投稿類別-自然探究

Nim 死不屈─尼姆遊戲對弈程式製作

作者：

張宸翰。縣立自強國中。八年六班
王聿典。縣立自強國中。八年六班

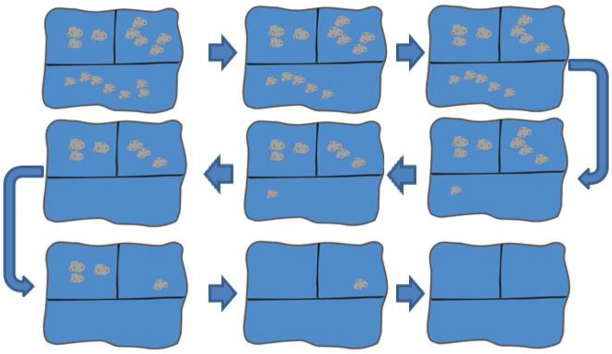
指導老師：

呂柏辰 老師
陳禹翔 老師

壹、 前言

一、研究動機

在獨立研究課時，老師讓我們玩一個小遊戲——三五七石頭棋，此遊戲會有三堆石頭，一堆三顆、一堆五顆、一堆七顆，玩家在自己的回合拿多少棋子都可以，但是不可越堆拿，而拿到最後一顆石頭的人就輸了，下圖是我們對弈的過程，如(圖一)。



（圖一）

我們對此遊戲產生強烈的興趣，便上網搜尋相關的文獻，發現無論如何分堆，顆數為何，都可以通稱為「尼姆遊戲 Nim」，而我們想要找出這種遊戲的必勝法，並更進一步的做出可以和玩家對弈的電腦程式，因此才會有這次的研究。

二、研究目的

1. 研究三五七石頭棋有無必勝法。
2. 做出三五七石頭棋的程式。
3. 整理尼姆遊藝遊戲的必勝法，並做出任意棋數的程式。

三、研究流程

貳、 正文

一、文獻探討

(一) 三五七石頭棋規則

1. 劃分三個區域，一區 3 顆一堆、一區 5 顆一堆、一區 7 顆一堆
2. 兩人輪流取石
3. 不限顆數(至少拿一顆)
4. 不可跨區拿
5. 拿到最後一顆的人輸

(二) 尼姆遊戲之必勝法

與三五七石頭棋相似的尼姆遊戲（ n 堆 a_1、 … 、a_n 顆）之必勝方法，是將每列棋子數轉換為二進位並進行 XOR 運算，如果結果每位數都是0，則是「安全殘局」；若有1出現，則是「不安全殘局」。玩家要做的就是將不安全局面變成安全局面。這可以透過取走適當數量的銅板，讓每位數相加結果變成0，變為安全殘局。當局面為安全殘局時，不管如何取棋都會變成不安全殘局。因此一旦取得了安

全殘局，便可在接續棋局中把握棋局的主控權。依照上述必勝規則取棋，直到只剩一堆的棋子數超過1，此時的策略便是針對超過一個棋子的那堆棋子取棋，使留下來的每一排都只有一個棋子，並要讓留下來的排數是奇數個。

(三) XOR（Exclusive OR；異或；互斥或）

XOR（$\oplus$）是對兩個運算元的一種邏輯分析類型。與一般的邏輯或不同，當兩兩數值相同時為否，而數值不同時為真。在二進制中，就是將運算元相加但不進位，

例如：$101 \oplus 110 = 011$，$\begin{array}{r} 1\ 0\ 1 \\ \oplus\ 1\ 1\ 0 \\ \hline 0\ 1\ 1 \end{array}$

二、三五七石頭棋必勝法

研究的一開始，我們想自己先研究看看三五七頭棋的必勝法是什麼，並嘗試做出製作上較方便，邏輯較簡單的三五七石頭棋程式，因此我們透過自己對弈的經驗，將三五七石頭棋的必勝組合分為八種，並製作程式。

(一) 基礎組合

我們先將棋數以數字表示，每一數字表示該堆棋子數，以$(a、b、c)$表示。

1. 在棋數為$(1、0、0)$的狀況下，先手會拿到最後一顆棋子，在製造出$(n、0、0)$且$n > 1$的情況，對手只需拿走$n - 1$顆棋子即可獲勝；若是$(1、n、0)$且$n > 0$的情況時，對手只需拿走n即可獲勝。
2. 在棋數為$(1、1、1)$的狀況下，先手會拿到最後一顆棋子。因此製造出$(1、1、n)$且$n > 1$的情況時，對手只需拿走$n - 1$顆棋子即可。

小結 1：(1)$(n、0、0)$且$n > 1$、(2)$(1、n、0)$且$n > 0$和(3)$(1、1、n)$且$n > 1$的情形先手必勝，因此要避免製造出這些情況。

(二) 雙數組合

1. 在棋數為$(2、2、0)$的情況時，會有兩種發展方式。

（1）$(2、2、0) \rightarrow (2、0、0)$

（2）$(2、2、0) \rightarrow (2、1、0)$

情況 1 符合上述小結(1)的情況，先手輸。

情況 2 符合上述小結(2)的情況，先手輸。

2. 因此我們發現在（2、2、0）的情況中，先手不管如何取棋都會輸。

在棋數為$(3、3、0)$的情況時，會有三種發展方式。

（1）$(3、3、0) \rightarrow (3、0、0)$

（2）$(3、3、0) \rightarrow (3、1、0)$

（3）$(3、3、0) \rightarrow (3、2、0)$

情況 1 符合上述小結(1)的情況，先手輸。

情況 2 符合上述小結(2)的情況，先手輸。

情況 3 符合上述$(2、2、0)$的情況，先手輸。

因此我們發現在$(3、3、0)$的情況中，先手不管如何取棋都會輸。

由上述$(2、2、0)$及$(3、3、0)$的討論，我們發現在$(n、n、0)$且$n > 2$的情況下，假設對手拿 a 顆棋子，且 $a \geq n - 2$，我們便另一堆拿 a 顆棋子，輪流取棋直到情況變為$(n、0、0)$且$n > 1$、（$1、n、0$） 且$n > 0$或$(2、2、0)$，再用上述之方法應對即可。

