#	6			

(**2;** 5,2,2)

1	3	4	7	8
2	#			
5	6			

(**2;** 5,2,2)

1	3	#	7	8
2	4			
5	6			

(**2;** 4,2,2)

1	3	7	8
2	4		
5	6		

(**2;** 4,4,1)

#	1	2	7
3	5	6	8
4			

(**2;** 4,4,1)

1	2	#	7
3	5	6	8
4			

(**2;** 4,4,1)

1	2	5	7
3	#	6	8
4			

(**2;** 4,4)

1	2	5	7
3	4	6	8

(**2;** 4,4,1)

#	3	4	7
1	5	6	8
2			

(**2;** 4,4)

1	3	4	7
2	5	6	8

***

(**2;** 4,4,1)

#	3	5	6
1	4	7	8
2			

(**2;** 4,4)

1	3	5	6
2	4	7	8

***

(**1;** 5,3,1)

#	3	5	7	8
1	4	6		
2				

(**1;** 5,3)

1	3	5	7	8
2	4	6		

***

(**1;** 3,3,3)

#	1	2
3	5	7
4	6	8

(**1;** 3,3,3)

1	2	#
3	5	7
4	6	8

(**1;** 3,3,2)

1	2	7
3	5	8
4	6	

***

(**1;** 3,3,3)

#	3	4
1	5	7
2	6	8

(**1;** 3,3,3)

1	3	4
2	5	7
#	6	8

(**1;** 3,3,3)

1	3	4
2	#	7
5	6	8

(**1;** 3,3,3)

1	3	#
2	4	7
5	6	8

(**1;** 3,3,2)

1	3	7
2	4	8
5	6	

***

(**1;** 3,3,3)

#	3	5
1	4	6
2	7	8

(**1;** 3,3,3)

1	3	5
2	4	6
#	7	8

(**1;** 3,3,2)

1	3	5
2	4	6
7	8	

***

(**0;** 4,4,1)

#	3	5	7
1	4	6	8
2			

(**0;** 4,4)

1	3	5	7
2	4	6	8

***

附錄二 集合 SDT 與 Motzkin Path ($n=1,2,3,4,5$) 的對應關係：

$SDT_{\le 3}(1)$ 對應到 Motzkin Path 的圖形： (數量：2)

符號定義：$(D; p,q,r)$
D：表示水平骨牌的數量。
(p,q,r)：分別表示 SDT 圖形第一列、第二列以及第三列的長度。

(**1;** 2)

1	2

—

(**0;** 1,1)

1
2

↗

$SDT_{\le 3}(2)$ 對應到 Motzkin Path 的圖形： (數量：5)

(**2;** 4)

1	2	3	4

——

(**2;** 2,2)

1	2
3	4

↗↘

(**1;** 3,1)

1	3	4
2		

↗—

(**1;** 2,1,1)

1	2
3	
4	

—↗

(**0;** 2,2)

1	3
2	4

↗↗

$SDT_{\le 3}(3)$ 對應到 Motzkin Path 的圖形： (數量：13)

(**3;** 6)

1	2	3	4	5	6

— — —

(**3;** 4,2)

1	2	3	4
5	6		

— ↗ ↘

(**3;** 4,2)

1	2	5	6
3	4		

↗ ↘ —

(**3;** 2,2,2)

1	2
3	4
5	6

↗ — ↘

(**2;** 5,1)

1	3	4	5	6
2				

↗ — —

(**2;** 4,1,1)

1	2	3	4
5			
6			

— — ↗

(**2;** 4,1,1)

1	2	5	6
3			
4			

— ↗ —

(**2;** 3,3)

1	2	5
3	4	6

↗ ↘ ↗

(**2;** 3,3)

1	3	4
2	5	6

↗ ↗ ↘

(**1;** 4,2)

1	3	5	6
2	4		

↗ ↗ —

(**1;** 2,2,2)

1	2
3	5
4	6

— ↗ ↗

(**1;** 2,2,2)

1	3
2	4
5	6

↗ — ↗

(**0;** 3,3)

1	3	5
2	4	6

↗ ↗ ↗

$SDT_{\leq 3}(4)$ 對應到 Motzkin Path 的圖形： (數量：35)

(**4;** 8)

1	2	3	4	5	6	7	8

— — — —

(**4;** 6,2)

