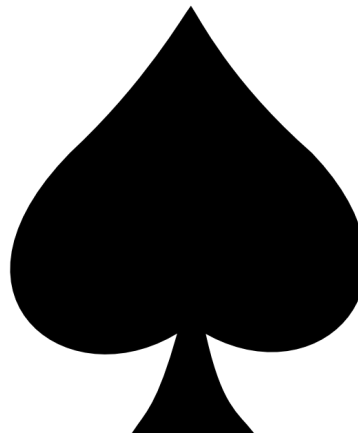
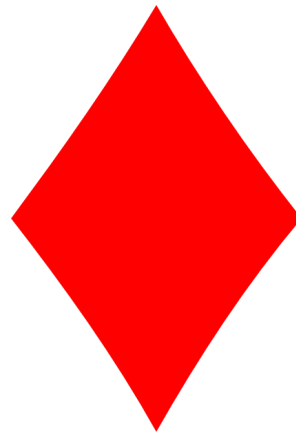
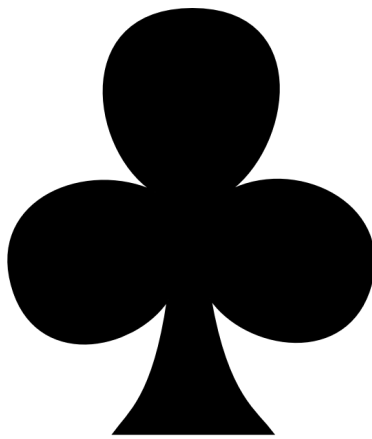


# Unoptimized Poker

**Version 0.5 User Manual By: Loading...™**

**Contributors: Paul Khayet, Jose Diaz, Maher Tarek Homsí, Arhant Katare, Victor Dam, Cameron Bagheri**



**05/09/22**

# Table of Contents

<u>Chapter</u>	<u>Page</u>
A. <b>Table of Contents</b> .....	1
B. <b>Glossary (of online poker)</b> .....	2
1. <b>Online Poker</b> .....	4
1.1 - Usage Scenario .....	4
• Sketch of a typical screenshot .....	4
1.2 - Goals .....	4
1.3 - Features .....	5
2. <b>Installation</b> .....	6
2.1 - System Requirements .....	6
2.2 - Setup and Configuration .....	6
2.3 - Uninstalling .....	6
3. <b>Poker Program Functions and Features</b> .....	7
3.1 - Detailed description of client and server communication .....	7
• User input, program output, screen shot, etc. ....	7
3.2 - Detailed description of dealer choice and card distribution.....	7
• User input, program output, screen shot, etc. ....	7
3.3 - Detailed description of poker game integration.....	7
• User input, program output, screen shot, etc. ....	7
4. <b>Back Matter</b>	
4.1 - Copyright.....	8
4.2 - Error Messages .....	8
4.3 - Index .....	9

## Glossary

- Action -** A fold, check call, bet or raise. For certain situations, doing something formally connected with the game that conveys information about your hand may also be considered as having taken action.
- Blind -** A required bet made before any cards are dealt.
- Button -** A player who is in the designated dealer position.
- Call -** To match, rather than raise, the previous bet.
- Check -** To waive the right to initiate the betting in a round, but to retain the right to act if another player initiates the betting.
- Dead Card -** A card that is not legally playable.
- Dead Hand -** A hand that is not legally playable.
- Deal -** To give each player cards or to put cards on the board. As used in these rules, each deal refers to the entire process from the shuffling and dealing of cards until the pot is awarded to the winner.
- Fold -** When a player decides to back out of the current round, giving up your hand when facing the current bet and current card river.
- Four of a Kind -** Also known as 'quads', a hand + river combination consisting of four cards of equal value, and one extra card.
- Flush -** A poker hand consisting of five cards of the same suit
- Full House -** A hand consisting of three of a kind and a pair
- High Card -** The lowest hand rank in poker. It is a hand of five cards that doesn't have two matching cards or all cards of the same suit.
- Pair -** A pair is a poker hand where we have two cards of identical rank.

**Royal Flush** - Consists of the ace, king, queen, jack and ten of a suit. A royal flush is an unbeatable hand.

**Raise** - To increase current bet for the round, in which the other players must either match or fold.

**Shuffle** - The act of mixing the cards before dealing out a hand

**Straight** - Five cards in consecutive rank

**Straight Flush** - A rare hand + river combination consisting of any straight of the same suit.

**Three of a Kind** - Also known as 'trips', a hand + river combination consisting of three cards of equal value, and two extra cards not of equal value.

**Two Pair** - A hand + river combination that contains two cards of equal value, another two cards of equal value, and one extra card.

