

中華民國第 55 屆中小學科學展覽會 作品說明書

國中組 數學科

030420

1 A 2 B 之我要贏電腦

學校名稱：臺中市立豐東國民中學

作者： 國二 張琇婷 國二 張詠儀	指導老師： 梁健彬 陳愉婷
---------------------------------	-----------------------------

關鍵詞：推理、數學遊戲、排列組合

摘要

從昌爸工作坊-數學遊戲中，我們發現了猜數字類型的遊戲，為了找出最少次的答題步驟，我們開始此次的研究。在反覆的計算及討論中，我們發現了其中的技巧、探討出如何過關的最佳方法，並延伸出自己的出題方式反制對手（電腦），為我方爭取更多的答題時間。

這次的研究運用了推理及簡單的排列組合，透過對手給予我方的提示，逐一的解題，並用此排列概念，研發出一套屬於自己的解題方式，並用此方法，繼續探討其他位數（二位數、三位數、五位數等）的解題模式。

此件科展作品注重於如何快速的答題及牽制住敵方，且破除網路留言板中所流傳的「絕對步數」，在七步內能夠完成，期望能夠保證在七步（或是更少）內一定完成，讓我方獲得勝利。

壹、研究動機

在一次的數學課程中，老師讓我們看了許多與數學有關的影片及節目，甚至在昌爸工作坊的數學遊戲中，都有此種遊戲。老師跟我們說只要能找到最快速的解決方式，而不是像無頭蒼蠅般地胡亂瞎猜，將會有意想不到的小禮物，於是，我們下定決心做這份研究，找到最快速的解題方式，勢必要獲得老師的禮物。

貳、研究目的

- 一、如何在最少的次數內找到正確的組合排列，並找出「必勝方法」，讓有興趣的人可以沉浸在勝利的成就感。
- 二、如何讓電腦猜的次數保持最多的答案組合。
- 三、了解電腦程式解題過程。
- 四、比較多種不同情況的結果。

參、研究設備及器材

一、紙

二、筆

三、電腦

肆、研究過程及方法

一、介紹遊戲規則：

雙方各出 4 個不重複的數字 (0~9)。A 為數字、位置皆對；B 為數字對但位置錯。依照對方猜的數字給予正確的組合 (EX：1A1B、2B……)，直至有一方猜出正確答案即結束。

二、猜題必勝大全：

我們發現電腦大多猜 7 次便能解出正解，所以我們必須在 7 次內先找出電腦的答案。

(一) 亂猜的情況 (先設正確答案為 5902)

次數	亂猜的數字				對手答覆我們的組合
(1)	1	3	6	9	1B
(2)	2	5	8	4	2B
(3)	3	9	4	6	1A
(4)	6	4	8	9	1B
(5)	5	7	4	0	1A1B
(6)	7	0	6	4	1A
(7)	5	0	4	8	1A1B
(7)	5	7	3	6	1B
...
(X)	5	9	0	2	4A

◎由此可知亂猜必須猜很久，甚至是無法猜到對手 (電腦) 的數字，因為這樣，我們決定要用推理的方式猜出。

(二) 我們的方法-邏輯式推理 (先設正確答案為 5902)

步驟一：先猜有規律性的數字 (EX：1234、5678、1357、2468……)，記住！數字不能夠重複，連續猜兩次。

步驟二：依照前面的方法，我們大致整理出敵方 (電腦) 的思緒及其組合的方式和結構。

範例：

次數	我們推理的數字				對手答覆我們的組合	我們的推理思考
(1)	1	2	3	4	1B	
(2)	5	6	7	8	1A	因為答案總共有 4 個數字，而 1234、5678 分別都有一個數字，而數字中只剩 9 和 0 這兩個數字，由此可知一定有 9、0 兩個數字，接著就用 9、0 與其配對。
(3)	5	0	9	3	1A2B	1A2B 中一定有 9、0，由此可知 5、3 其中一個是對的。
(4)	5	4	0	9	2A1B	(1)由前兩個提示可推出，5 應該是 A，但還不能真正確定。 (2)9、0 其中一個數字是對的，沒有 3、4(3 是由第三個及第四個提示可知)。
(5)	5	9	0	1	3A	可知沒有 1(因為一定有 5、9、0 這三個數字)。
(6)	5	9	0	2	4A	