1	2	3	4	5	6
7	8				

— — ↗ ↘

(**4;** 6,2)

1	2	3	4	7	8
5	6				

— ↗ ↘ —

(**4;** 6,2)

1	2	5	6	7	8
3	4				

↗ ↘ — —

(**4;** 4,4)

$\begin{array}{cccc} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{array}$

$\nearrow\nearrow\searrow\searrow$

(**4;** 4,4)

$\begin{array}{cccc} 1 & 2 & 5 & 6 \\ 3 & 4 & 7 & 8 \end{array}$

$\nearrow\searrow\nearrow\searrow$

(**4;** 4,2,2)

$\begin{array}{cccc} 1 & 2 & 3 & 4 \\ 5 & 6 & & \\ 7 & 8 & & \end{array}$

$-\nearrow-\searrow$

(**4;** 4,2,2)

$\begin{array}{cccc} 1 & 2 & 5 & 6 \\ 3 & 4 & & \\ 7 & 8 & & \end{array}$

$\nearrow--\searrow$

(**4;** 4,2,2)

$\begin{array}{cccc} 1 & 2 & 7 & 8 \\ 3 & 4 & & \\ 5 & 6 & & \end{array}$

$\nearrow-\searrow-$

(**3;** 7,1)

$\begin{array}{ccccccc} 1 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & & & & & & \end{array}$

$\nearrow---$

(**3;** 6,1,1)

$\begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \\ 7 & & & & & \\ 8 & & & & & \end{array}$

$---\nearrow$

(**3;** 6,1,1)

$\begin{array}{cccccc} 1 & 2 & 3 & 4 & 7 & 8 \\ 5 & & & & & \\ 6 & & & & & \end{array}$

$--\nearrow-$

(**3;** 6,1,1)

$\begin{array}{cccccc} 1 & 2 & 5 & 6 & 7 & 8 \\ 3 & & & & & \\ 4 & & & & & \end{array}$

$-\nearrow--$

(**3;** 5,3)

$\begin{array}{ccccc} 1 & 2 & 5 & 7 & 8 \\ 3 & 4 & 6 & & \end{array}$

$\nearrow\searrow\nearrow-$

(**3;** 5,3)

$\begin{array}{ccccc} 1 & 3 & 4 & 5 & 6 \\ 2 & 7 & 8 & & \end{array}$

$\nearrow-\nearrow\searrow$

(**3;** 5,3)

$\begin{array}{ccccc} 1 & 3 & 4 & 7 & 8 \\ 2 & 5 & 6 & & \end{array}$

$\nearrow\nearrow\searrow-$

(**3;** 4,3,1)

$\begin{array}{cccc} 1 & 2 & 3 & 4 \\ 5 & 7 & 8 & \\ 6 & & & \end{array}$

$-\nearrow\searrow\nearrow$

(**3;** 4,3,1)

$\begin{array}{cccc} 1 & 2 & 5 & 6 \\ 3 & 7 & 8 & \\ 4 & & & \end{array}$

$-\nearrow\nearrow\searrow$

(**3;** 3,3,2)

$\begin{array}{ccc} 1 & 2 & 5 \\ 3 & 4 & 6 \\ 7 & 8 & \end{array}$

$\nearrow\searrow-\nearrow$

(**3;** 3,3,2)

$\begin{array}{ccc} 1 & 2 & 7 \\ 3 & 4 & 8 \\ 5 & 6 & \end{array}$

$\nearrow-\searrow\nearrow$

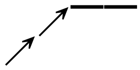
(**3;** 3,3,2)

$\begin{array}{ccc} 1 & 3 & 4 \\ 2 & 5 & 6 \\ 7 & 8 & \end{array}$

$\nearrow\nearrow-\searrow$

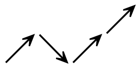
(**2;** 6,2)

1	3	5	6	7	8
2	4				



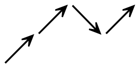
(**2;** 4,4)

1	2	5	7
3	4	6	8



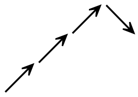
(**2;** 4,4)

1	3	4	7
2	5	6	8



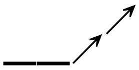
(**2;** 4,4)

1	3	5	6
2	4	7	8



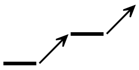
(**2;** 4,2,2)