小結 2：$(n、n、0)$且$n > 2$的情況後手必勝，因此要避免製造出(4)$(n_1、n_2、0)$且$n_1 \neq n_2$和(5)$(n_1、n_1、n_2)$且$n_1 > 1, n_2 > 0$的情況。

(三) 等差數列組

1. 在棋數為$(1、2、3)$的情況時，會有三種發展方式。

   (1)$(1、2、3) \to (n_1、n_2、0)$，$n_1 \neq n_2$

   (2)$(1、2、3) \to (n_1、n_1、n_2)$, $n_1 > 1$, $n_2 > 0$

   (3)$(1、2、3) \to (1、1、n)$且$n > 1$

   三種情況先手皆輸，因此在$(1、2、3)$的情形中先手必輸；

   因此$(1、2、n)\ n > 3, (1、n、3)\ n > 2, (n、2、3)\ n > 1$的情形中後手必輸。

2. 在棋數為（3、4、5）的情況時，取 b、c 堆會造成(2)、(3)、(4)、(5)和$(1、2、n)\ n > 3, (1、n、3)\ n > 2, (n、2、3)\ n > 1$的情形，也就是拿 b、c 堆都會輸，所以在 a 堆拿 2 顆棋子，迫使對手拿 b、c 堆：

   $(3、4、5) \to (1、4、5) \to$

   $(1、2、n)\ n > 3, (1、n、3)\ n > 2, (n、2、3)\ n > 1$

   在此也發現$(1、4、5)$的情況後手勝。

3. 在$(2、4、6)$的情況時，不能取 b 堆的棋子，避免造成$(1、2、n)\ n > 3, (1、n、3)\ n > 2, (n、2、3)\ n > 1$的情形；另兩堆拿$> 1$顆棋子則會造成(4)、(5)和$(1、2、n)\ n > 3, (1、n、3)\ \mathrm{n} > 2, (n、2、3)\ n > 1$的情形，引此只有兩種拿法：

   （1）$(2、4、6) \to (1、4、6) \to (1、4、5)$

   （2）$(2、4、6) \to (2、4、5) \to (1、4、5)$

此兩種拿法最後都會導致先手迫不得已把棋局製造成死局，所以在$(2、4、6)$的情況後手獲勝。依上述推論$(2、4、n) n > 6, (2、n、6)\ n > 4, (n、4、6)\ n > 2$，而另一項大於剩下的那個數，則先手獲勝。

小結：$(1、2、3)$、$(2、4、6)$和$(1、4、5)$的情況後手必勝，因此要避免製造出 (6)$(1、2、n)\ n > 3, (1、n、3)\ n > 2, (n、2、3) n > 1$ (7)$(1、4、n)\ n > 5, (1、n、5)\ n > 4, (n、4、5) n > 1$和 (8)$(2、4、n) n > 6, (2、n、6)\ n > 4, (n、4、6) n > 2$的情況。

(四) 357 石頭棋必勝法

我們歸納出了以下應對方法：

遇到的情況	需製造出的情況
(1)$(n、0、0)$ 且 $n > 1$	$(1、0、0)$
(2)$(1、n、0)$ 且 $n > 0$	$(1、0、0)$
(3)$(1、1、n)$ 且 $n > 1$	$(1、1、1)$
(4)$(n_1、n_2、0)$ 且 $n_1 \neq n_2$	$(n、n、0)$
(5)$(n_1、n_1、n_2)$ 且 $n_1 > 1, n_2 > 0$	$(n、n、0)$
(6)$(1、2、n)\ n > 3$, $(1、n、3)\ n > 2$, $(n、2、3)\ n > 1$	$(1、2、3)$
(7)$(1、4、n)\ n > 5$, $(1、n、5)\ n > 4$, $(n、4、5)\ n > 1$	$(1、4、5)$
(8)$(2、4、n)\ n > 6$, $(2、n、6)\ n > 4$, $(n、4、6)\ n > 2$	$(2、4、6)$

