

程式著要是透過 `ciacular queue` 的方式來完成這個作業，運用 `front`, `rear` 來作為判斷排庫中目前的第一張牌以及最底下那張牌的位置。

主要執行步驟：

1. 先建立一個 `string` 陣列存放每個撲克牌的文字顯示，再將其一個個 `mapping` 到 `Poker` 這個 `structure` 每張牌對應到的文字顯示。
2. 透過 `shuffle` 這個函式進行隨機洗牌
3. 一開始預設的從 13 到 1 開始抽牌，直到抽完
4. 每次做完一個回合便將牌目前的結果透過 `Print_Deck` 這個函式印出

主要函式

`isEmpty()`, `isFull()` :

藉由這兩個函式來判斷目前這個撲克牌 `queue` 是否已經被抽完(`empty`)或是已經為滿的狀態(`full`)

`Delete()`, `Add()` :

`Delete`: 當要清除牌庫時，將原本的位置的文字顯示清空，並且將數字改作 0，並把 `front` 指到下一張牌上

`Add`: 把下一個空出來的位置的文字顯示，以及數字的大小 `assgin` 到這個 `queue` 中下一個位置，並把 `rear` 的位置指向它

`Shuffle()` :

透國隨機抽出兩張牌替換位置的方式，替換 13 次

`Print Deck()` :

主要分成兩種情形，`front > rear` 和 `front <= rear`

1. `front <= rear` 時程式可以直接順著從牌頂開始依序印出直到排底 `rear`。
2. `front > rear` 時，代表 `queue` 已經發生輪轉的情況，此時就先從 `front` 開始依序印出牌型直到整個陣列的底，後面再接著從一開始陣列的頭接續印出，直到印到位在排底 `rear` 的位置。