

Programming Assignment #1

Crazy Eights!

CS 3358.253, Spring 2015

Instructor: Jill Seaman

Due: Wednesday, 2/11/2015 (upload electronic copy by 1:30pm)

Problem:

You will write a program that simulates the card game "Crazy Eights". We will use the following slightly simplified version of the game:

How to play: In a two-player game, each player is dealt seven cards. The rest of the deck goes facedown in a pile, with the top card turned up beside it. This is the discard pile. The player to the left of the dealer discards a card from his hand that matches either the number or suit of the top card in the discard pile. For example, if the card is a five of hearts, he could play any heart or any five. If he does not have a matching card, he continues picking up cards from the deck until he gets one that is playable. Eights are wild and can be put down on any suit. For example, an eight could be played to match a heart. The next player must match their card to the suit on the eight that was played. Play continues with players matching the card at the top of the discard pile. The first player to use up all his cards wins. If the deck runs out before the game is over, the game ends in a draw.

The main program will simulate the playing of the game, with two players.

Classes: The classes needed to support the main program are listed here. The following files will be available on the class website:

card.h - simulates one playing card (a suit and a value).

deck.h - simulates a deck of 52 cards

player.h - simulates a player in the game

card_demo.cpp - a demo of a main program that will deal some cards

The deck contains an array of cards, and the player contains a vector of cards. Do not change the *.h files.

You will need to add an implementation file for each of the header files, and implement a new driver called crazy_eights.cpp, which simulates the playing of the game.

```
Harry has : 2d 9d Ks Jh 6h 5s 7c
Ron has : 2h 7s Kc Ah 5h Qs 5d
The discard pile contains: Jc
Harry plays a Jh
Ron plays a 2h
Harry plays a 2d
Ron plays a 5d
Harry plays a 5s
Ron plays a 5h
Harry plays a 6h
Ron plays a Ah
Harry draws 10d
Harry draws 6c
Harry draws 10c
Harry draws 10h
Harry plays a 10h
Ron draws 3h
Ron plays a 3h
Harry draws As
Harry draws 3c
Harry plays a 3c
Ron plays a Kc
Harry plays a Ks
Ron plays a Qs
Harry plays a As
Ron plays a 7s
Game over
Ron is the winner.
```

Output: For the output of the main program, you should print each play to the screen and indicate the eventual winner, or if the game ended in a draw.

Note: The computer will play the hands of both players. There is no input from the user.

NOTES:

- Start by implementing the Card class and testing it. Then the Deck class. Then the Player. You should have a test driver for each class (but don't submit them).
- I recommend using a makefile, but it's not required. We will cover samples in class.
- The purpose of this assignment is to remember how to compile and execute programs (objective iii), review classes and multi-file programs (objectives i and iv), and to get some practice using the vector template from the Standard Template Library.

- Some topics you may need to learn for this assignment (we will cover them in class):
 - rand (for random number generation)
 - enum types
 - assert
 - vectors
-

Style:

See the Style Guidelines document on the course website.

Note: Put function description headers in your *.cpp files (since you are not changing/submitting the *.h files).

Logistics:

Since there are multiple files for this assignment, you need to combine them into one file before submitting them. You can use the zip utility from the Linux/Unix command line:

```
[...]$zip assign1_XXXXXX.zip crazy_eights.cpp player.cpp  
deck.cpp card.cpp
```

This combines the 4 files into one zip file, assign1_XXXXXX.zip. Then you should submit only assign1_XXXXXX.zip.

Name your file **assign1_XXXXXX.zip** where XXXXX is your TX State NetID (your txstate.edu email id). It should look something like this: assign1_js236.zip.

Submit: an electronic copy only, using the Assignments tool on the TRACS website for this class.