而先手第一步只能拿一顆棋子才能規避必輸的狀況，所以會有三種取法。

（1）$(3、5、7) \rightarrow (2、5、7)$

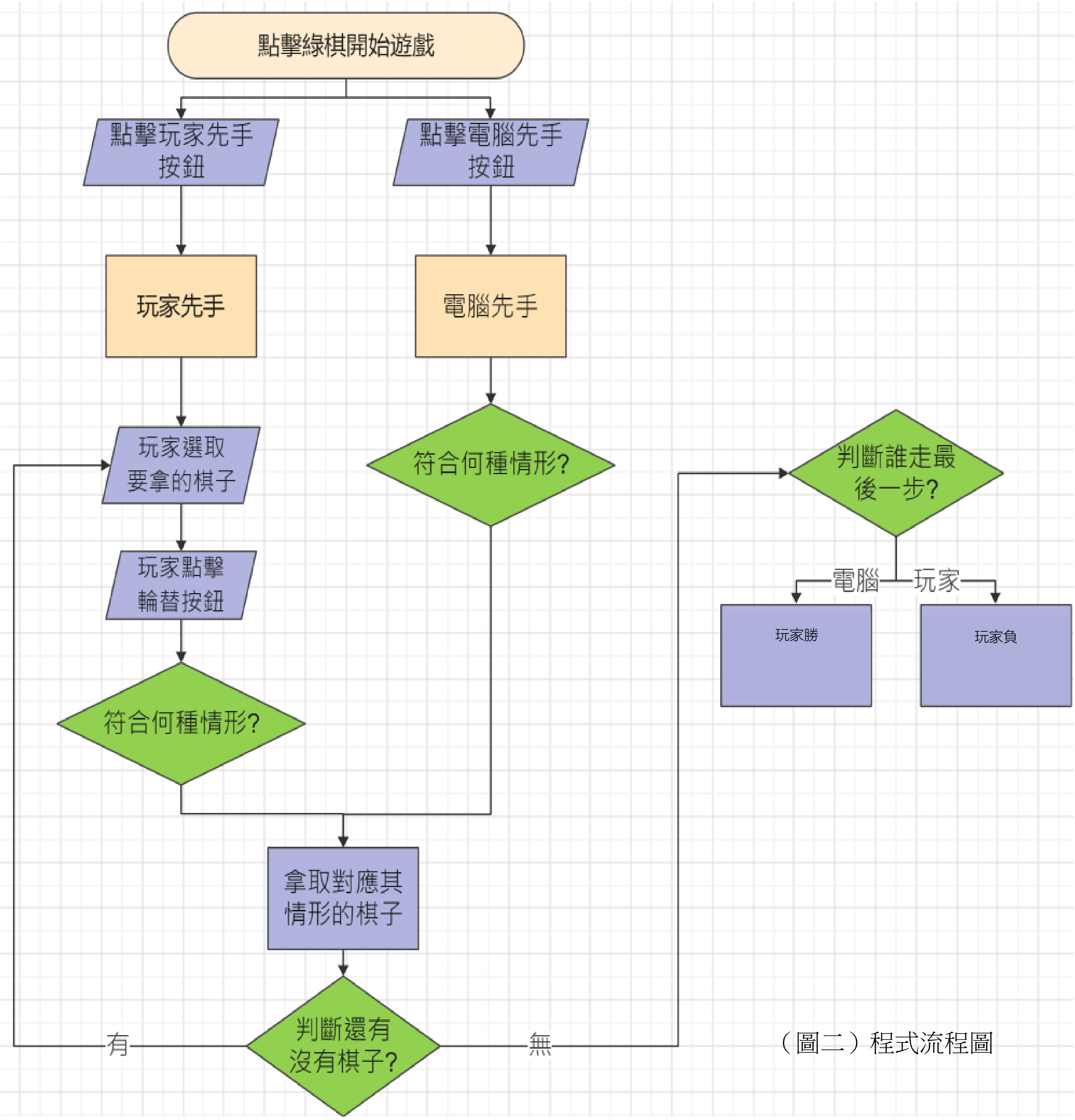
（2）$(3、5、7) \rightarrow (3、4、7)$

（3）$(3、5、7) \rightarrow (3、5、6)$

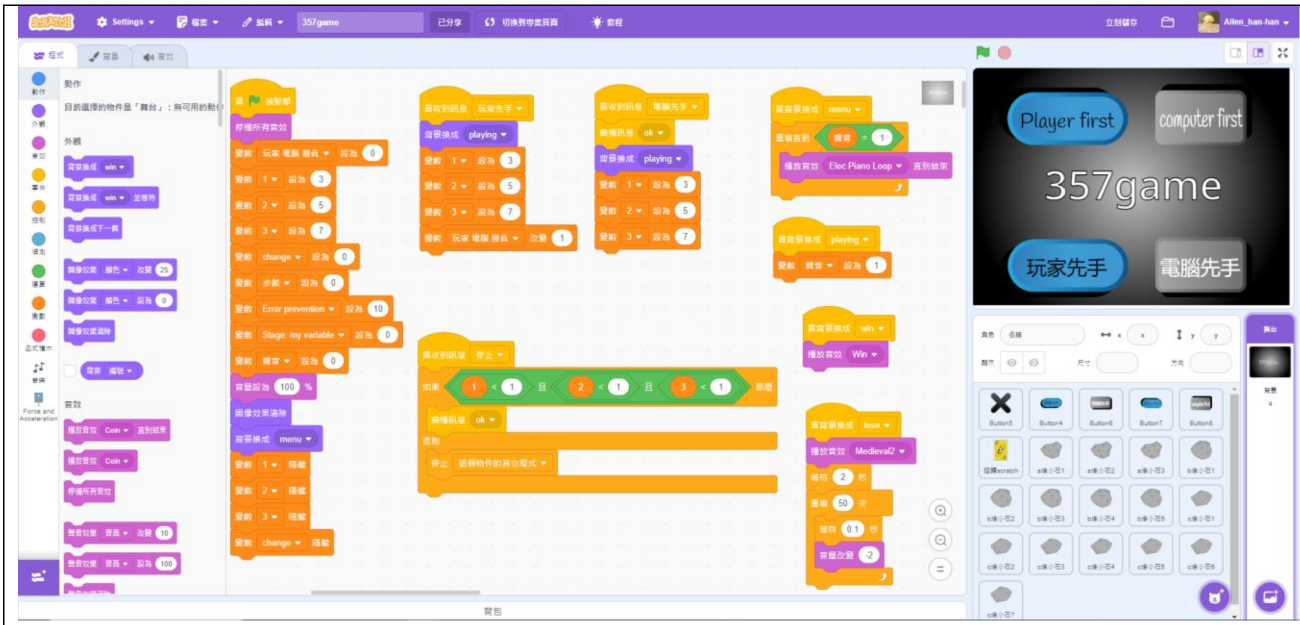
先手拿一顆子，後手這時如何取子都會陷入以上八種狀況。之後再以上述方式致勝。

