

# 大樂透投資報酬率模擬計算

實習目標：

1. 函式設計與製作練習
2. 陣列與陣列初始化
3. 多重選項條件判斷敘述
4. srand() rand() 練習
5. Monte Carlo method and evaluation
6. 如何判斷隨機模擬程式表現是否正確
7. 如何比對兩個使用隨機亂數程式的表現是否相同?
8. 瞭解 rand() 可以用在哪些應用, 不能用在哪些應用

丁培毅

1

# 大樂透投資報酬率模擬計算

1. 請盡量運用函式撰寫一個程式模擬 6000 次大樂透彩券購買，完全電腦隨機選號(快選)，然後再模擬開獎，由 49 個號碼裡開出六個號碼外加一個特別號，103086 期大樂透獎金分配如右表，請撰寫計算某一注所獲得彩金的函式，並計算 6000 注獎金的總金額，假設這 6000 次是同一個人購買的，請計算其投資報酬率 (全部獲得彩金除以投資總成本)

獎項	對中獎號數	單注獎金	中獎注數
頭獎	6 個	19,100,192	0
貳獎	任 5 個 + 特別號	1,569,878	0
參獎	任 5 個	32,705	48
肆獎	任 4 個 + 特別號	9,781	107
伍獎	任 4 個	1,286	2238
陸獎	任 3 個 + 特別號	1,000	3133
柒獎	任 2 個 + 特別號	400	30279
普獎	任 3 個	400	35512

<http://www.9800.com.tw/html/a1/103086.html>  
103086期 銷售金額101,116,450 (2022329 注)

2. 程式範例  
輸出如右：

頭獎:0,貳獎:0,參獎:0,肆獎:0,伍獎:4,陸獎:6,柒獎:85,普獎:75 Total returns = 75144 (out of 300000 investments) ROI=25.05%
頭獎:0,貳獎:0,參獎:0,肆獎:1,伍獎:4,陸獎:8,柒獎:70,普獎:102 Total returns = 91725 (out of 300000 investments) ROI=30.58%
頭獎:0,貳獎:0,參獎:1,肆獎:1,伍獎:6,陸獎:8,柒獎:74,普獎:101 Total returns = 128202 (out of 300000 investments) ROI=42.73%

2

## 分析

1. 這個題目需要使用 `stdlib.h` 裡提供的虛擬亂數產生器 `rand()` 以及初始化函式 `srand()`，一般常用 `time.h` 裡的 `time()` 函式來取得 1970/01/01 00:00:00 到現在之間的秒數來作為亂數種子，如果需要產生均勻分佈的 0~(n-1) 之間的整數，將這個均勻分佈在 [0,1) 的浮點數乘 n 再轉為整數  $(int)((double) rand() / (RAND\_MAX+1) * n)$  即可，平常偷懶一點的方法是 `rand()%n`，均勻分佈的特性較差。在這個程式裡，需要模擬購買彩券 (每一注六個號碼)，需要模擬開獎 (每一期六個號碼外加一個特別號)，基本上都需要是均勻分佈的亂數
2. 這是一個模擬程式，程式要求先買彩券，然後開獎，然後兌獎計算投資報酬率，這是很合理的程序，不過如果我們照著做的話，會需要把每一注彩券的 6 個號碼都記下來，等到開獎以後才能夠兌獎，當投注的人數很多的時候，需要相當多的記憶體，否則就需要用檔案存下來；不過稍微想一下就會發現如果把開獎一道買彩券之前，就不需要記錄所有投注，也不會影響到程式要求計算的投資報酬率

3

3. 首先我們先設計需要的變數並且適當初始化  
各獎項名稱: `const char *prizeName[] = {"頭獎", "貳獎", "參獎", "肆獎", "伍獎", "陸獎", "柒獎", "普獎"};`  
各獎項獎金: `const int prizes[] = {19100192, 1569878, 32705, 9781, 1286, 1000, 400, 400};`  
開獎號碼: `int winningNumbers[7]; // 第 7 個是特別號`  
中獎人數: `int nWinners[8] = {0};`  
投注號碼: `int betNumbers[6];`  
投注次數: `const int nBets=6000;`
4. 電腦投注和開獎都需要由 49 個數字裡均勻選出不重複的 6~7 個數字，可以寫成一個函式 `void getNumbers(int size, int numbers[])`，指定 `size` 為 6 或是 7，函式由 1~49 均勻地選取指定個數的數字出來，因為要求不重複，所以不能只寫一個單一迴圈直接呼叫 `rand()%49+1` 就做完了。基本上有幾種作法：第一是和電腦洗牌的程式一樣，不斷地交換兩個數字，把 49 個數字均勻的打散，然後把前面的 `size` 個數字拷貝到 `numbers[]` 陣列中；第二種方法則是因為看到我們每次只需要 6 到 7 個數字，不需要把 49 個數字都打亂，只需要在一個袋子裡均勻地抽數字出來，抽 6 次就可以得到  $1/49 \cdot 1/48 \dots 1/44$  的均勻分佈機率

4

- 程式先呼叫 `getNumbers(7, winningNumbers)` 得到開獎的 7 個不重複的號碼，然後進入一個迴圈，每一次迴圈裡會先投注，也是用 `getNumbers()` 來得到 6 個不重複的電腦選號號碼，然後需要比對 `winningNumbers[]` 來確定是否有中獎，這裡設計一個函式 `int match(winningNumbers, betNumbers)` 來比對，回傳 -1,0~7 代表未中獎、以及頭獎~普獎  
這個函式裡需要有一個迴圈來逐一檢視 `betNumbers[i]` 是否是特別號或是得獎號碼，計算有幾個得獎號碼，有了這兩個資料以後，用一個多重選項的 `if` 敘述來判斷並回傳得到哪一種獎項
- 步驟 5 中的投注和兌獎的動作要重複 6000 次，把對中每一個獎的總注數記錄在整數陣列 `nWinners[]` 中，除了陸獎、柒獎和普獎是固定金額外，其它獎的總獎金佔總銷售金額的比例是固定的，由於 6000 次佔該期 2022329 注的比例很低，在模擬時各獎項的中獎次數應該都比實際中獎注數少，所以獎金直接以 `nWinners[i]*prizes[i]` 來計算，最後請印出各獎項中獎注數、總獎金以及投資報酬率

5

## 程式寫對了嗎？

- 『程式可以執行、不會當掉』並不代表程式的結果正確、有科學上或是工程上的參考價值
- 有些程式很容易確定到底對了沒有，例如排序、搜尋的程式，都是這一類，前面寫的這個程式就是一個你（在看到程式輸出時）不容易判斷它到底寫對了沒有的東西
- 最主要的原因是亂數，每次執行的結果都不太一樣，就算看到某一個結果不太對，下一次執行時問題就不見了，尋找錯誤和除錯有點困難，這個狀況其實很容易解決 - 固定程式用的亂數種子，程式的表現就不會亂動了，看到結果不太對時，重新執行一定會看到一樣的結果
- 另外程式的結果和同學寫的程式的結果、和範例程式的結果都不太一樣，沒辦法直接比較... 通常這可以由兩個方向來處理：如果你希望和同學寫的程式比較，你需要 ❶ 固定使用亂數的模組，例如前面寫的 `getNumbers()`，大家都用同一個，同時使用的次數一樣，那麼就可以比較其他不同的程式部份的表現； ❷ 用統計量來比較，比方說你的一次隨機實驗得到結果 0.3，那麼做 1000 次實驗，結果的平均值和標準差是可以拿來比較的