◎根據上述推論結果，推理的方式會比亂猜還要來的快很多，接下來，我們將介紹我們的推理方法及模式。

(三) 推理方式總結：

步驟一：先猜規律性的數字(不可重複)，之後，由前兩次的組合就可以先知道是否有剩下的兩個數字(如範例的第一次推理)，也可以從兩次的組合中先推出一些線索。

範例：1234(1B)、5678(1A1B)

→可以先設定 5678 中的某個位置是對的，對等下的推理有一定的幫助

步驟二：依序前面的方式慢慢拆解其組合，由前面的提示可以知道數字的位置及其組合(如範例的推理介紹)

◎由推理方法的模式可以快速的猜出敵方(電腦)的數字組合，我們目前希望能夠在 7 步之內完成，打破電腦的絕對步數。

(四) 如果以排列組合來解釋的話：

千位數	可能結果：0、1、2、3、4、5、6、7、8、9
百位數	可能結果：0、1、2、3、4、5、6、7、8、9
十位數	可能結果：0、1、2、3、4、5、6、7、8、9
個位數	可能結果：0、1、2、3、4、5、6、7、8、9

→全部有 5040(10*9*8*7)種不同的組合。

以上一個範例組合來說明：

次數	對手給予的提示	我們的想法
(1)	1234→1B	表有一個數字是對的，其總組合還是有五千零四十種不同的組合(因為還未確定其數字)。
(2)	5678→1A	此時的組合還是有五千零四十種，因無法確定究竟是哪個數字正確。
(3)	5093→1A2B	此時可去除 3 這個數字(因為可確定有 9 跟 0。)，此時組合變成三千零二十四(9*8*7*6)種。

(4)	5409→2A1B	可確定 5 是 A(由上兩種提示推理出)，且沒有 4 這個數字，此時組合有一百二十種(1*6*5*4)不同組合方式。
(5)	5901→3A	可確定無 1 這個數字，由此提示可知 5、9、0 的位置，此時只剩下 5902 此組合(因第一個提示是 1、2、3、4 中有一個數字是對的)。
(6)	5092→4A	由上面的組合模式可知要在此步即猜出的機率為 100%。

→由此方式可快速的達出對手的組合，再搭配我們的推理方式，會更快速的答出組合。

三、如何使電腦猜得比較久：

- (一) 如果一開始就先假設好我們的數字，那麼敵方就能以前面的方式快速的猜出來，所以我們將研究如何使我們的數字難以猜出來。此方式只研究敵方為電腦這種程式固定的敵方，且不討論人對人的方式，故跟第一個方式並不會起衝突。

範例一：假設好了數字的情況(設數字為 5902)

次數	電腦猜的數字	我們給予的組合
(1)	0123	2B
(2)	1346	0
(3)	2057	3B
(4)	5209	2A2B
(5)	5902	4A

◎電腦一開始可能不會依照規律猜，但電腦的程式是固定的，所以電腦在幾次的一問一答中就可猜出，就如上述的實驗中，電腦因為猜出了一組 0 的數字組合，所以五次就猜出我方的數字，這對我方來說為不利，所以接下來將研究如何使敵方猜比較久。

範例二：一開始並未假設數字的情況下

次數	電腦猜的數字	我們給予的組合	我們的思考邏輯
(1)	0617	1A1B	一開始就跟敵方答 1A1B，此時自己尚未確定答案。
(2)	0319	1B	答 1B，此時尚未確定答案。
(3)	4068	2B	由此組合及前兩組給的提示可以拼出「6947」，這就是我們的答案，再依照這個答案給出 2B 這個提示。
(4)	3687	1A1B	
(5)	2807	1A	
(6)	6497	2A2B	
(7)	6947	4A	

◎因為一開始我們並未確定自己的答案，在給敵方提示的同時，自己也正用這些提示拼湊出我方的答案，在這過程中，敵方須猜的次數就會增加，這對我方將有更多的時間去猜出敵方的數字，所以這是一個制敵的好方法。