三五七石頭棋程式製作

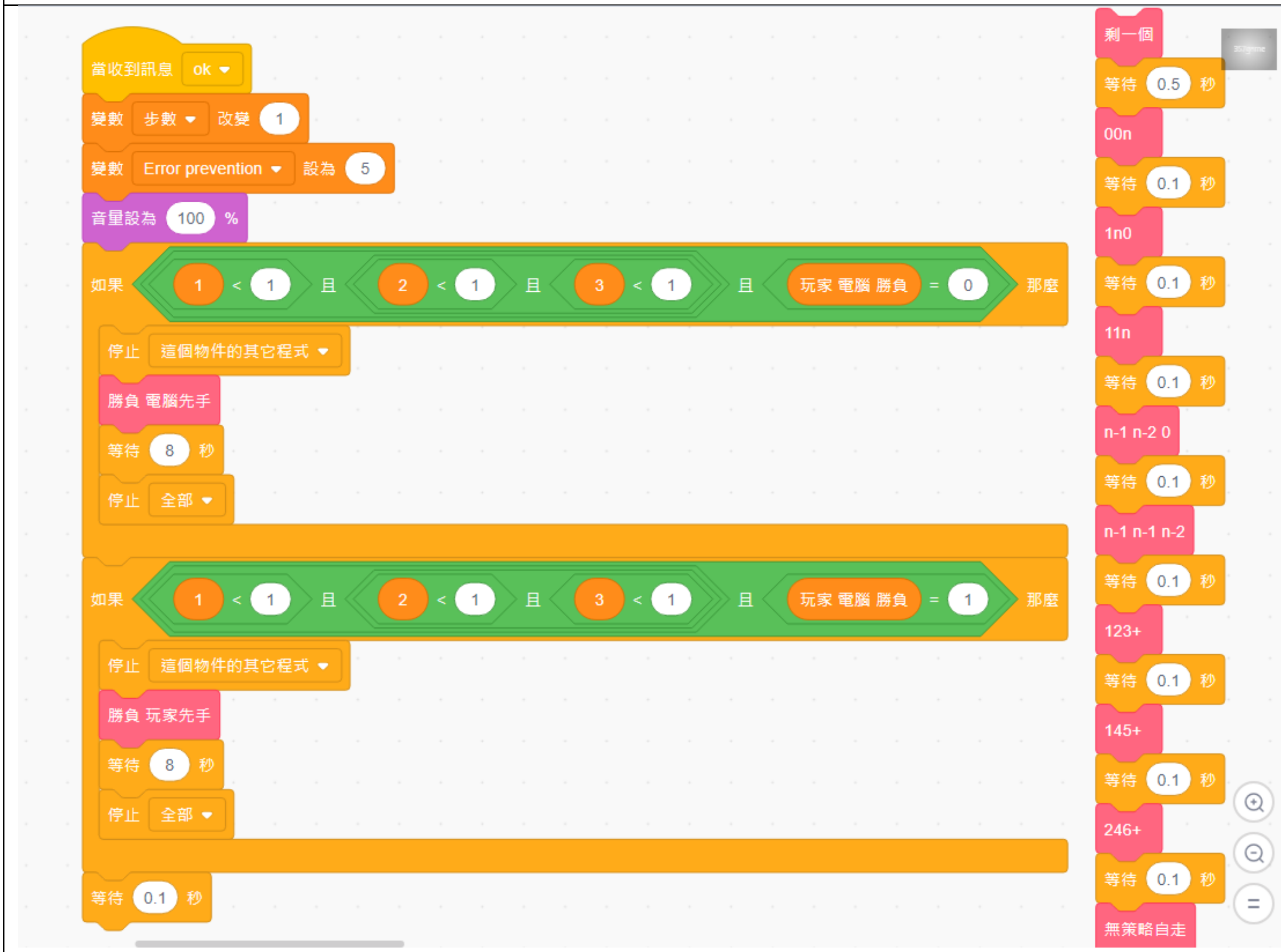
（一）以下是主要程式部分



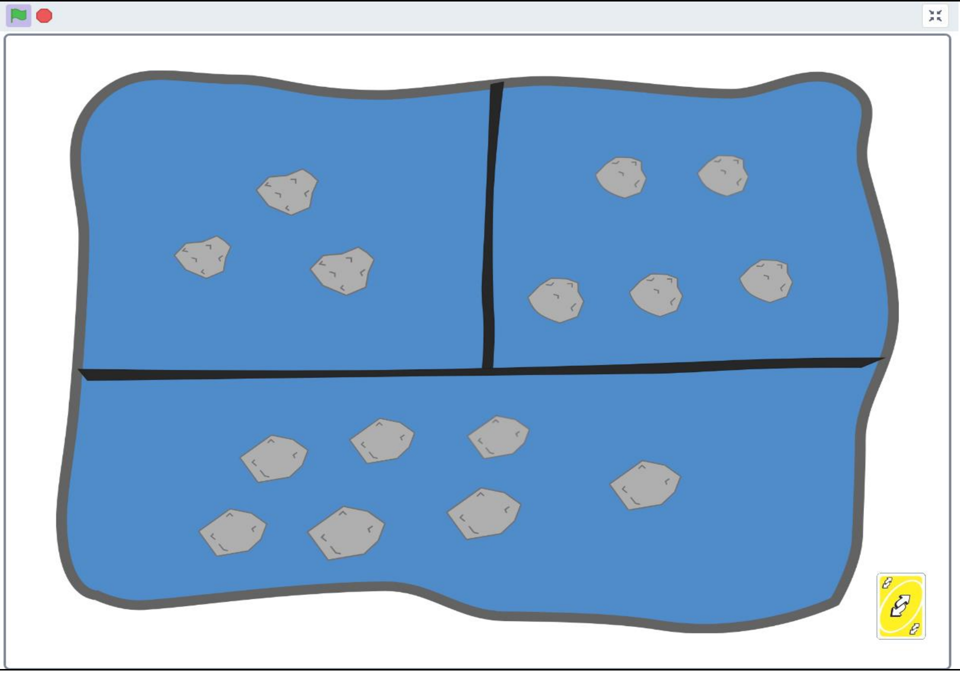
（圖二）程式流程圖



（圖三）初始化的變數設定和事件方塊。

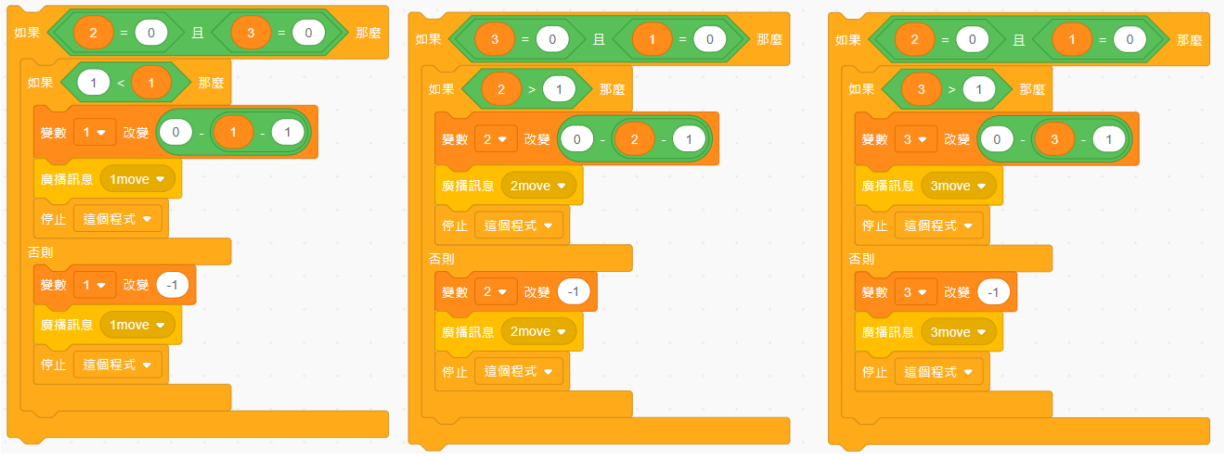


（圖四）電腦判斷程式主幹。



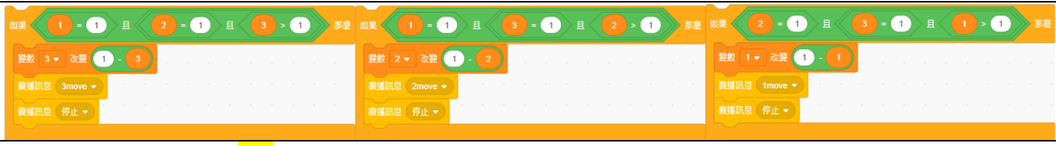
