

FIT9133 Assignment 2: Not Freecell (20%).

Due: 23:55:00, Friday 22th September, 2017.

1 Description

The powers that be are very impressed by your passport generator and have sought your skills once again. You are tasked with making Not-FreeCell, a freecell clone. Freecell is a solitaire based game that comes standard with most Windows systems.

Freecell can be played at <http://www.123freecell.com/>. You may also wish to read the article on wikipedia: <https://en.wikipedia.org/wiki/FreeCell>.

In short, Freecell is made up of four foundations where you stack cards in their suits, four cell slots that can hold any cards, and cascade slots that holds all 52 cards stacked on top of each other.

1.1 The Rules

The game follows the following rules:

- Only one card may be moved at a time.
- The four foundations must be built starting from the Ace of the appropriate suit, followed by the 2, then the 3 etc. until the King is placed.
- If a card is placed on a cascade, it must be placed of a card of the opposite colour, and of a suit that is one higher than itself. E.g. a red 2 can be placed on a black 3, but not a black 4.
- Any card may be placed in an empty cascade.
- Cards from the foundations may be placed back onto a cascade, or an empty cascade slot.
- Only one card at a time can occupy a cell slot.

Victory is achieved when all four foundations are filled with their respective suits from Ace to King.

1.2 The Requirements

As part of your program, you will need the following:

- A class called “Card” that represents a card in play. “Card” must also contain the following:
 - The attribute “card_face” and “card_suit” that contain the face value and suit of the card respectively.
- A class called “Deck” that is made up of cards. “Deck” must also contain the following:
 - An `__init__(self, value_start, value_end, number_of_suits)` so that a deck of cards can be constructed using a known start value, a known end value, and a known number of suits. E.g. `Deck(1,11,1)` will generate a deck consisting of Ace to 9 and Jack in a single suit, thus a deck of 11 cards.
 - A shuffle method to randomize the cards.
 - A function to add a card to the deck.
 - A function to draw a card.
- A class called “NotFreecell” that uses a deck and implements the rules of freecell.
- “NotFreeCell” should be playable, and thus is able to create new games and know when the game has ended as a result of victory.

- It is NOT necessary to check if the game ends as a result of no actions, NOR do you need to make a Graphical User Interface (GUI). You may do these things but they are not worth marks.

In addition to the above, you will need to add functions and attributes to all of the classes that come standard with object construction such as accessors and mutators.

1.3 A Report

Unlike in Assignment 1, Assignment 2 is complicated enough to warrant the submission of a report. This report should detail and outline all design choices such as attributes and functions.

2 Marking Guide

I personally do not like writing strict marking guides, however it is a policy that I do, and people will complain if I don't. So here you go:

1. Functionality (30%):
 - (a) Freecell rules are implemented correctly.
 - (b) Game is able to be played.
 - (c) Testing is demonstrated in an appropriate manner.
2. Code Architecture (40%)
 - (a) str and len are implemented appropriately.
 - (b) Classes are comprised of appropriate attributes.
 - (c) Methods in classes are useful and justified.
 - (d) Classes are imported to allow for implementation.
3. Coding Style (15%)
 - (a) Uses appropriate spacing.
 - (b) Has appropriate variable names.
 - (c) The logic is clear and meaningful.
4. Documentation (15%)
 - (a) Explains the purpose of variables.
 - (b) Documents complicated parts of code.
 - (c) Fully documents functions
 - (d) If there needs to be user documentation for programs use then it is well explained.
 - (e) Report is descriptive and explains code choices.

Note: All of the above is assuming that the code is effective. Meaning, if you just write a while loop that prints an index over and over, it won't count. Your code must do something towards the goal of the assignment.

3 Submission

You should submit this assignment in the form of one or multiple python scripts zipped together. While you may want to do your testing and development in notebooks, it must function in a python script.

Please submit your file with the name "Assignment2{STUDENTID}{STUDENTNAME}.zip" where STUDENTID is your student ID without the brackets, and STUDENTNAME is your name without the brackets.