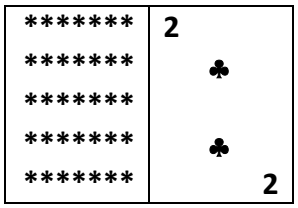
COSC 350 System Software lab 11

Client-Server Program: black jack

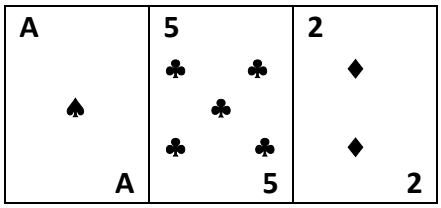
- Need build a concurrent Server
- Server need save client's information to a file: registration record, money information…
- Server ask to client to register if user does not register yet.
- If a client is already registered, need check registration record based on user name and pass word.
- Server need ask seed money to play
- A Black Jack client play against server
    - For each game, server ask a client to play or not
    - Before each game, server ask client to bet. The maximum bet amounts per each game will be the amounts of money the player currently have. A player only can play if the player has money.
    - Server shuffle 52 cards for each game
    - For each game, the software initially draws two cards for player and dealer. Player opens all cards and dealer only opens one card.
    - If server's open card is an ace card, the server need ask insurance to the client. Insurance money amount will be 50% of what the client bet. If dealer got black jack, then pay back amount of money bet by player. But, player loses the insurance money. If dealer do not have black jack, your software ask hit or hold to the client.
    - The server must hit a card if the total is less or equal to 16.
    - The server cannot hit a card if current total is greater or equal to 17.
    - If a client got Black Jack, the server immediately pay the reward ($\times$ 1.5) to the client
    - A reward per each game is very simple.
        - If a client gets Black Jack, reward will be 1.5 × bet amount.
        - If a client has higher card than the server, reward will be same amount as what the client bet for the game.
        - If a client got same as server has, the client does not lose any money.
        - If the client busts, the client loses bet money immediately.
    - Server need keep each client's money information. If a client does not have money to play, server need ask to deposit money or quit.

- Each snap shot must be displayed by a client. Display cards a in console application can be done by using Unicode.
- A client need display a snap shot for server's cards and the client card as following format by using Unicode.

Server card: ?



Player Card: currently 18 OR 8




```
#include <stdio.h>
const char spade[]="\xe2\x99\xa0";
const char club[]= "\xe2\x99\xa3";
const char heart[] = "\xe2\x99\xa5";
const char diamond[]="\xe2\x99\xa6";
void CardAce(const char[]);
int main()
{
      CardAce(spade);
      return 1;
}

//*****************************************************
// Function Name: CardOne CardTwo ...
// return type: none
// parameter: char
// Display a card
//*****************************************************
// function for display card one
void CardAce(const char Pic[])
{
       printf("---------\n");
       printf("|A      |\n");
       printf("|       |\n");
       printf("|       |\n");
       printf("|       |\n");
       printf("|   %s   |\n",Pic);
       printf("|       |\n");
       printf("|       |\n");
       printf("|       |\n");
       printf("|      A|\n");
       printf("---------\n");
}
```

Use the following layout format to define functions to display cards.