6

## 程式寫對了嗎？(cont'd)

- 前面這個重複隨機實驗的次數，牽涉到的是結果的可信賴度，這是獨立的隨機實驗的特性，我們可以運用 **Chebychev Inequality** 或是 **Chernoff Bound** 來限制多次實驗結果的標準差，從而得到一個可信賴的平均數量
- 另外一個需要掌握的東西是理論預測值，你看到你的程式的模擬結果，ROI 在 20% ~ 40% 是合理的嗎？如果你的程式結果是 1% 以下，是 300% 可能就要思考一下原因了
- ROI 真的合理嗎？不是大部份的人去買都是槓龜嗎？怎麼有 30%？
- 樂透的 ROI 其實應該就是其總獎金的比例，樂透只是財富很不均勻的重新分配，雖然 ROI 不像一般的投資會是 80% ~ 120% 間，但是也不會低到 1%，如果不是有特別加碼或是獎金累積，也不會有超過 100% 的

7

## 大樂透投資報酬率

- 每期提撥之總獎金，為當期銷售總金額的 55%，分配比率：總獎金去掉陸柒普獎後，頭獎 73%、貳獎 6%、參獎 6%、肆獎 4%、伍獎 11%，數人同中某獎項(伍獎以上)，則均分該獎項之獎金
- 中獎機率：
  - 頭獎： $1/C(49,6)=1/13983816=7.1511 \times 10^{-8}$
  - 貳獎： $C(6,5)/C(49,6)=6/13983816=4.2907 \times 10^{-7}$
  - 參獎： $C(6,5)C(42,1)/C(49,6)=252/13983816=1.8021 \times 10^{-5}$
  - 肆獎： $C(6,4)C(42,1)/C(49,6)=630/13983816=4.5052 \times 10^{-5}$
  - 伍獎： $C(6,4)C(42,2)/C(49,6)=12915/13983816=9.2357 \times 10^{-4}$
  - 陸獎： $C(6,3)C(42,2)/C(49,6)=17220/13983816=1.2314 \times 10^{-3}$
  - 柒獎： $C(6,2)C(42,3)/C(49,6)=172200/13983816=0.012314$
  - 普獎： $C(6,3)C(42,3)/C(49,6)=229600/13983816=0.01642$
- 如果伍獎以上都有中獎者（沒有任何獎金保留到次期），則平均的投資報酬率應該就是 55%。為什麼前面的模擬結果才 20% 30%？

8

# 大樂透投資報酬率

- 103086 大樂透開獎開出號碼為 09, 10, 14, 20, 36, 47, 特別號 08, 獎金分配如右表, 請撰寫一個程式模擬下 2022329 注 (請注意: 單注獎金要依照前頁的比例重新算過, 不能拿右表來算, 否則總獎金不會是總投注金額的 55%, ROI 也不會對)

獎項	對中獎號數	單注獎金	中獎注數
頭獎	6 個	19,100,192	0
貳獎	任 5 個 + 特別號	1,569,878	0
參獎	任 5 個	32,705	48
肆獎	任 4 個 + 特別號	9,781	107
伍獎	任 4 個	1,286	2238
陸獎	任 3 個 + 特別號	1,000	3133
柒獎	任 2 個 + 特別號	400	30279
普獎	任 3 個	400	35512

<http://www.9800.com.tw/html/a1/103086.html>  
103086期 銷售金額101,116,450 (2022329 注)

- 程式範例執行四次輸出如下:

seed	頭獎	貳獎	參獎	肆獎	伍獎	陸獎	柒獎	普獎	單注獎金	中獎注數
127	21805280	1792214	57813	14935	1713	1000	400	400		
	0	0	31	80	1918	2637	24727	33040		
	Total returns = 32016337 (out of 101116450 investments) ROI=31.66%									
123	21784694	1790522	47119	11589	1808	1000	400	400		
	0	1	38	103	1815	2504	24991	33179		
	Total returns = 33828231 (out of 101116450 investments) ROI=33.45%									
162	21813310	1792874	43728	15323	1824	1000	400	400		
	1	0	41	78	1802	2440	24839	33393		
	Total returns = 53821000 (out of 101116450 investments) ROI=53.23%									
124	21835940	1794734	49853	12594	1843	1000	400	400		
	1	1	36	95	1785	2471	24966	33111		
	Total returns = 55613367 (out of 101116450 investments) ROI=55.00%									

