CSE 231 **Fall 2013**

Computer Project #8

Assignment Overview

This assignment focuses on the design, implementation and testing of a Python program which uses classes to solve the problem described below.

It is worth 50 points (5% of course grade) and must be completed no later than 11:59 PM on Monday, November 11.

Assignment Overview

You will design and implement a Python program which provides the infrastructure for a user to play the card game FreeCell. The game rules to be used for this assignment are:

> **Start:** All 52 cards are dealt into eight columns (the tableau), where four of the columns contain seven cards each and four of the columns contain six cards each. All of the cards are visible.
>
> In addition to the tableau, there are four foundation piles (initially empty) and four free cells (initially empty). The foundation piles are used to hold all of the cards for a given suit, in order from Ace to King. Each of the free cells can hold one card during game play.
>
> **Objective:** The object of the game is to build up each foundation, in sequence, from Ace to King. The player wins if he is able to move all 52 cards to the four foundations.
>
> **Game Play:** Only one card may be moved at a time. Only the bottom card of each of the eight tableau columns and any cards in the free cells are eligible to be moved.
>
> To move a card from the tableau or a free cell to a foundation pile, it must be the correct suit and it must have the correct rank (since each foundation pile is built up from Ace to King within a given suit). Once a card is moved onto a foundation pile, it cannot be moved again.
>
> To move a card from the tableau or a free cell to the bottom of a given column in the tableau, it must be the correct color and it must have the correct rank (since the tableau columns are built down by rank and by alternating color). For example, only the Two of Hearts or the Two of Diamonds can be moved onto the Three of Spades (a Two is one less than a Three, and red alternates with black).
>
> Any card may be moved from the tableau to an empty free cell (but each free cell may only hold one card at a time).
>
> Any card may be moved from the tableau or the free cells to an empty column in the tableau.
>
> The game is over when the player wins (by moving all 52 cards to the foundations), loses (can no longer make any moves), or resigns.

Your program will permit the user to play FreeCell and will enforce the game rules.

Assignment Specifications

1. Your program will use the Card and Deck classes found in the file named “cards.py”. You may not modify the contents of that file.

2. Your program must be subdivided into meaningful functions.

3. Your program will recognize the following menu of options:

```
t2f T  F  -- move from Tableau T to Foundation F
t2c T  C  -- move from Tableau T to Cell C
t2t T1 T2 -- move from Tableau T1 to Tableau T2
c2t C  T  -- move from Cell C to Tableau T
c2f C  F  -- move from Cell C to Foundation F
h -- help (displays the menu of options)
q –- quit
```

   All foundations and tableau columns will be entered as integer numbers.

4. Your program will detect, report and recover from the following errors:

   a. Attempt to move a card in violation of the game rules
   b. Invalid menu option
   c. Invalid operands for menu options

5. Your program will display the free cells, foundations, and tableau after each move. The display will be appropriately formatted (use an existing FreeCell program as a guide). Your program does not need to print every card in the foundations – printing only the top card is sufficient.

6. Your program will check after each move and determine if the user has won. If so, your program will display an appropriate message and halt.

Assignment Deliverables

The deliverable for this assignment is the following file:

proj08.py – the source code for your Python program

Be sure to use the specified file name and submit it for grading via the **handin** system before the project deadline.

Reminder: your source code must follow the CSE 231 Coding Standard, including the required comments (Header, Program Overview and Function Overview).

Assignment Notes

1. There is a lot of information about FreeCell on the web. For example:

   http://www.solitairecentral.com/rules/FreeCell.html

   http://solitairelaboratory.com/fcfaq.html

2. You would be wise to familiarize yourself with the rules by actually playing the game. One website that allows you to play FreeCell online:

   http://games.aarp.org/games/freecell/freecell.aspx

3. The Card class and Deck class are provided in the file named “cards.py”. Those classes are general purpose, so they contain methods which may not be necessary for this project.

4. You will not submit “cards.py” as part of your solution. Instead, each student’s solution will be tested using the instructor-supplied file. That implies that you may not alter the contents of “cards.py” in any way.

5. Examine the file named “cardsDemo.py”, which contains a Python program which imports the “cards” module and demonstrates some of the class methods.

6. The file named “freecellStart.py” contains a framework which you might find useful for your program. Note that you are not required to use that framework.

7. You would be wise to subdivide your program into multiple pieces, such as game setup, game play, display, and so on. You can work on each piece in isolation, then assemble the pieces to form your program.

8. Initially, you may wish to assume that all user input is error-free. Develop the main logic of your program, then handle errors. Please note that error handling may require quite a bit of code.