1	2	3	4
5	7		
6	8		



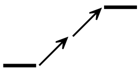
(**2;** 4,2,2)

1	2	5	6
3	7		
4	8		



(**2;** 4,2,2)

1	2	7	8
3	5		
4	6		

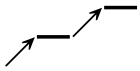


(**2;** 4,2,2)

1	3	5	6
2	4		
7	8		

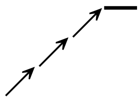
(**2;** 4,2,2)

1	3	7	8
2	4		
5	6		



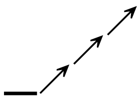
(**1;** 5,3)

1	3	5	7	8
2	4	6		



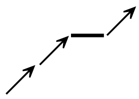
(**1;** 3,3,2)

1	2	7
3	5	8
4	6	



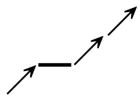
(**1;** 3,3,2)

1	3	5
2	4	6
7	8	



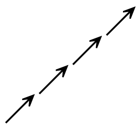
(**1;** 3,3,2)

1	3	7
2	4	8
5	6	



(**0;** 4,4)

1	3	5	7
2	4	6	8



$SDT_{\leq 3}(5)$ 對應到 Motzkin Path 的圖形： (數量：96)

(**5;** 10)

1	2	3	4	5	6	7	8	9	0

— — — — —

(**5;** 8,2)

1	2	3	4	5	6	7	8
9	0						

— — — ↗ ↘

(**5;** 8,2)

1	2	3	4	5	6	9	0
7	8						

— — ↗ ↘ —

(**5;** 8,2)

1	2	3	4	7	8	9	0
5	6						

— ↗ ↘ — —

(**5;** 8,2)

1	2	5	6	7	8	9	0
3	4						

↗ ↘ — — —

(**5;** 6,4)

1	2	3	4	5	6
7	8	9	0		

— ↗ ↗ ↘ ↘

(**5;** 6,4)

1	2	3	4	7	8
5	6	9	0		

— ↗ ↘ ↗ ↘

(**5;** 6,4)

1	2	3	4	9	0
5	6	7	8		

↗ ↗ ↘ ↘ —

(**5;** 6,4)

1	2	5	6	7	8
3	4	9	0		

↗ ↘ — ↗ ↘

(**5;** 6,4)

1	2	5	6	9	0
3	4	7	8		

↗ ↘ ↗ ↘ —

(**5;** 6,2,2)

1	2	3	4	5	6
7	8				
9	0				

— — ↗ — ↘

(**5;** 6,2,2)

1	2	3	4	7	8
5	6				
9	0				

— ↗ — — ↘

(**5;** 6,2,2)

1	2	3	4	9	0
5	6				
7	8				

— ↗ — ↘ —

(**5;** 6,2,2)

1	2	5	6	7	8
3	4				
9	0				

↗ — — — ↘

(**5;** 6,2,2)

1	2	5	6	9	0
3	4				
7	8				

↗ — — ↘ —

(**5;** 6,2,2)

1	2	7	8	9	0
3	4				
5	6				

↗ — ↘ — —

(**5;** 4,4,2)

1	2	3	4
5	6	7	8
9	0		

↗ ↗ ↘ — ↘

(**5;** 4,4,2)

1	2	3	4
5	6	9	0
7	8		

↗ ↗ — ↘ ↘

(**5;** 4,4,2)

1	2	5	6
3	4	7	8
9	0		

↗ ↘ ↗ — ↘

(**5;** 4,4,2)

1	2	5	6
3	4	9	0
7	8		

↗ — ↗ ↘ ↘

(**5;** 4,4,2)

1	2	7	8
3	4	9	0
5	6		

↗ — ↘ ↗ ↘

(**4;** 9,1)

1	3	4	5	6	7	8	9	0
2								

↗ — — — —

(**4;** 8,1,1)

1	2	3	4	5	6	7	8
9							
0							

— — — — ↗

(**4;** 8,1,1)

1	2	3	4	5	6	9	0
7							
8							

— — — ↗ —

(**4;** 8,1,1)

1	2	3	4	7	8	9	0
5							
6							

— — ↗ — —

(**4;** 8,1,1)

1	2	5	6	7	8	9	0
3							
4							

— ↗ — — —

(**4;** 7,3)