（圖五） 此為程式遊玩頁面。

（二）以下是函數方塊的程式

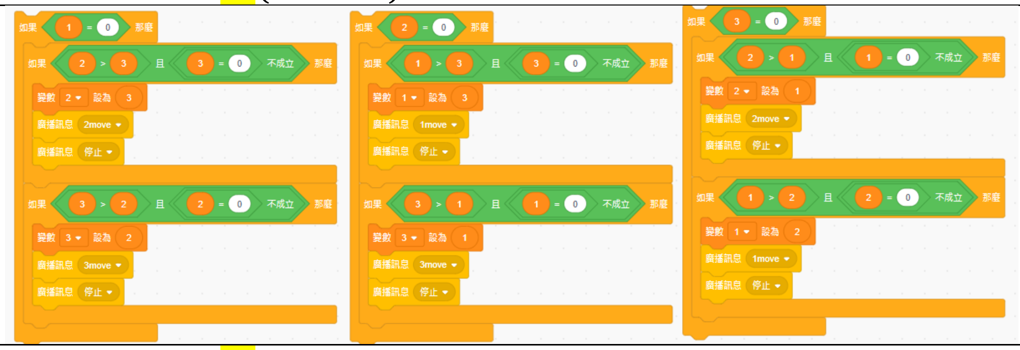


1.此組程式表示電腦在(1)$(n、0、0)$的狀況時會從n的堆取$n-1$。

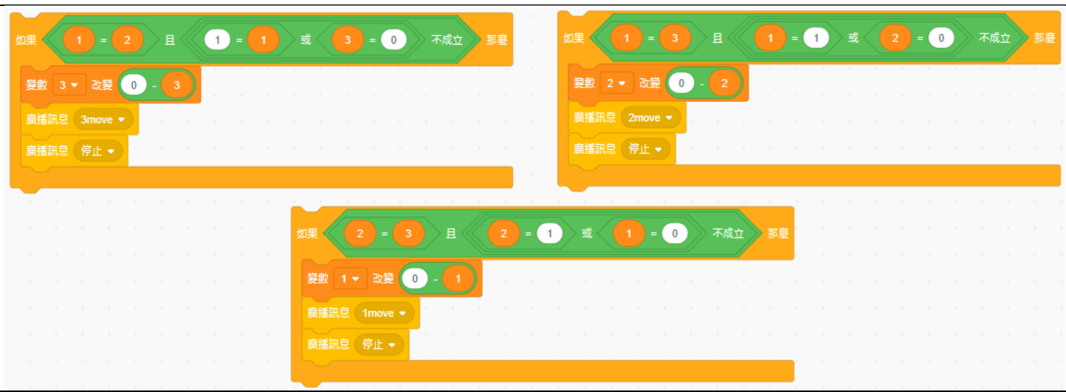
2.此組程式表示電腦在(2)$(1、n、0)$的狀況時會從n的堆取n。



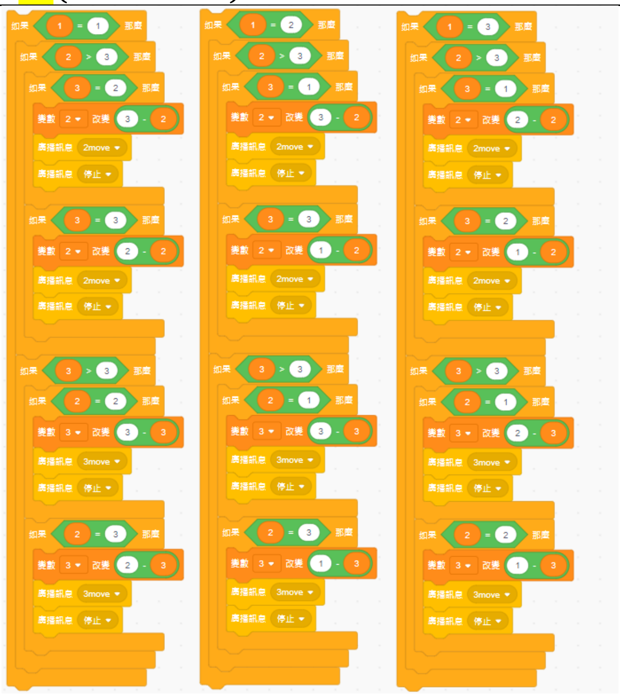