範例三：如果一開始給的提示不妥，將會促使敵方更快答出，以下將示範不妥的提示

次數	電腦猜的數字	我們給予的組合	我們的思考邏輯
(1)	1234	1A	
(2)	5678	1B	
(3)	8930	2B	由此步電腦可確定沒有 8、3
(4)	9704	2B	由此步電腦可確定沒有 7、4
(5)	1096	2B	由此步電腦可確定有 2、5
(6)	0259	4A	

◎由上述情況來看，因為一開始的提示就告知敵方我們有確定的數字，儘管後面一直 2B 可稍微拖延時間，但這種方式固定 6 步內即可猜到我方的數字，這對我方是不利的，故此方法是不利於我方的拖延戰術。

四、了解電腦程式的解題思維：

為了讓我方了解電腦的想法及邏輯，我們在網路上看到一個詳細的猜數字程式碼，並深入研究，希望能達到知己知彼，百戰百勝的效果。

(一) 大略猜題技巧：

解題步驟	大概作法流程
步驟一	將四個不連續不相同的數字，共 5040 種組合，一一列舉出來。
步驟二	利用亂數，取出這 5040 種組合中的其中一組(第一次猜測)。
步驟三	將第一次所猜的數字與我方的提示中，從 5040 種組合中比對，將符合的排列留下，不符合的排列刪除。
步驟四	將陣列排序。
步驟五	將剩下的組合的排列利用亂數取出一組(第二次猜測)。
步驟六	將剩下符合的陣列元素與第二次猜測的數字比對，符合的留下，不符合的陣列元素則刪除。
...	(重複第步驟五、六步驟，即可讓電腦解出答案)。

→結論：由上述可知，電腦也是利用排列組合的方式一一篩選，由前面的說明中可知到了第四步即可將 5040 種不同的組合縮減至 10 種內（或是更少），故，電腦大多可在七次內完成答題。（大多是在第五、六步內答出）

*程式在附錄一

五、比較多種不同情況的結果

情況一	得 1B&1B	1. 得第一個 1B 時剩 1440 個組合 (此四位數的位置皆錯) 2. 得第二個 1B 時剩 216 個組合 (此四位數的位置皆錯) 3. 可確定有另兩個數字	剩 216 種組合
情況二	得 1B&2B	1. 得 1B 時為 1440 個組合 (此四位數的位置皆錯) 2. 得 2B 時為 504 個組合 (此四位數的位置皆錯) 3. 還不能確定第四個數字	剩 504 種組合
情況三	得 1B&3B	1. 得 1B 時為 1440 個組合 (此四位數的位置皆錯) 2. 得 3B 時為 132 個組合 (已確定無另兩個數字)	剩 132 種組合
情況四	得 1B&1A	1. 得 1B 時為 1440 個組合 (此四位數的位置皆錯) 2. 得 1A 時為 72 個組合 3. 可確定有另兩個數字	剩 72 種組合

情況 五	得 1B&2A	1. 得 1B 時為 1440 個組合 (此四位數的位置皆錯) 2. 得 2A 時為 72 個組合 3. 無法確定第四個數字	剩 72 種組合
情況 六	得 1B&3A	1. 得 1B 時 1440 組合 (此四位數的位置皆錯) 2. 得 3A 時為 12 個組合 (已確定無另兩個數字)	剩 12 種組合
情況 七	得 2B&0	1. 得 2B 時為 1260 組合 (此四位數的位置皆錯) 2. 得 0 時為 84 個組合 3. . 可以確定有另兩個數字	剩 84 種組合
情況 八	得 2B&2B	1. 得第一個 2B 時為 1260 個組合 (此四位數的位置皆錯) 2. 得第二個 2B 時為 294 個組合 (已確定無另兩個數字)	剩 294 種組合
情況 九	得 2B&1A	1. 得 2B 時為 1260 個組合 (此四位數的位置皆錯) 2. 得 1A 時為 168 個組合 3. 無法確定第四個數字	剩 168 種組合
情況 十	得 2B&2A	1. 得 2B 時為 1260 組合 (此四位數的位置皆錯) 2. 得 2A 時為 42 個組合 (已確定無另兩個數字)	剩 42 種組合

情況 十一	得 3B&0	1. 得 3B 時為 264 個組合 (此四位數的位置皆錯) 2. 得 0 時為 88 個組合 3. 無法確定第四個數字	剩 88 種組合
情況 十二	得 3B&1A	1. 得 3B 時為 264 個組合 (此四位數的位置皆錯) 2. 得 1A 時為 44 個組合 (已確定無另兩個數字)	剩 44 種組合
情況 十三	得 1A&1A	1. 得第一個 1A 時為 480 個組合 2. 得第二個 1A 時為 24 個組合 (可確定有另兩個數字)	剩 24 種組合
情況 十四	得 1A&2A	1. 得 1A 時為 480 個組合 2. 得 2A 時為 24 個組合	剩 24 種組合
情況 十五	得 1A&3A	1. 得 1A 時為 480 個組合 2. 得 3A 時為 4 個組合 (可確定無另兩個數字)	剩 4 種組合
情況 十六	得 2A&0	1. 得 2A 時為 180 個組合 2. 得 0 時為 12 個組合 (可確定有另兩個數字)	剩 12 種組合
情況 十七	得 2A&2A	1. 得第一個 2A 時為 180 個組合 2. 得第二個 2A 時為 6 個組合 (可確定無另兩個數字)	剩 6 種組合
情況 十八	得 3A&0	1. 得 3A 時為 24 個組合 2. 得 0 時為 8 個組合	剩 8 種組合

(註 1：此探討情況以固定第一次皆猜 1234，第二次皆猜 5678 為例。如為 1B&1B 為猜 1234 得 1B，猜 5678 得 1B；1B&2B 為猜 1234 得 1B，猜 5678 得 2B，以此類推。)

(註 2：此表格是用程式加以輔助完成，程式畫面位於附錄二。)

伍、延伸問題

一、為什麼一般常見是四位數，而不是二位、三位、五位？

我們根據我們上述的猜題方法去解其他位數，看是不是比較簡單抑或困難。

(一) 如果是猜二位數：

我們猜的數字及敵方給予組合	猜的次數	我們的思考邏輯
10(1A)、16(0)、80(1A)、50(1A)、40(2A)	5 次	從 16 為零可以知道 0 在第二個數字
38(1A)、30(1B)、08(2A)	3 次	從 30 為 B 可以知道 8 在第二個數字，0 在第一個數字
91(1A)、96(0)、81(1A)、61(1A)、31(1A)、01(1A)、51(2A)	7 次	從 96 為零可以知道 1 在第二個數字
73(0)、26(0)、85(1B)、54(0)、18(2A)	5 次	從 54 為零可以知道 8 在第二個數字
72(1A)、75(0)、92(1A)、32(2A)	4 次	從 75 為零可以知道 2 在第二個數字
83(1B)、36(0)、18(1A)、48(1A)、58(1A)、08(1A)、28(1A)、78(1A)、98(1A)	9 次	從 36 為零可以知道 8 在第二個數字

28(1A)、26(1B)、68(2A)	3 次	26 為 B 可以知道 8 在第二個數字，6 在第一 個數字
----------------------	-----	-----------------------------------

◎由實驗結果可知，2 位數可以是最好猜的，但也可能是最難猜的。因為數字只有兩個，即使已經掌握其中一個數字，另一個也不見得能立刻得知，有可能像例 2 一樣快，卻也可能像例 3 跟例 6 一樣慢，所以 2 位數的解題當中，運氣是占很大比例的。

(二) 如果是猜三位數：

我們猜的數字及敵方給予組合	猜的次數	我們的思考邏輯
123(1B)、456(1B)、789(0)、 205(2A)、204(3A)	5 次	1. 因為 123 為 1B「表取一」,456 為 1B「表取一」, 又 789 皆沒有，故必有 0 2. 從 205 可知 0 為第二個數字
123(1B)、456(1A1B)、425(1B)、 653(3A)	4 次	1. 從 123 為 1A、456 為 1A1B 可知沒有 7890 四 個 數字 2. 從 425 為 1B 可之 2&4 不是 A
123(1A)、456(0)、789(1A1B)、 728(1A)、793(3A)	5 次	1. 從 789 為 1A1B 可知沒有 0 2. 從 728 為 1A 可 7 的位置是對的→如果 2 或 8 的位置是對的，就不符合 789 為 1A1B 的條件
123(0)、456(1B)、789(2B)、 974(3A)	4 次	1. 從 456(1B)、789(2B)可知沒有 0、1、2、3
123(0)、456(0)、789(2B)、 908(3B)、890(3A)	5 次	1. 從前三次可知應該有 7、8、9 其中兩個，也 必定有 0

123(1B)、456(0)、789(1B)、 902(1B)、018(3A)	5 次	1. 從前三項可知必定有 0
--	-----	----------------

◎由實驗結果可知，3 位數較 2 位數適合用推理的方式進行，但 3 位數所用到的邏輯思考卻少於 4 位數，故 3 位數比 4 位數少見

(三) 如果是猜五位數：

我們猜的數字	對手給予我們的組合	我們的思考邏輯
(1) 12345	1A3B	表 67890 只有一個數字對
(2) 67890	1B	無法知道哪個數字的位置，如果是 A，即可開始代入
(3) 13427	1A2B	假設 1 是 A，234 是 B
(4) 14539	4B	表 1 不是 A，有 4 個數字是對的
(5) 59412	1A2B	表有 3 這個數字，且沒有 2
(6) 31456	3A1B	表沒有 6 這個數字，所以第五個位置是 5
(7) 31405	5A	這一個是運氣成分占較大，如果前一個的 3A 代的並不是 314，則結果將會增加

→猜了 7 次

我們猜的數字	對手給予我們的組合	我們的思考邏輯
(1) 12345	2A1B	表有兩個數字是對的
(2) 06789	2B	表有兩個對的數字位置錯
(3) 52397	2A1B	設 23 位置是對的，得 2A，下一個依舊會猜 23
(4) 62350	4A	表有四個數字是對的

(5) 62358	3A1B	表沒有 6 這個數字，且 2350 位置是對的，而 8 的位置是錯的
(6) 82350	5A	

→猜了 6 次

◎由實驗結果可知五位數並不容易猜出，思考邏輯要很清晰，條理分明，但也因數字過多，所以我們下一個將用不是規律性(12345、67890)的推理的方式推出

我們猜的數字	對手給予我們的組合	我們的思考邏輯
(1) 68205	2A1B	
(2) 98271	1A	由此可知 8 和 2 只有一個是對的
(3) 38504	3A1B	可確定 8 是對的，0 和 5 也可能是 A，由這兩項繼續推下去
(4) 48506	2A2B	可知 8 和 5 的位置是對的，4 和 6 是 B，所以依前一個的組合再繼續推理
(5) 68534	5A	

→5 次

我們猜的數字	對手給予我們的組合	我們的思考邏輯
(1) 61395	1A2B	
(2) 82470	2B	不能與前一個猜數的數字重複
(3) 69723	3B	表 6 不是 A
(4) 31648	2A2B	假設 3 和 1 是 A，6 和 8 是 B，依不同的組合繼續推理下去

(5) 31869	4A	可知 3、1、8、6 四個數字皆對，依前面的 3B 繼續往下推理
(6) 31867	5A	

→6 次

◎由實驗結果可知五位數用非規律性的推理會比規律性的快，但非規律性的運氣成分占很大部分，如果一開始的方向錯誤，那會比規律性更難察覺，所以也有可能猜超過 10 次以上。

二、相關遊戲及應用方法：

(一) 小孩很忙-決戰 ATM、寶貝 ATM(電視節目)

1. 遊戲規則:有三組參賽者，剛開始每名參賽者都擁有 10,000 元資金，以一些生活中益智、常識趣味的問答遊戲為主，進行搶答，最終優勝者可以參加獎金挑戰賽，也就是決戰 ATM 或寶貝 ATM)。
2. 寶貝 ATM、決戰 ATM 遊戲流程：參賽者先出題，將答案放置到安全地方後即不能更改，而主持人會有五次機會猜出答案以守住獎金，A 表示位置及數字正確，如猜出 A，將會直接告訴主持人；B 表示數字對，位置錯，但並不會告訴主持人 B 的數字。

(1) 範例：(設答案為 5902) 主持人猜 6805(1A1B)，而參賽者會告知主持人說 0 是正確的，但並不會告訴主持人 5 就是 B。



(寶貝 ATM 遊戲畫面)

(二) 昌爸工作坊—猜數字、1A2B(網路遊戲)

1. 規則介紹及講解： 玩家先攻，與電腦對抗，需比電腦早答出答案。A為數字位置都對；B為數字對但位置錯，無論猜出A或B，都不會告訴對方。

(1) 範例：(設答案為 5902)

如 6805(1A1B)，我們並不會告訴敵方 0 是正確的或是 5 就是 B。



(昌爸猜數字遊戲勝利畫面)

陸、 結論

一、 如何在最少的次數內找到答案組合：

需要由規律性 (1234、5678、1357……) 開始，接下來猜的數字組合都須參考前面給的提示，才能在最少步數中猜出敵方的數字。

二、 如何有效讓敵方 (電腦) 猜的次數保持最多的答案組合：

為讓我方有充裕的時間，故我們必須擁有一個良好的數字組合，一個良好的組合須有以下的條件：

- (一) 組合的數字不能過於相近
- (二) 不必一開始就想好我方的數字組合
- (三) 須給敵方”好”的提示，再從提示中找出我方的數字組合 (如研究過程三)
- (四) (1B&2B)及(2B&2B)是最好的兩個提示，可使電腦花最多時間。

三、 四位數的活用性遠超出其他位數的原因主要有以下幾點：

- (一) 推理成分占絕大部分
- (二) 難易度適中（不會比其他位數難或是簡單）
- (三) 步數及方式較固定（如延伸問題一）

四、探討多種不同情況

- (一) 如果電腦給予(1B&2B)、(2B&2B)，必須花較多步數才能猜出。
- (二) 如果電腦給予(1A&3A)、(2A&2A)，則可在較少的步數內猜出。
- (三) 第一個提示如給予 1B，則要考慮的情況將會是最多種(1440 種)；
第一個提示如給予 3A，則要考慮的情況將會是最少種(24 種)。

柒、參考資料

1. 劉昭麟，數學、資訊科學與數字遊戲，*科學月刊*，第 32 卷第 3 期，第 250 到 255 頁，財團法人臺北市科學出版事業基金會，2001 年。

取自：<http://www.cs.nccu.edu.tw/~chaolin/papers/science3203.pdf>

2. 李信昌，昌爸數學工作坊之數學遊戲

取自：<http://www.mathland.idv.tw/game/guassab.htm>

3. ASP.NET 程式設計學分班，勞新部落格，隨意窩 Xuite 日誌。

取自：

<http://blog.xuite.net/ahsinchang/wretch/177479216-%E3%80%90ASP.NET%E7%A8%8B%E5%BC%8F%E8%A8%AD%E8%A8%88%E5%AD%B8%E5%88%86%E7%8F%AD+%E3%>

捌、附錄

附錄一：程式

```
<html>
<head>
<meta http-equiv="X-UA-Compatible" content="IE=edge" >
</head>
<script>
var n_numbers;          //存放尚有幾個待選字
var numbers=new Array(); //存放待選數字
function new_game()
{
//遊戲開始,填入全部不重複4位數
n_numbers=0;
for(i=0;i<=9;i++){
for(j=0;j<=9;j++){
for(k=0;k<=9;k++){
for(l=0;l<=9;l++){
if((i!=j)&&(i!=k)&&(i!=l)&&(j!=k)&&(j!=l)&&(k!=l))
{
numbers[++n_numbers]=''+i+j+k+l;
}
}
}
}
}
}
```

```

}
function compareAB(a, b)
{
//比較兩個數字幾 A 幾 B,
//傳入一個四位數字串, 傳回一個 2 位數字串, 前面 A, 後面 B
//例如傳回 21 表 2A1B
var a, b;
var i, j, k, l;
if((a.length!=4) || (b.length !=4))
    return '00' ;
k=0;
l=0;
for(i=0;i<4;i++)
{
    if(a[i]==b[i]) k++;
    for(j=0;j<4;j++)
        if(a[i]==b[j]) l++;
}
l-=k;
return('' + k+l);
}
function is_diff(a)
{
//檢查是否四位數都不重複
var a;
var i, j, k;
k=0;
if(a.length != 4) return(false);
for(i=0;i<4;i++)
    for(j=0;j<=4;j++)
        if(a[i]==a[j])k++;
if(k==4) return(true);
return false;
}

```

```

}
function check_feedback(n, a)
{
    //檢查符合使用者回饋 AB 的數字
    //把不符合者剔除
    //傳回剩餘幾個數字
    //若是格式不合或已無待選字,則傳回-1
    var n, a;
    var i, j;
    if((n.length!=4)|| (a.length!=2)) return(-1);
    if(!is_diff(n)) return(-1);
    if(n_numbers==0) return(-1);
    j=1;
    for(i=1;i<=n_numbers;i++)
    {
        if(compareAB(numbers[i],n)==a) numbers[j++]=numbers[i];
    }
    n_numbers=--j;
    return(n_numbers);
}
function get_number()
{
    //從待選數字中隨機挑一個數字出來
    //若已沒有數字可選,則回覆'0000'
    if(document.main_form.guess_times.value==1) return('1234');
    if(document.main_form.guess_times.value==2) return('5678');
    if(n_numbers<=0) return('0000');
    return(numbers[Math.floor(n_numbers * Math.random()+1)]);
}
//-----
//以下是測試上面程式的程式
function test_AB(n)
{

```

```

new_game();
var ans=numbers[Math.floor(n_numbers * Math.random()+1)];
document.write("測試:");
}
//-----
//以下是與使用者互動操作的部分
var gs;    //猜第幾次
function game_start()
{
    new_game();
    gs=1;
    document.main_form.guess_times.value=gs;
    document.main_form.comp_ans.value=get_number();
    document.main_form.respond.value="";
    document.main_form.respond.focus();
}
function user_reply()
{
    u=''+document.main_form.respond.value;
    a=''+document.main_form.comp_ans.value;
    check_feedback(a,u);
    document.main_form.respond.value="";
    document.main_form.guess_times.value=++gs;
    document.main_form.comp_ans.value=get_number();
    document.main_form.respond.focus();
}
</script>
<body>
<form name=main_form>
<input type=button value=開始新一局 onclick="game_start()"><br>
電腦猜:<input type=text name=comp_ans><br>
結果是:<input type=text name=respond><input type=button value=確定 onclick="user_reply()"><br>
    請用 2 位數表示，例如 2 1 表示 2 A 1 B、0 2 表示 0 A 2 B<br>