1	2	5	7	8	9	0
3	4	6				

↗ ↘ ↗ — —

(**4;** 7,3)

1	3	4	5	6	7	8
2	9	0				

↗ — — ↗ ↘

(**4;** 7,3)

1	3	4	5	6	9	0
2	7	8				

↗ — ↗ ↘ —

(**4;** 7,3)

1	3	4	7	8	9	0
2	5	6				

↗ ↗ ↘ — —

(**4;** 6,3,1)

1	2	3	4	5	6
7	9	0			
8					

— — ↗ ↘ ↗

(**4;** 6,3,1)

1	2	3	4	7	8
5	9	0			
6					

— — ↗ ↗ ↘

(**4;** 6,3,1)

1	2	3	4	9	0
5	7	8			
6					

— ↗ ↘ ↗ —

(**4;** 6,3,1)

1	2	5	6	7	8
3	9	0			
4					

— ↗ — ↗ ↘

(**4;** 6,3,1)

1	2	5	6	9	0
3	7	8			
4					

— ↗ ↗ ↘ —

(**4;** 5,5)

1	2	3	4	9
5	6	7	8	0

↗ ↗ ↘ ↘ ↗

(**4;** 5,5)

1	2	5	6	9
3	4	7	8	0

$\nearrow\searrow\nearrow\searrow\nearrow$

(**4;** 5,5)

1	2	5	7	8
3	4	6	9	0

$\nearrow\searrow\nearrow\nearrow\searrow$

(**4;** 5,5)

1	3	4	5	6
2	7	8	9	0

$\nearrow\nearrow\nearrow\searrow\searrow$

(**4;** 5,5)

1	3	4	7	8
2	5	6	9	0

$\nearrow\nearrow\searrow\nearrow\searrow$

(**4;** 5,3,2)

1	2	5	7	8
3	4	6		
9	0			

$\nearrow\searrow - - \nearrow$

(**4;** 5,3,2)

1	2	5	9	0
3	4	6		
7	8			

$\nearrow\searrow - \nearrow -$

(**4;** 5,3,2)

1	2	7	9	0
3	4	8		
5	6			

$\nearrow - \searrow\nearrow -$

(**4;** 5,3,2)

1	3	4	5	6
2	7	8		
9	0			

$\nearrow - \nearrow - \searrow$

(**4;** 5,3,2)

1	3	4	7	8
2	5	6		
9	0			

$\nearrow\nearrow - - \searrow$

(**4;** 5,3,2)

1	3	4	9	0
2	5	6		
7	8			

$\nearrow\nearrow - \searrow -$

(**4;** 4,3,3)

1	2	3	4
5	6	9	
7	8	0	

$- \nearrow - \searrow\nearrow$

(**4;** 4,3,3)

1	2	3	4
5	7	8	
6	9	0	

$- \nearrow\searrow - \nearrow$

(**4;** 4,3,3)

1	2	5	6
3	4	9	
7	8	0	

$\nearrow - - \searrow\nearrow$

(**4;** 4,3,3)

1	2	5	6
3	7	8	
4	9	0	

$- \nearrow\nearrow - \searrow$

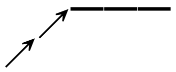
(**4;** 4,3,3)

1	2	7	8
3	4	9	
5	6	0	

$\nearrow - \searrow - \nearrow$

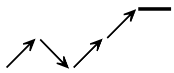
(**3;** 8,2)

1	3	5	6	7	8	9	0
2	4						



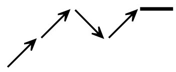
(**3;** 6,4)

1	2	5	7	9	0
3	4	6	8		



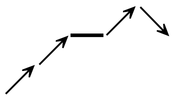
(**3;** 6,4)

1	3	4	7	9	0
2	5	6	8		



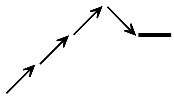
(**3;** 6,4)

1	3	5	6	7	8
2	4	9	0		



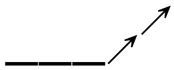
(**3;** 6,4)

1	3	5	6	9	0
2	4	7	8		



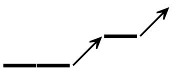
(**3;** 6,2,2)

1	2	3	4	5	6
7	9				
8	0				



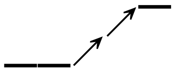
(**3;** 6,2,2)