3.此組程式表示電腦在(3)$(1、1、n)$的狀況時會從n的堆取$n-1$。



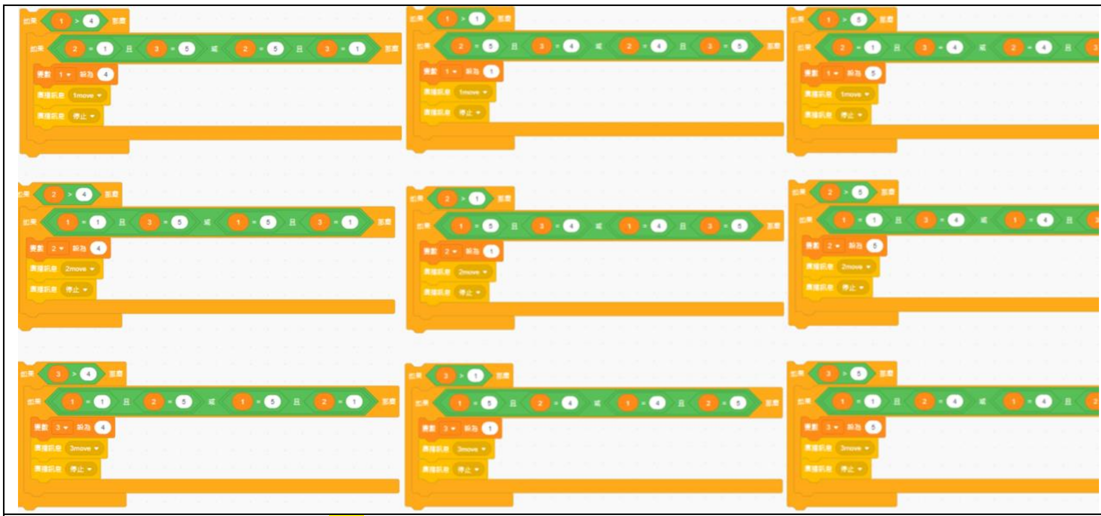
4.此組程式表示電腦在(4)$(n_1、n_2、0)$的狀況時，若$n_1 > n_2$則會從n_1的堆取$n_1 - n_2$；若$n_2 > n_1$則會從n_2的堆取$n_2 - n_1$。



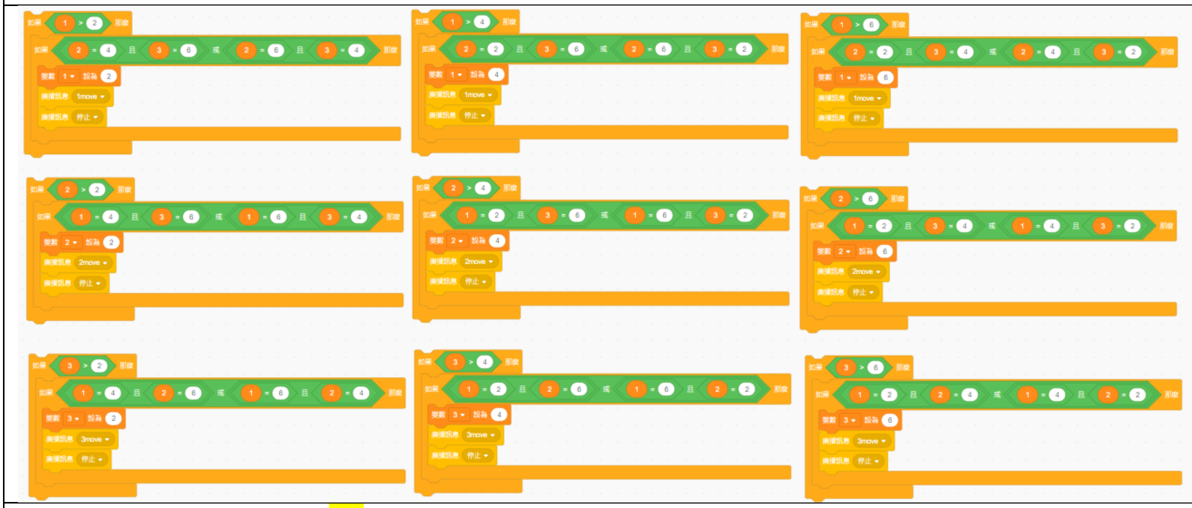
5.此組程式表示電腦在(5)$(n_1、n_1、n_2)$的狀況時，會取走n_2。