9

# 大樂透投資報酬率

- 104066 期大樂透開獎開出號碼為 13, 16, 18, 30, 32, 40, 特別號 29, 實際獎金分配如右表, 請撰寫一個程式下 2718051 注 (請注意: 單注獎金要依照前頁的比例重新算過, 不能拿右表來算, 前期頭獎及貳獎皆有累積彩金)

獎項	對中獎號數	單注獎金	中獎注數
頭獎	6 個	312,162,640	1
貳獎	任 5 個 + 特別號	4,808,025	1
參獎	任 5 個	69,460	37
肆獎	任 4 個 + 特別號	19,251	89
伍獎	任 4 個	2,201	2140
陸獎	任 3 個 + 特別號	1,000	3016
柒獎	任 2 個 + 特別號	400	32778
普獎	任 3 個	400	39463

<http://www.9800.com.tw/html/a1/104066.html>  
104066 期, 銷售金額135,902,550 (2718051 注),  
前期累積彩金頭獎 280,893,819 貳獎 2,237,985  
累計12次未開出頭獎

- 程式範例執行四次輸出如下:

seed	頭獎	貳獎	參獎	肆獎	伍獎	陸獎	柒獎	普獎	單注獎金	中獎注數
65	310225220	4648785	52408	12961	1766	1000	400	400		
	0	0	46	124	2502	3336	33534	44542		
	Total returns = 43002864 (out of 135902550 investments) ROI=31.64%									
342	310066226	4635717	59943	13546	1732	1000	400	400		
	0	1	40	118	2538	3365	33745	44803		
	Total returns = 47811881 (out of 135902550 investments) ROI=35.18%									
2134	310193100	4646145	50170	13721	1770	1000	400	400		
	1	0	48	117	2494	3414	33299	44692		
	Total returns = 353231397 (out of 135902550 investments) ROI=259.92%									
453	310222884	4648593	46357	11561	1741	1000	400	400		
	1	1	52	139	2537	3334	33551	44538		
	Total returns = 357875537 (out of 135902550 investments) ROI=263.33%									

10

# 大樂透包牌投資報酬率模擬

- 104/07/31 大樂透開獎開出號碼為 13, 16, 18, 30, 32, 40, 特別號 29, 獎金分配如下表, 請撰寫一個程式, 模擬下列兩種情境的投資報酬率
  - 選 12 個號碼一般包牌, C(12,6)=924 注
  - 聰明包牌, 選 12 個號碼  
中 6 保 5, 38 注

獎項	對中獎號數	單注獎金
頭獎	6 個	312,162,640
貳獎	任 5 個 + 特別號	4,808,025
參獎	任 5 個	69,460
肆獎	任 4 個 + 特別號	19,251
伍獎	任 4 個	2,201
陸獎	任 3 個 + 特別號	1,000
柒獎	任 2 個 + 特別號	400
普獎	任 3 個	400