1	2	3	4	7	8
5	9				
6	0				



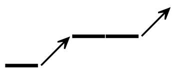
(**3;** 6,2,2)

1	2	3	4	9	0
5	7				
6	8				



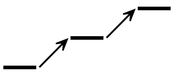
(**3;** 6,2,2)

1	2	5	6	7	8
3	9				
4	0				



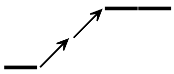
(**3;** 6,2,2)

1	2	5	6	9	0
3	7				
4	8				



(**3;** 6,2,2)

1	2	7	8	9	0
3	5				
4	6				

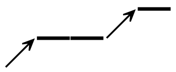


(**3;** 6,2,2)

1	3	5	6	7	8
2	4				
9	0				

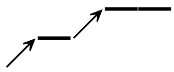
(**3;** 6,2,2)

1	3	5	6	9	0
2	4				
7	8				



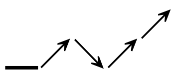
(**3;** 6,2,2)

1	3	7	8	9	0
2	4				
5	6				



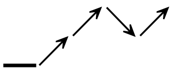
(**3;** 4,4,2)

1	2	3	4
5	7	9	0
6	8		



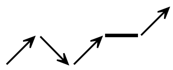
(**3;** 4,4,2)

1	2	5	6
3	7	9	0
4	8		



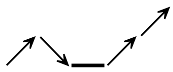
(**3;** 4,4,2)

1	2	5	7
3	4	6	8
9	0		



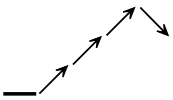
(**3;** 4,4,2)

1	2	5	9
3	4	6	0
7	8		



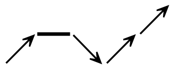
(**3;** 4,4,2)

1	2	7	8
3	5	9	0
4	6		



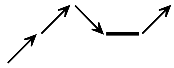
(**3;** 4,4,2)

1	2	7	9
3	4	8	0
5	6		



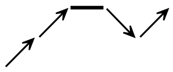
(**3;** 4,4,2)

1	3	4	7
2	5	6	8
9	0		



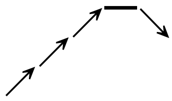
(**3;** 4,4,2)

1	3	4	9
2	5	6	0
7	8		



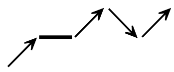
(**3;** 4,4,2)

1	3	5	6
2	4	7	8
9	0		



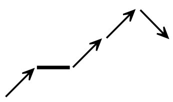
(**3;** 4,4,2)

1	3	5	6
2	4	9	0
7	8		



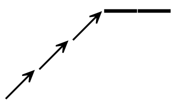
(**3;** 4,4,2)

1	3	7	8
2	4	9	0
5	6		



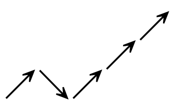
(**2;** 7,3)

1	3	5	7	8	9	0
2	4	6				



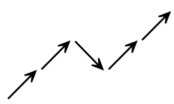
(**2;** 5,5)

1	2	5	7	9
3	4	6	8	0



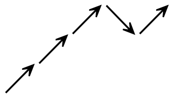
(**2;** 5,5)

1	3	4	7	9
2	5	6	8	0



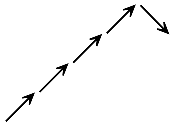
(**2;** 5,5)

1	3	5	6	9
2	4	7	8	0



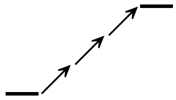
(**2;** 5,5)

1	3	5	7	8
2	4	6	9	0



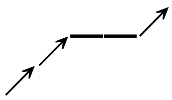
(**2;** 5,3,2)

1	2	7	9	0
3	5	8		
4	6			



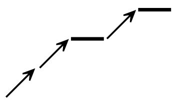
(**2;** 5,3,2)

1	3	5	7	8
2	4	6		
9	0			



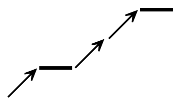
(**2;** 5,3,2)

1	3	5	9	0
2	4	6		
7	8			



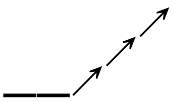
(**2;** 5,3,2)

1	3	7	9	0
2	4	8		
5	6			



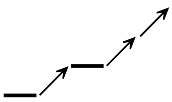
(**2;** 4,3,3)

1	2	3	4
5	7	9	
6	8	0	



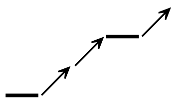
(**2;** 4,3,3)

1	2	5	6
3	7	9	
4	8	0	



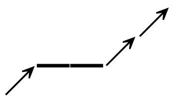
(**2;** 4,3,3)

1	2	7	8
3	5	9	
4	6	0	



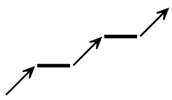
(**2;** 4,3,3)

1	3	5	6
2	4	9	
7	8	0	



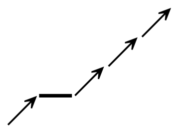
(**2;** 4,3,3)

1	3	7	8
2	4	9	
5	6	0	



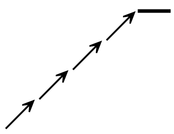
(**1;** 4,4,2)

1	3	7	9
2	4	8	0
5	6		



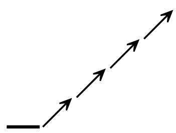
(**1;** 6,4)

1	3	5	7	9	0
2	4	6	8		



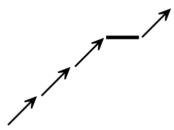
(**1;** 4,4,2)

1	2	7	9
3	5	8	0
4	6		



(**1;** 4,4,2)

1	3	5	7
2	4	6	8
9	0		

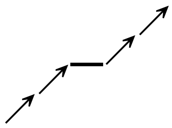


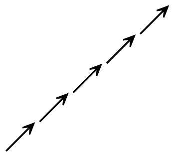
(**1;** 4,4,2)

1	3	5	9
2	4	6	0
7	8		

(**0;** 5,5)