6.此組程式表示電腦在(6)$(1、2、n),\ n > 3$的狀況時，會從n的堆取$n-3$；在$(1、n、3),\ n > 2$的狀況時，會從n的堆取$n-2$；在$(n、2、3),\ n > 1$的狀況時，會從n的堆取$n-1$。



7.此組程式表示電腦在(7)$(1、4、n)$, $n > 5$的狀況時，會從n的堆取$n-5$；在$(1、n、5)$, $n > 4$的狀況時，會從n的堆取$n-4$；在$(n、4、5)$, $n > 1$的狀況時，會從n的堆取$n-1$。

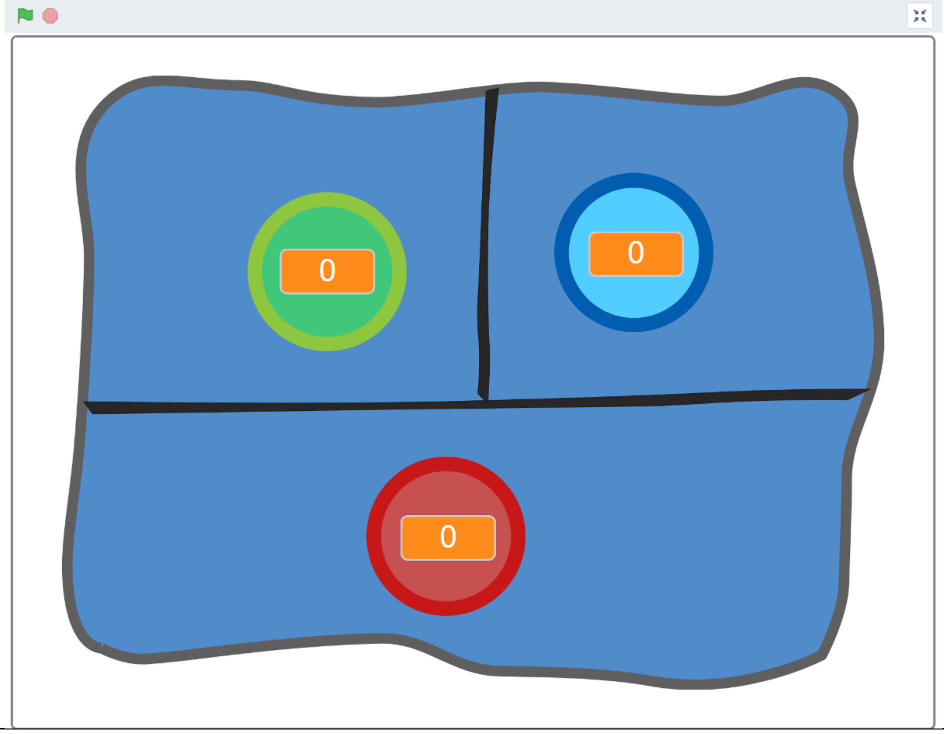


8.此組程式表示電腦在(8)$(2、4、n)$, $n > 6$的狀況時，會從n的堆取$n-6$；在$(2、n、6)$, $n > 4$的狀況時，會從n的堆取$n-4$；在$(n、4、6)$, $n > 2$的狀況時，會從n的堆取$n-2$。

