

Crazy Eights Card Game

Leah Omaiye

CSC 5

Spring 2017

42641

Introduction

Rules of Crazy Eights

Goal:

The goal is to empty your hand.

Number of Players:

The game can be played with two or more people. Each person gets eight cards. For a two player game, each person gets seven cards. If there are more than five players, two decks are used.

Gameplay:

The deck is shuffled and one card is flipped face-up. Players must match one of their cards with either the suit or the value of the face-up card or play an 8 of any suit. The other players must then match that suit or play another 8. If a player does not have a playable card, they draw from the deck until they do. The game ends once a player empties their hand, or the deck empties.

Scoring:

For two players; the winner is the first person to reach 100 points.

For three players; the winner is the first person to reach 150 points.

Four players; first person to reach 200.

Five players; first to reach 250

6+ players; etc.

Translating This to C++

In general, I was going to use the regular rules of the game. However, I wanted to start with a two player game first and the first user to empty their hand wins. I was going to include the scoring to determine the winner after I had successfully programmed the hands arrays, but the logic required for this project proved to be quite a challenge. There were also additional rules for the Zimbabwean version of the game (similar to UNO) that I eventually wanted to include but couldn't.

Flowchart

found here:

https://www.draw.io/?lightbox=1&highlight=0000ff&edit=_blank&layers=1&nav=1#G0Bydj5iTYWBpYM2dWc1VSZVIEWUU

References

Starting Out with C++: From Control Structures through Objects. Gaddis, Tony. 8th Edition.

Crazy Eights. https://en.wikipedia.org/wiki/Crazy_Eights.

PseudoCode

```
/*
 * File:  main.cpp
 * Author: Leah Omaiye
 * Created on May 31, 2017 11:11 PM
 * Purpose: Crazy Eights Project v6
 */

//System Libraries
//Input - Output Library
// use random_shuffle
//file I/O
//format output
//use srand to set seed
//time for rand
//string
//Name-space under which system libraries exist

//User Libraries

//Global Constants

//Function Prototypes
//display rules of game
//initialize deck with 52 cards

//function to get top card
//function to get suit
//function to get card number
//function to deal hand
//function for players to draw card
//function to calculate score

//Execution begins here
//display rules of game first

//set rand num seed

//Declare and initialize variables
//input file
//output file
```

```

//string name of players
//p1 and p2's cards
//players' choices
//size of hand array
//7 cards for each player
// initial deck starts with 52 cards
//cards left in deck after dealing/drawing
//deck as a pointer
//players' hands as pointers

//Input data
//file to store scores and player names
//function to initialize deck

//function to shuffle deck

//top card dealt from deck
//represent actual card number as value of card and suit

//functions to deal hands to players
//populate p1 hand array
//display actual card number
//display value of card and suit
//same for p2 hand array
//players choose to draw a card or play a card
//players choose which cards to play from their hand
//check to see if card is playable
//compare number of cards in players' hands
//least number of cards wins
//calculate score
// delete pointers

//Exit stage right!
//close files

//display rules
//function to initialize 52 cards in the deck
//function to get top card and deal cards
//deck size shrinks as cards are dealt
//function to get suit of card
// function to get value of card
//function to populate hand pointer with dealt cards
//function to calculate score

```


Program: