

- 基本邏輯：

1. Class

- (a) Card

- 功能：設定牌堆中的一張牌，總共有 52 個 Card 物件
- Private：
  - `_suit`、`_number` 分別儲存牌的花色及數字
- Public：
  - 建構子：將輸入的參數傳遞給 `_suit` 和 `_number`
  - `getNumber()`、`getSuit()`：取得該張牌數字及花色
  - `function()`：根據卡牌的功能回傳不同值

- (b) Deck

- 功能：撲克牌堆
- Private：
  - `_deck`：Card 型態的 `vector`，表一副牌
- Public：
  - `setDeck()`：初始化一副完整的牌
  - `Shuffle()`：洗牌
  - `dealCard()`：依據輸入的值從牌堆中取 `n` 張牌
  - `deckSize()`：取得牌堆大小

- (c) Show

- 功能：輸出撲克牌圖樣
- Private：
  - `Vector<int> _number`：用來存放卡牌數字的部分
  - `Vector<char> _suit`：用來存放卡牌花色
  - `num` 函數：輸出卡牌的第二及倒數第二行，處理數字輸出，當數字為 11、12、13 時除輸出 J、Q、K 外還會在後面括號花色
  - `spade`、`heart`、`diamond`、`club`、`king`、`queen`、`solider` 函數：輸出卡牌花色部分
- Public：
  - `process()`：將輸入的字串拆解成花色部分及數字部分，並分別存入 `number` 和 `suit` 的 `vector` 中
  - `call()`：用於呼叫對應輸入的輸出函數，假如輸入了四張牌，則此函數由左至右分別輸出四張牌的第一行，再回到開頭輸出四張牌的第二行，以此類推
  - `clean()`：清除 `_number` 和 `_suit` 的設定

- (d) Player

- 功能：一位玩家的設定
- Private：
  - `_name`、`_id`、`_card`：儲存玩家名字、ID
  - `_cardPoint`、`_totalPoint`：儲存玩家出牌的累積點數、玩家的總積分
  - `_status`、`_inGame`：儲存玩家是電腦還是本人、玩家是否還在遊戲內
- Public：
  - 建構子：玩家初始化設定
  - `clearPoint()`、`clearCard()`：清除玩家出牌點數、玩家手中的牌

- `setStatus()`、`setID()`、`setInGame()`：設定 `_status`、`_id`、`_inGame`
- `getCP()`、`getTP()`：取得玩家出牌點數、玩家總積分
- `getID()`、`getName()`、`getStatus()`、`getInGame()`：取得玩家 `_`、`_name`、`_status`、`_inGame`
- `getCard()`、`getCardSize()`、`getCardStr()`：取得玩家的某張牌、手牌數量、手牌的花色和數字
- `addCard()`、`addTotal()`、`addPoint()`：增加手牌、增加出牌點數、增加總積分

(e) Game：

- 功能：遊戲主要邏輯，繼承 Show 物件
- Private：
  - `_member`、`_round`：儲存遊戲人數、回合數
  - `_player`、`_deck`：n 名玩家的 vector 集合、該回合遊戲的那副撲克牌
- Public：
  - `Welcome()`、`Intruaction()`、`finish()`：輸出歡迎語並呼叫 `Start()`、遊戲說明、結束語與排名
  - `showRound()`：輸出目前為第幾輪
  - `Start()`：初始化所有訊息，並遍歷所有玩家進行遊戲直到符合結束條件
  - `setPlayerID()`：每人抽一張牌並依規則設定每人的 ID
  - `setPlayer()`：設定玩家名字、身分（機器人或本人）
  - `human()`：玩家本人出牌邏輯
  - `robot()`：電腦玩家出牌邏輯
  - `judgeOver()`、`judgeLose()`、`judgeWin()`：判斷該輪遊戲是否結束、當前玩家有沒有輸、當前玩家有沒有贏

2. Main：main.h 中儲存程式需要的標頭檔，main.cpp 執行時呼叫 Game 物件中的 `Welcome()`

- 加分設計：
  1. 可設定玩家名字（機器人預設為 `robot + 編號`）
  2. 遊戲初始可選擇要遊戲說明還是開始遊戲
  3. 撲克牌特殊設計
  4. 可顯示當前牌堆還剩下幾張牌
- 執行畫面：

```
=====
Welcome to 99 points!!!

Are you ready to use your wits and luck to conquer the game?
Start your challenge and enjoy every thrilling moment!
=====
1. Instruction
2. Start the game
(chOOSE what you want, 1 or 2):
```

遊戲起始歡迎語

```
=====
1. Set how many rounds you want to play.
2. Set how many players you want (2 ~ 5), robots will play
   with you.
3. Every player will have 5 cards
4. Once you play all the cards in your hand and your points
   are less than 99, you will become the winner this round.
5. After each round, you will get some points. You will become
   the final winner if you have the most points.
```

#Special cards:

- 4: reverse (when players more than 4)
- 5: choose a player to play
- 10: add 10 or subtract 10
- J: skip this round
- Q: add 20 or subtract 20
- K: add points to 99

```
=====
1. Instruction
2. Start the game
```

(choose what you want, 1 or 2):

呼叫規則介紹

1 round! Order of this round: pp -> robot2 -> robot1

Deck Size: 37

Your turn! You have 0 points now

Q(diamond) ^ /\V/ V/\V/ V *\    /*    Q(diamond)	Q(club) ^ /\V/ V/\V/ V *\    /*    Q(club)	3 ^ /\V/ V/\V/ V *\    /*    3	5 ^ /\V/ V/\V/ V *\    /*    5	2 ^ /\V/ V/\V/ V *\    /*    2
1	2	3	4	5

Which card do you want? Enter the number:

顯示當前第幾輪，以及玩家本人遊玩時的畫面

Deck Size: 35

robot1's turn! robot1 have 0 points now

robot1 played this card:

7 ^ /\V/ V/\V/ V *\    /*    7
1

機器人出牌畫面

This round is over

pp's total points: 3

robot1's total points: 2

robot2's total points: 1

round: 0

每輪遊戲結束時的畫面

```
=====
That's the game! Congrats to the winner, and well played everyone!
Here are the final rankings
See you next time ~
=====
```

```
---
| 1 | pp    Points: 6
---
```

```
---
| 2 | robot2  Points: 3
---
```

```
---
| 2 | robot1  Points: 3
---
```

遊戲完全結束時（若有人積分相同則排名相

同）