```

請用 2 位數表示，例如 2 1 表示 2 A 1 B

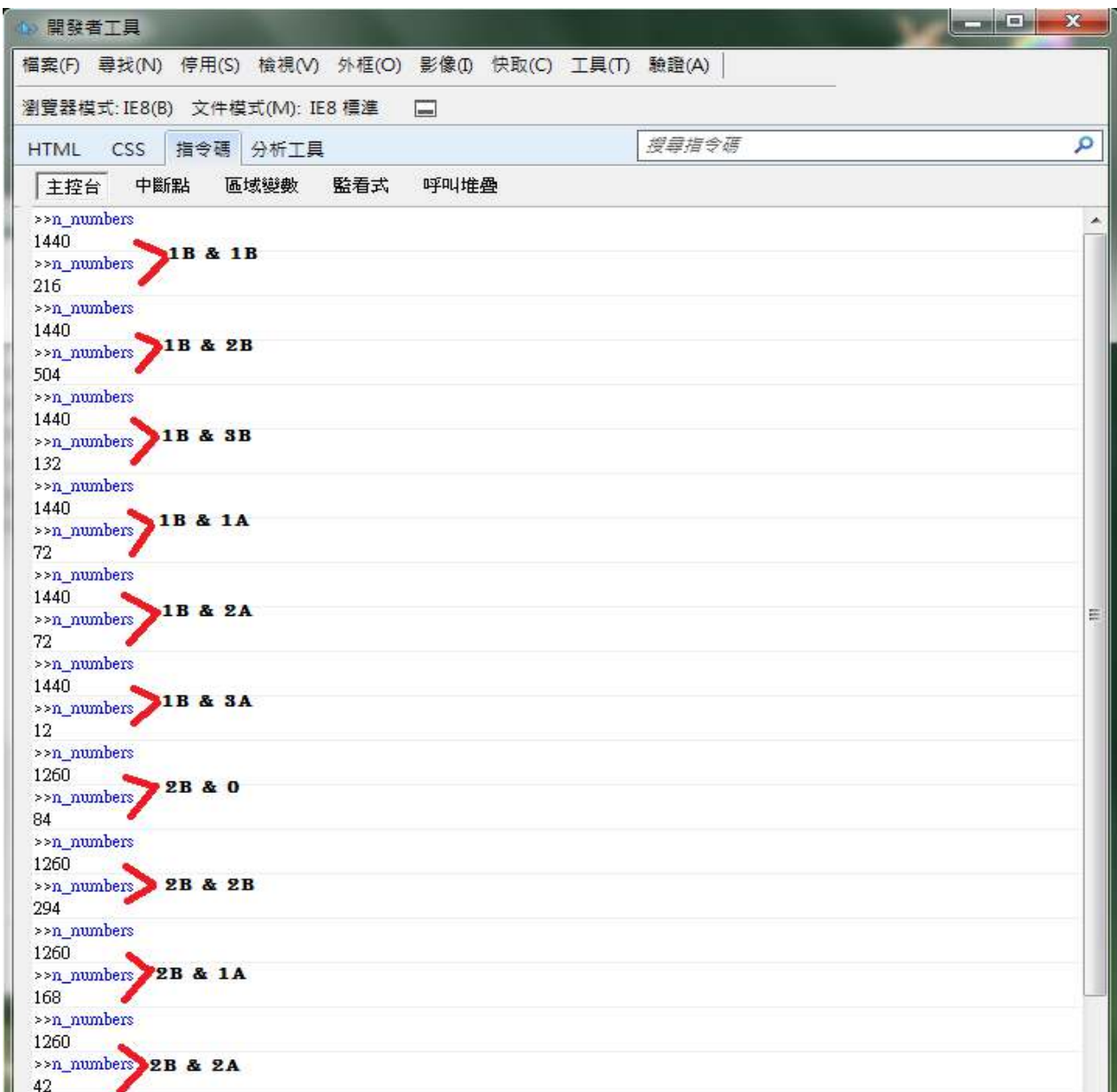
現在猜第<input type=text name=guess_times size=3>次

</form>

</body>

</html>

附錄二：程式畫面



開發者工具

檔案(F) 尋找(N) 停用(S) 檢視(V) 外框(O) 影像(I) 快取(C) 工具(T) 驗證(A)

瀏覽器模式: IE8(B) 文件模式(M): IE8 標準

HTML CSS 指令碼 分析工具 搜尋指令碼

主控台 中斷點 區域變數 監看式 呼叫堆疊

```
>>n_numbers
264
>>n_numbers } 3B & 0
88
>>n_numbers
264
>>n_numbers } 3B & 1A
44
>>n_numbers
480
>>n_numbers } 1A & 1A
24
>>n_numbers
480
>>n_numbers } 1A & 2A
24
>>n_numbers
480
>>n_numbers } 1A & 3A
4
>>n_numbers
180
>>n_numbers } 2A & 0
12
>>n_numbers
180
>>n_numbers } 2A & 2A
6
>>n_numbers
24
>>n_numbers } 3A & 0
8
```


【評語】 030420

作者利用電腦程式篩選方法，得出猜幾 A 幾 B 的程式，並經由實驗說明約在 7 步後即可猜出數字，作者有電腦程式之功力，思考亦清晰。然作品缺乏定量之結論，沒有分析此猜測方法的最優與否，是較為不足之處。