11

# 聰明包牌, 選 12 個號碼, 中 6 保 5, 38 注

1	01, 02, 03, 04, 05, 07	20	02, 03, 04, 06, 10, 11
2	01, 02, 03, 04, 05, 12	21	02, 03, 05, 06, 08, 09
3	01, 02, 03, 04, 06, 10	22	02, 03, 05, 06, 10, 11
4	01, 02, 03, 06, 08, 12	23	02, 03, 05, 08, 09, 10
5	01, 02, 03, 07, 09, 10	24	02, 03, 07, 08, 11, 12
6	01, 02, 04, 08, 09, 11	25	02, 03, 07, 09, 11, 12
7	01, 02, 05, 06, 09, 11	26	02, 04, 05, 07, 08, 11
8	01, 02, 05, 10, 11, 12	27	02, 04, 06, 07, 09, 12
9	01, 02, 06, 07, 08, 10	28	02, 04, 08, 09, 10, 12
10	01, 03, 04, 08, 09, 11	29	02, 05, 06, 07, 10, 12
11	01, 03, 05, 06, 07, 11	30	03, 04, 05, 09, 11, 12
12	01, 03, 05, 08, 10, 11	31	03, 04, 06, 07, 08, 09
13	01, 03, 06, 09, 10, 12	32	03, 04, 07, 08, 10, 12
14	01, 04, 05, 06, 08, 10	33	03, 05, 06, 07, 10, 12
15	01, 04, 05, 06, 09, 10	34	04, 05, 06, 08, 11, 12
16	01, 04, 06, 07, 11, 12	35	04, 05, 07, 09, 10, 11
17	01, 04, 07, 08, 09, 12	36	05, 07, 08, 09, 11, 12
18	01, 04, 07, 10, 11, 12	37	06, 07, 08, 09, 10, 11
19	01, 05, 07, 08, 09, 12	38	06, 08, 09, 10, 11, 12

12

## 包牌投資報酬率

104000066 期

選12中6機率 =  $C(43,6)/C(49,12) = 0.00006608$

選12中5 + 特別號機率 =  $(C(6,5)C(42,6))/C(49,12) = 0.0003411$

1. 一般包牌(系統組合),  $C(12,6)=924$ 注, 成本= $924*50=46200$

報酬  $\geq 0.00006608*(312162640$ 頭獎) +

$0.0003411*(4808025$ 貳獎) = 20627.7+528.8

報酬率  $\geq 20627.7/46200 = 0.4579$

2. 聰明包牌, 選 12 個號碼中6保5, 38注, 成本= $38*50=1900$

報酬  $\geq 0.00006608*(38/924*312162640$ 頭獎+69460參獎)+

$0.0003411*(19251$ 肆獎) = 847.05+5.8620

報酬率  $\geq 852.92/1900 = 0.4489$

所以聰明包牌法是用很低的成本，得到和一般包牌法幾乎一樣的報酬率比先前這個實習用的電腦選號(6000注)的成本低很多而且報酬率(~36%)高很多

13

## 關於虛擬亂數在程式中的使用

1. 前面這個範例裡樂透的開獎，還有先前的擲骰子、撲克牌洗牌、象棋暗棋、線上遊戲(?A?B, RPG, ...) 的這幾個範例，雖然作法很常見，卻很容易讓同學誤會：以為這些都可以用 `srand() / rand()` 來實作 - 其實是不可行的
2. `srand() / rand()` 只是提供一個統計性質還不錯的虛擬亂數，工程上為了模擬自然界中的現象，比如水或空氣的亂流，風力的變化，地震強度變化，受力的改變，通訊時雜訊的強度，溫度的變化等等，可以由這種均勻的亂數來產生具有不同分佈的亂數，巨觀來看系統的表現就很像自然界中的表現
3. `srand() / rand()` 和真正的亂數有很大的差別，真正的亂數是不可預測的，它們是可以預測的，想像一下如果有人預測得到你的線上撲克牌遊戲的亂數序列，你的洗牌程式的表現他就完全可以預測，完全知道其他玩家手上的牌，那就沒有公平性了 - 過去有線上德州撲克的遊戲就被人發現亂數是用 `rand()`，因此寫了一個程式來預測伺服器上的時間與亂數序列而完全破解
4. 另外在資訊安全應用裡很多密碼系統都需要亂數序列作為密鑰，這些也絕對不能使用 `rand()` 的，否則就完全沒有安全性了，在資訊安全應用裡需要用基於單向函式的密碼學虛擬亂數產生器 - 以前 IE 和 Netscape 瀏覽器裡面的 SSL 實作也被人發現過是用 `rand()`

14