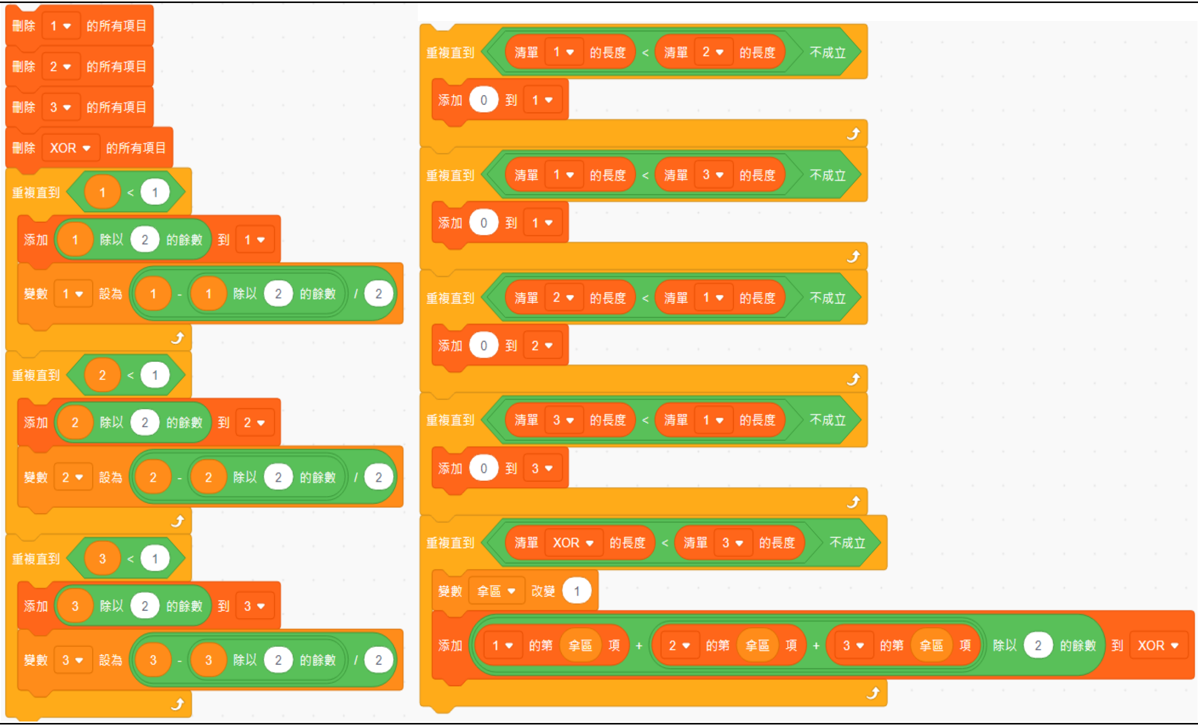
9.此組程式表示電腦在無法造成必勝棋局時，會隨機從其中一堆取 1。會這樣設計的原因，其一為必勝第一步必取任一堆的一顆；其二為若玩家知道必勝法，則可只取 1 以拖慢對手，加大其錯誤的可能性。

三、製作任意棋數程式

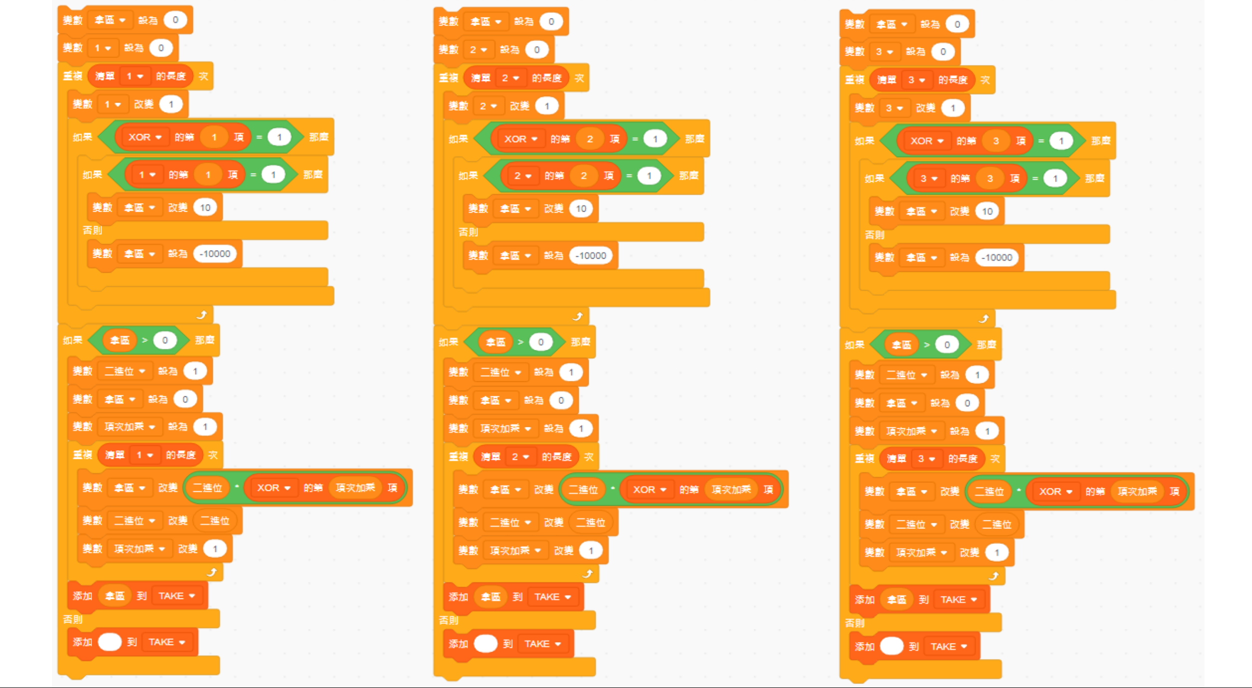
我們利用尼姆遊戲的必勝法，製作三堆、任意棋數的電腦程式，此程式邏輯較難，但是程式量較少，我決定直接用變數當作每堆棋子，以避免棋數過多棋子互相覆蓋和方便程式製作。



1.此為程式遊玩頁面。



2.此組程式是將玩家輸入的十進位數字換為二進位。



3.此組程式是在將三堆棋數換成二進為之後的結果，進行 XOR 運算。

4.此組程式是判斷電腦要如何取棋。

參、 研究結果與討論

一、我們推導出三五七石頭棋的八個必勝棋型，並得出三五七石頭棋必勝法。
二、我們利用三五七石頭棋的必勝法製作出可與玩家對弈的三五七石頭棋程式。
三、我們利用尼姆遊戲的必勝法製作出可與玩家對弈的三堆、任意顆數程式。

肆、 引注資料

一、 拈 (Nim)

https://calculus.math.nycu.edu.tw/maple/Site/carnival/game/202.htm

二、 拈及其各種變形遊戲

https://episte.math.ntu.edu.tw/articles/mm/mm_03_2_02/index.html

三、 三五七石頭棋

https://scratch.mit.edu/projects/1072726777