






- 基本邏輯：
 1. 說明：輸入格式為「花色+數字+空格」並在最後輸入*做結尾，主要輸出邏輯請看 class 的 public 的 call 函數，程式執行會因輸入撲克牌數有不同程度的停頓，此為正常現象 (有時候輸出圖形會跑掉，可能是因為 terminal 不夠寬，不是程式有 bug!)
 2. Class：
 - (a) 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、joker、king、queen、solider、jump 函數：輸出卡牌花色部分
 - (b) Public：
 - process 函數：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 函數：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三行，以此類推
 - Random 函數：回傳隨機數字，數字範圍為 1~n (n：輸入參數)
 3. Main：
 輸入撲克牌花色+數字直到出現*結束，每輸入一張牌停頓 0.5 秒 (用於取用隨機數字)，設定觸發彩蛋的變數 surprise，觸發機率為 1/8，將輸入的字串加入 vector <string> enter 內，待輸入完成後將 enter 作為參數傳給 class 內的 process 函數拆分成花色 vector 和數字 vector，而後呼叫 class 的 call 函數進行圖形繪製
- 加分設計：
 1. 根據輸入撲克牌數量決定一次輸出幾張牌，不用事先決定輸出幾張牌
 2. 輸出的撲克牌可以決定自己想要的花色和數字，不限定一行內都是相同的
 3. 有九種撲克牌外型
 4. 不論花色字母大小寫程式均可運行
 5. 輸入不合規定時會輸出 joker 的撲克牌
 6. 可選擇隨機出牌(R)，可能出現撲克牌為四種花色的 1~13 及 joker
 7. 每組輸入的牌都有 1/4 的機率會觸發小彩蛋，輸出非撲克牌的牌種 (助教如果要測試可以一次輸出 5 張牌，機基本上都可以觸發至少一次)
 8. 可重複輸入及輸出撲克牌
- 執行畫面 (以黑桃 1、方塊 6、梅花 10、紅心 5、錯誤輸入為例)：

```
Enter: S1 D6 C10 h5 d18 *
```

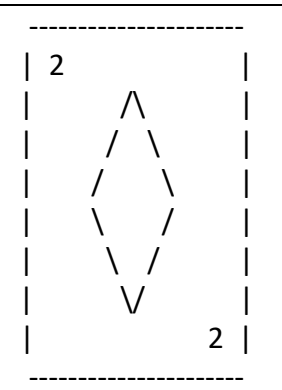
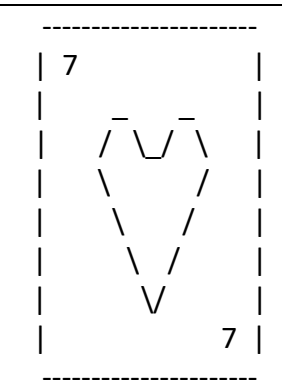
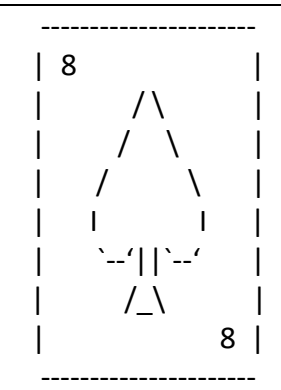
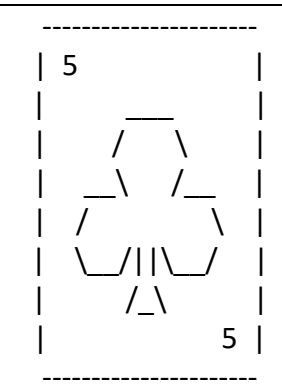
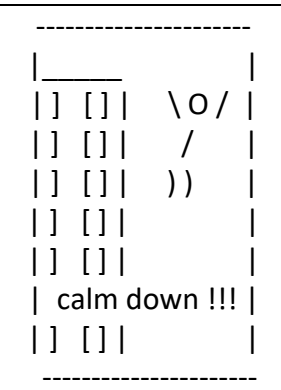
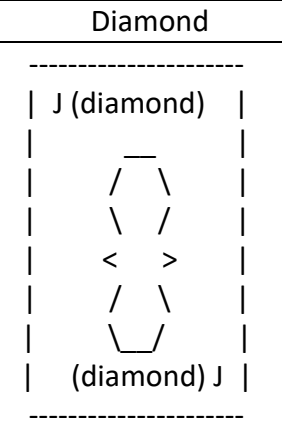
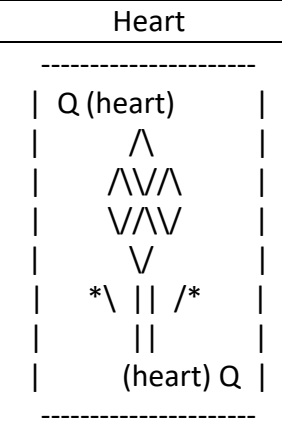
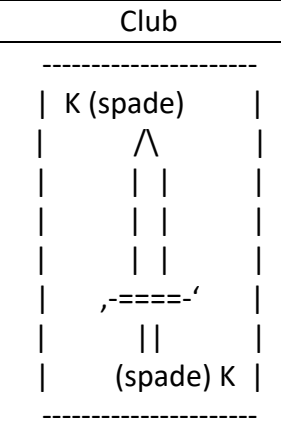
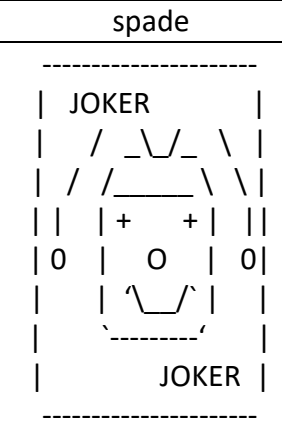
```
Do you want to execute the program again? (enter 'y' for yes, 'n' for no): y
```

```
Enter: s11 c12 d13 r x1 *
```

| | | | | |
|---|--|---|--|--|
|  <p>J(spade)</p> |  <p>Q(club)</p> |  <p>K(diamond)</p> |  <p>5</p> |  <p>JOKER</p> |
|---|--|---|--|--|

```
Do you want to execute the program again? (enter 'y' for yes, 'n' for no):
```

- 撲克牌設計：

| | | | | |
|--|---|---|--|--|
|  |  |  |  |  |
| Diamond | Heart | Club | spade | 特殊牌(彩蛋) |
|  |  |  |  | |
| solider (number = 11) | queen(number = 12) | King(number = 13) | Joker | |