1	3	5	7	9
2	4	6	8	0





附錄三　「骨牌與路徑對應規則」函數之 C 語言程式碼

```
int SDTMapMotzkin(char *Motzkin,char *Original){
    int Possibly[10001][5],Possibly_Count;
    int Special_Pair_List[10001][2],Special_Pair_Count;
    int Pair_Count;
    char PairStr[101],PCStr[101];
    bool Running=1;
    int i,j,k,t,a,b,c,ra,rb,rc,p,q;
    a=b=c=0;
    for(i=0;i<N;i++){
        Motzkin[i]='-';
    }
    while(Running){
        Pair_Count=Special_Pair_Count=0;
        for(i=0;i<N;i++){
            if(Motzkin[i]!='+'){
                Motzkin[i]='-';
            }
        }
        for(i=0;i<N;i++){
            if(Original[i]=='c'&&Motzkin[i]=='-'){
                t=p=q=-1;
                for(j=i-1;j>=0;j--){
                    if(Original[j]=='b'&&Motzkin[j]=='-'){
                        p=j;
                        break;
                    }
                }
                for(j=i-1;j>=0;j--){
                    if((Original[j]=='A')&&Motzkin[j]=='-'){
                        if(t==-1){
                            t=j;
                        } else {
                            q=j;
                            break;
                        }
                    }
                }
                if(t>0&&q>0&&p<q){
                    Motzkin[i]='0';
                    Special_Pair_List[Special_Pair_Count][0]=i;
                    Special_Pair_List[Special_Pair_Count][1]=t;
                    Motzkin[t]='=';
                    Special_Pair_Count++;
                } else {
                    rb=p;
                    rc=i;
                    for(j=p-1;j>=0;j--){
                        if(Original[j]=='a'&&Motzkin[j]=='-'){
                            break;
                        }
                    }
                    if(j>=0){
```

```
                    ra=j;
                    Motzkin[ra]='1';
                    Motzkin[rb]='0';
                    Motzkin[rc]='2';
                    Pair[Pair_Count++]=(PAIR){3,ra,rb,rc};
                }
            }
        }
    }
    for(i=0;i<N;i++){
        if(Original[i]=='b'&&Motzkin[i]=='-'){
            rb=i;
            for(j=i-1;j>=0;j--){
                if(Original[j]=='a'&&Motzkin[j]=='-'){
                    break;
                }
            }
            ra=j;
            if(j>=0){
                Motzkin[ra]='1';
                Motzkin[rb]='2';
                Pair[Pair_Count++]=(PAIR){2,ra,rb};
            }
        }
    }
    for(i=0;i<N;i++){
        if(Original[i]=='a'&&Motzkin[i]=='-'){
            Motzkin[i]='0';
        }
    }

    sort(Pair,Pair+Pair_Count);

    while(true){
        Possibly_Count=0;
        for(i=N-1;i>=0;i--){
            if(Motzkin[i]=='-'||Motzkin[i]=='='){
                for(j=0;j<Pair_Count;j++){
                    if(Pair[j].type==2){
                        if(Pair[j].a<i&&Pair[j].b>i){
                            Possibly[Possibly_Count][0]=1;
                            Possibly[Possibly_Count][1]=i;
                            Possibly[Possibly_Count][2]=j;
                            Possibly_Count++;
                        }
                    }
                    if(Pair[j].type==3){
                        if(Pair[j].a<i&&Pair[j].b>i){
                            Possibly[Possibly_Count][0]=4;
                            Possibly[Possibly_Count][1]=i;
                            Possibly[Possibly_Count][2]=j;
                            Possibly_Count++;
                        } else if(Pair[j].b<i&&Pair[j].c>i){
                            Possibly[Possibly_Count][0]=2;
                            Possibly[Possibly_Count][1]=i;
                            Possibly[Possibly_Count][2]=j;
                            Possibly_Count++;
                        }
                    }
                }
                for(j=Special_Pair_Count-1;j>=0;j--){
                    if(Special_Pair_List[j][1]==i){
                        Possibly[Possibly_Count][0]=3;
                        Possibly[Possibly_Count][1]=j;
                        Possibly_Count++;
                    }
                }
            }
        }
        k=-1;
        for(j=0;j<Possibly_Count;j++){
            if(Possibly[j][0]==1){
                if(k<Pair[Possibly[j][2]].b){
                    k=Pair[Possibly[j][2]].b;
                    t=j;
```

```
                        }
                    }
                    if(Possibly[j][0]==2||Possibly[j][0]==4){
                        if(k<Pair[Possibly[j][2]].c){
                            k=Pair[Possibly[j][2]].c;
                            t=j;
                        }
                    }
                    if(Possibly[j][0]==3){
                        if(k<Special_Pair_List[Possibly[j][1]][1]){
                            k=Special_Pair_List[Possibly[j][1]][1];
                            t=j;
                        }
                    }
                }
                if(k>0){
                    if(Possibly[t][0]==1||Possibly[t][0]==4){
                        Swap_Motzkin(Motzkin,Original,Possibly[t][1],Pair[Possibly[t][2]].b);
                        Pair[Possibly[t][2]].b=Possibly[t][1];
                        break;
                    }
                    if(Possibly[t][0]==2){
                        Swap_Motzkin(Motzkin,Original,Possibly[t][1],Pair[Possibly[t][2]].c);
                        Motzkin[Pair[Possibly[t][2]].b]='2';
                        Motzkin[Possibly[t][1]]='+';
                        Pair[Possibly[t][2]].type=2;
                        Original[Possibly[t][1]]='a';
                        break;
                    }
                    if(Possibly[t][0]==3){
                        Swap_Motzkin(Motzkin,Original,Special_Pair_List[Possibly[t][1]][1],
                            Special_Pair_List[Possibly[t][1]][0]);
                        Original[Special_Pair_List[Possibly[t][1]][0]]='a';
                        Special_Pair_List[Possibly[t][1]][0]=Special_Pair_List[Possibly[t][1]][1]=-1;
                        break;
                    }
                } else {
                    Running=0;
                    break;
                }
            }
        }
    for(i=N-1;i>=0;i--){
        if(Motzkin[i]=='-'||Motzkin[i]=='='){
            Motzkin[i]='1';
        } else if(Motzkin[i]=='+'){
            Motzkin[i]='0';
        }
    }
    Motzkin[N]=0;
    draw(Motzkin);
}
```

評語

本作品是計數幾何學中重要的研究課題：計算滿足給定條件的情況下如何把骨牌適當排列出來的所有可能數。作者充分利用這個量與 Motzkin path 的個數之間的對應關係，明確的算出它的答案，甚至在矩形區域有缺格的情況下也能精確地算出答案，是非常好的研究成果，以中學生程度，有此貢獻非常難能可貴，因此，評為獲獎作品。