# 1

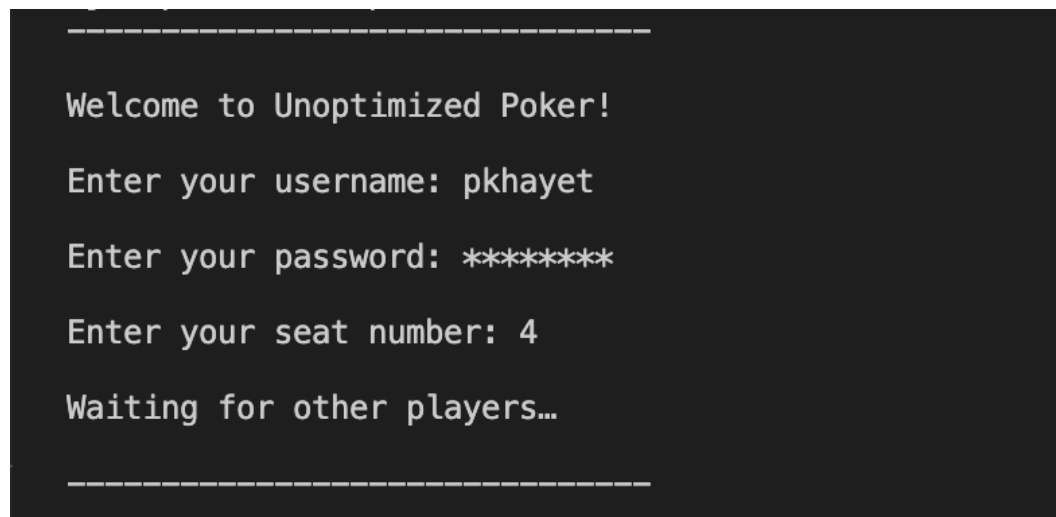
## Online Poker

### 1.1 Usage Scenario:

The usage scenario for the first implementation of our program design includes an interaction between a client and server through a socket. For the initial version, the client first sends the command Enter, followed by the name to be referred to, followed by 'SEAT' and the number, followed by 'PASSWORD' and the password to establish the user's seat location at the beginning of the game.

ENTER Kevin SEAT 3 PASSWORD unoptimized.

- *Sketch of a typical screenshot*



```
-----  
Welcome to Unoptimized Poker!  
Enter your username: pkhayet  
Enter your password: *****  
Enter your seat number: 4  
Waiting for other players...  
-----
```

### 1.2 Goals

Our goal at Loading...™ for the initial version is to provide a clear, easy-to-use interface that employs a central host organizing a game of poker (Texas Hold'em). The game would permit guest users to join the central user who hosts this game of poker, type their names, and then select their seating location. After this selection, dealing would commence as each user would receive 2 cards. For each round of the game, player number one would make an initial bet, upon which subsequent players can fold, raise, or call. As rounds progress and community cards are placed at the center of the table with their faces showing. Then, during the showdown phase, each player can compare their 'hands' to determine who has the highest score. There is no real money involved in our implementation, but players may still play either for leisure or competitively. Goals for the program are that it does not result in a segmentation fault, and permits players to fold, call, or raise appropriately, as well as calculates the user with the highest

score during showdown. In addition, the program should not suddenly break, end the game abruptly or freeze during gameplay.

### *1.3 Features:*

Features of our initial release include the following:

Users can connect to the game being hosted by the central user.

Users may select their seats and type in their names that shall be displayed during the game.

Card dealership should have random shuffling, and a user is chosen randomly to become a dealer.

Program should update the accounting based on the amount a player is betting and the values of the cards they are currently holding.

The showdown stage should begin after all the community cards are revealed. The players must reveal their hand in this situation, and their hands are compared to determine the winner.

For the most basic implementation of our program design, we desire a game with the following characteristics. A simple GUI displaying the users cards, the table's cards, and their options (Bet, Call, Raise). However for our initial release these options might not fully work as intended.

For the future, our program might consist of other advanced/bonus features that are desirable such as: a simple timer, a scoreboard at the end showing player rankings, etc.

## 2

# Installation

## 2.1 System requirements

This program requires the following:

1. Linux System Version (kernel release information is version 2.6.32-754.33.1.el6.x86\_64 or later)
2. GCC Compiler
  - a. g++ (GCC) 4.4.7 20120313 (Red Hat 4.4.7-23) version or later
3. Keyboard, and a Screen
4. Internet Access for client-server interaction

## 2.2 Setup and Configuration

1. To unpack the user tar package while using the Linux terminal, use the command 'gtar Poker\_Alpha.tar.gz.' and 'gtar Poker\_Alpha\_src.tar.gz.' to extract the tar balls in your current directory.
2. Once the files are extracted in the correct folder, open the console and make sure you are in the correct directory. Then proceed by typing 'cd SourceCodePackage' .
3. Use the command 'make' to compile the poker game without the GUI or use 'make poker' to play the game.
4. Type 'cd bin' to reach the bin folder.
5. Once the make file has been created type the command './poker' in order to start the game!

## 2.3 Uninstalling

Uninstallation would proceed by typing 'sudo apt-get --purge remove Poker\_Alpha\_src.tar.gz' in the Linux terminal.

For the time being, to uninstall the program, simply type 'make clean' . This will remove the executable file called 'poker' from the bin folder

Then type the command 'ls' to see all of the files inside of the folder, and then type 'rm filename' for all of the files.

