

## T12 Project B 專案說明

### (一) 專案題目與源起

#### ● 專案題目：21 點賭神遊戲—你能贏電腦嗎？

Something super~ fun

並且用到 (1)亂數、(2)陣列

(跟電腦以 1-1 正面生死鬥，可謂未來科技化後賭神最愛的遊戲概念股)

#### ● 專案源起：

In the beginning, we all had many ideas. We thought of mini-game, 踩地雷, but in the end, we found that there were many obstacles to implementation. Then we thought that we could design a game called Bingo(賓果), but we felt it sounds boring if we only make the Bingo(賓果) game. We decided to combine Bingo(賓果) with Blackjack(21 點).

After some twists and turns, finally decided the ideal option under the mutual agitation of creativity: "21 Point Gambling". We chose it for the following reasons:

#### 1. Something fun 且用到 (1)亂數、(2)陣列

撲克牌遊戲在程式中皆能應用到陣列和亂數的功能，在同時需要儲存多張手牌時需要用到陣列來儲存，而在抽牌時則需要使用亂數的特性，加上撲克牌遊戲的普遍性，使得其規則為大眾所熟悉，較容易在短時間內上手。

#### 2. 「好玩性」

考量到若是只有單純 21 點似乎略顯乏味，我們有思考過嘗試將賓果和 21 點做結合，但在規則設計和玩法上似乎仍無法達到具備「好玩性」的要求，難以盡善盡美，因此我們轉而思考其他方式，將人類天生對於博弈的熱愛和幾近瘋狂的著迷納入考量之內。所謂小賭可以怡情，大賭可以興邦立業，人類天生對於金錢博弈容易產生興奮感，只要一上賭桌，便會感到心跳加快、血行加速。因此我們將賭金系統與 21 點結合，在娛樂休閒之餘增加一點刺激，以期能達到「好玩性」的要求。

Knowing loving to win and hating to lose is one of the basic human trait. We have added the factor of winning and losing the game into our program. The player will “win” the game when they have reached the condition we have set into the game, that is the player will win the game when their money reach 5000. On the other hand the condition for losing is also added into the game. The player will lose the game when their money reached 0. The end will end immediately when the player reach the condition needed to win or lose. Another special condition to win has also been added to the game where the player will

immediately win the game when his card reaches 5 and his points have not exceed 21.

## (二) 程式說明

- 輸入資料
  - 擇賠率大小 → int mag
  - 1. 輸入賭注金額 → int pri
  - 2. 選擇是否要再抽一張牌 → char answer
- 輸出資料
  - 1. 在每一次玩家抽取下一張牌後，印出目前玩家的所有手牌和電腦所取得的第一張牌
  - 2. 在每回合結束後印出結果，包括
    - i. 玩家和電腦的所有手牌
    - ii. 目前所擁有的資金
    - iii. 本輪結果，以玩家為主體，分為獲勝、失敗、平手等情況
  - 3. 在最終當資金大於等於 5000 或小於等於 0 時，印出成功或失敗
- 參數設定
  - 整數
    - 1. usermoney → 玩家資金
    - 2. i, j → 作為陣列的 index
    - 3. total → 玩家的牌值總和
    - 4. total\_2 → 電腦的牌值總和
    - 5. bothbust → 判斷是否有兩邊同時爆牌的狀況，預設為 0
    - 6. mag → 賠率
    - 7. pri → 本輪所下注的金額
  - 布林值
    - 1. fiveCard → 判斷是否為特殊情況，預設為 False
  - 陣列
    - 1. player → 玩家手牌
    - 2. computer → 電腦手牌
  - 函數
    - 1. draw() → 使用 rand() 隨機產生 1~13 之間的數字來模擬抽牌
    - 2. oneMoreOrNot → 使電腦判斷是否要抽下一張卡
    - 3. sum → 計算牌值總和
    - 4. printcards → 印出目前手牌，包含能判斷要印出 J Q K 而非 11 12 13

- 處理方法

玩家開始時預設資金為 1000 元，並可在每一輪開始前選擇賠率大小，分別為 1:1、1:2、1:5，接著可選擇下注金額。玩家獲勝條件為資金成長至 5000 元。每一次抽牌後檢查玩家手牌值和電腦手牌值是否有超過 21 點，使用 if-else 條件句，若有一方或雙方同時爆牌則直接跳出迴圈，進入結果頁面。計算時使用 sum() 輔助計算牌值，J, Q, K 牌值皆須計算為 10，並將目前玩家和電腦總和儲存。玩家是否繼續抽牌由玩家自行決定，而電腦是否抽下一張牌則由 oneMoreOrNot() 決定，此函數計算電腦在十三張牌中有幾張是符合條件(抽到後不會爆牌)，將此牌數除以 13 作為不會爆牌的機率，若此機率大於 0.416 則抽下一張牌，反之則否。當玩家不再要牌後，使用 if-else 條件句判斷輸贏或平手，最後計算所得賭金 = 投入賭金 \* 賠率。接著重複幾輪直到玩家資金達到 5000 或輸光為止。

特殊情況：(i) 玩家取得五張牌後尚未爆牌(超過 21 點)，則直接獲勝，由布林值 fiveCard 判斷。(ii) 若雙方同時爆牌則為平手，由變數 bothBust 判斷。

### (三) 分工狀況

- 程式設計：
  - 張鈞翔(21 點主程式和函數部分)
  - 余侑謙、江詠筑、柯慧慧(賭金系統部分)
- 說明文件：
  - 張鈞翔(專案題目與源起、程式說明)
  - 柯慧慧(專案題目與源起)
- 投影片：江詠筑
- 報告：余侑謙