Whiskey Poker

06-Final_Project_proposal

Emiliano Peredo A01422326

Raziel Nicolás Martínez Castillo A01410695

Camila Rovirosa Ochoa A01024192

Description

In this assignment we will implement a variation of the game Whiskey Poker, called “Viuda Negra” using different programming topics in order to be able to connect different computers to play the game.

The topics that will be necessary to complete this assignment are: Dynamic memory, Pointers, Process creation and threads.

The rules of the game

The game of Whiskey Poker is composed of a set of different rules from which we had taken the next rules for this project:

1. The game can be played from 2-8 players using a standard 52-card pack without jokers.
2. The objective is to collect the best 5-card poker hand, by exchanging cards with a spare hand dealt to the table and be the last player alive at the end of the game.
3. Before the game each player gets 3 lives (chips). Each round he/she can lose a life or stay with the same. If one player loses all his/her lives he/she is out of the game.
4. At the beginning of the game each player gets 5 random cards and 5 cards are faced-down in the center of the table.
5. The first player in turn can decide if he wants to change all his/her cards with the ones that are in the table. If he does, he has to take the 5 cards and put his previous cards face-up in the center of the table. If he doesn’t, the next player in turn takes the same decision. This is done until one player decides to change his cards. If no one changes his cards then every player shows his/her cards and the winner is chosen based on their hands.
6. If a player changed his/her cards then the game keeps going with the cards in the table faced-up. The next player can change his 5 cards or exchange one with the ones in the table putting one of his cards face-up in his turn and so on with the next players.
7. The game keeps going until one player decides to “Knock” in his turn. The player that stopped cannot make any exchange in that turn. When this happens it means that it is the last round and every player after him has one last turn to exchange cards until they reach the player who stopped.
8. When this finish every player show his cards. The player who has the worst hand lose one life (chip).
9. The last player alive (the one that has 1 or more lives), gets the bets of all the players.

https://www.pagat.com/commerce/whisky_poker.html
https://www.taringa.net/+juegos/reglas-de-juego-de-cartas-viuda-negra_yna1s

The hand types, from high to low, are:

Solution

The functionality of the program will be like this:

- At the beginning of the game each player will be prompted to enter the amount they have to play. After that, they will be prompted to the amount they want to bet. All the players have to agree with this amount and have enough money to play. On each round the players can see their cards and the cards on the table and on their turn they will be prompted with options to exchange a card, all their cards or stop.

- We will create a server and clients using sockets. The server will do all the calculations in order to show to the clients the results. The clients will be where the players connect. The clients will have to connect to the same port of the server and there will be up to 8 clients because, as described in the rules, the maximum players that can play this game is 8. For the calculations we will need to create threads in order to have them done in the order we want based on the turns of each player.
- The clients will be prompted with options to exchange a card, all their cards or stop.
- There will also be an automatic client in order to test the program or if there is only one player. This automatic client will take automatic decisions as another player based on the decisions that the player takes.
- In order to pass on the calculations of the server and the inputs of the players we will use a struct to save the data and this struct will be passed on between the clients and the server.
- To do the calculations in the server we will be passing the structure by reference (as a pointer) between the functions to make it more readable and easier to give parameters to the functions.
- The server has to be aware if a player disconnects or loses in order to not take him/her into account for the next rounds. The server will act as a kind of dealer as well (even though there isn't a dealer in the original rules), in order to keep track of the results and the cards assigned to each player at the beginning of the game.
- We will assign values to each hand and cards in order to determine which one is a better hand, it could also useful to have an array of the 52 cards to keep track of how they are assigned to the player.
- The players will be informed of the current stage of the game, their cards, their balance and what was the result and why.
- We will code the program in C++ in order to be able to use a graphic library from SFML to make a graphic interface for the players.

List of how you will use the topics seen in class

A quick overview of how we will use them as seen in the solution:

- Dynamic memory: We will need to create an array of structs to store the data of each player (cards, viuda, etc.). As there can be a different amount of players each game it will be necessary to have a dynamic array at the beginning of the game.
- Pointers: We will use this by passing a structure as reference between functions in the server to make calculations and are also necessary to read and send between a server and client.
- Process creation: This is also necessary when creating concurrent clients.
- Threads: We will use them in the server to keep track of the clients and make them do operations in the order we want depending on their turn.