### 3

## Poker Program Functions and Features

### 3.1 Detailed Description of client and server communication

- *User input, program output, screen shot, etc.*

Our program is split into three main parts. One part is reserved for the actual games and rules of poker, having data structures for gameplay (this part will be on the server). Another part of the program deals with the communication between the client and server. The last part of the program will be the client's side of the game where they will see the GUI and be able to input all of the necessary information to get the game started. The program will use sockets for client-server communication within the game. We have opted not to include a screenshot until further details are finalized in the future.

### 3.2 Detailed description of dealer choice and card distribution

- *User input, program output, screen shot, etc.*

Our program will identify a random dealer out of the 6 or 7 users playing the game. Once the dealer is identified, they would distribute 2 cards to each client in the order of seats they picked. The cards will be dealt to the players and randomly shuffled. The GUI will also have a central area displaying the community cards that are face up for the showdown stage. We have opted not to include a screenshot until further details are finalized in the future, but a display of a typical game might look like the following.



### 3.3 Detailed description of poker game integration

- *User input, program output, screen shot, etc.*

The server will know everyone's cards, point values, and seat number. When clients fold or raise, the server would update their specific data. The GUI would then show the specific cards each client is holding which would be called from the server. The river would also have 5 cards which would be known to the server, and once everyone has folded, the GUI would flip the river cards, which is called from the server. We have opted not to display a screenshot until further details are updated in the future.



## 4

# Back matter

### 4.1 Copyright

Copyright © 2022 Loading...™. All rights reserved. Developed in the United States of America. For usage in binary and source forms, modified or unmodified, the source code may be redistributed having satisfied the conditions below:

\*Source code redistributions should display the copyright notice above, these asterisked stipulations, and the disclaimer below.

\*Binary code redistributions must display the copyright notice above, these asterisked stipulations, and the disclaimer below as well as other materials offered with the distribution.

\*Neither the name of contributors of Loading...™ nor that of itself are to be used to endorse items developed from this software unless permission has been granted.

UNDER NO CIRCUMSTANCE SHALL CONTRIBUTORS OR COPYRIGHT OWNERS BE HELD RESPONSIBLE FOR DAMAGES THAT WERE CONSEQUENTIAL, INCIDENTAL, INDIRECT, OR DIRECT (SUCH AS LOSS OF DATA, PROFITS, OR PROCUREMENT OF SUBSTITUTE GOODS) FOR ANY REASON AND REGARDING ANY THEORETICAL LIABILITY, RESULTING IN ANY WAY FROM THE USAGE OF THIS PROGRAM, EVEN IF ENCOURAGED OF SUCH POTENTIAL DAMAGE.

### 4.2 Error Messages

Error messages are displayed if the client types a message that is not a command in the socket interface. In this case, the message below is displayed to the user should the user type an input that is not one of the options provided to them.

The error message for inputting an invalid option.

- **Communication Example: Initial Poker Protocol**
  - **Client:** Hello!
  - **Server:** ERROR invalid message "Hello!"

The error message to deny the user to check if another player raises a bet he is unable to match, but does not want to fold.

- **Communication Example: Initial Poker Protocol**
  - **Client (User 1):** raise 50
  - **Server:** ERROR invalid move made by user 1

### 4.3 Index

#### A

Action 2

#### B

Blind 2

Button 2

#### C

Call 2, 4, 5, 7

Check 2, 8

#### D

Dead Card 2

Dead Hand 2

Deal 2, 4, 5, 7

#### F

Flush 2, 3

Fold 2, 3, 4, 7, 8

Four of a Kind 2

Full House 2

#### G

GCC Compiler 6

Goals 4

Graphical User Interface 5, 6, 7

(GUI)

#### H

High Card 2

#### I

Internet 6

#### L

Linux System 6

#### M

Main 7

Make Clean 6

#### P

Pair 2, 3

#### R

Royal Flush 3

Raise 2, 3, 4, 5, 7, 8

#### S

Socket 4, 7, 8

Shuffle 3, 7

Straight 3

Straight Flush 3

#### T

Three of a Kind 2, 3

Two Pair 3

#### U

Uninstall 6

## 4.4 References

### Poker\_Handout

Details on required features for this program were provided by T.A. Vivek Govindasamy, Yutong Wang, and Prof. Rainer Dömer in the Project 2 Handout.

<https://www2.gov.bc.ca/assets/gov/sports-recreation-arts-and-culture/gambling/licences/poker-definitions.pdf>

Contains information about common poker terms

<https://www.arkadium.com/games/texas-holdem-poker-sit-and-go/?playAgain=>

Contains a sample GUI that we may base our program on

<https://www.winzip.com/en/learn/tips/extract-files/tar-gz-file/>

Contains information that we used in our setup and configuration section

<http://upswingpoker.com/poker-hands-rankings/>

Contains information regarding different hands in poker.

<https://go.dev/doc/install>

Contains copyright notice of Go Installer that we modeled our copyright message after.

<https://www.howtogeek.com/229699/how-to-uninstall-software-using-the-command-line-in-linux/>  
<https://askubuntu.com/questions/939923/how-to-uninstall-software-that-was-installed-by-make-install>

Contains information on uninstallation in Linux.

[https://techterms.com/definition/system\\_requirements](https://techterms.com/definition/system_requirements)

Contains info about system requirements