

Assignment #4

Working Games

(100 pts) You are provided a problem statement that you must solve by writing a computer program. For each programming assignment, you will carry out Polya's 4 steps for solving problems:

1. Understanding the problem (*Recognizing what is asked.*),
2. Devising a plan (*Responding to what is asked.*),
3. Carrying out the plan (*Developing the result of the response.*),
4. Looking back (*Checking. What does the result tell me?*),

(15 pts) You are provided a problem statement with each programming assignment, and you are required to turn in a written document (**as a pdf**) addressing Polya's steps to solving a problem with step 3 being the Java code you write to carry out or implement your plan. With this said, your written document must include these three sections:

Understanding the Problem

In your own words, explain what YOU think the problem is asking you to do. In this section, document your uncertainties about the problem and anything else that you feel was unclear or vague. This is to ensure that your understanding matches my understanding of the problem

Devising a Plan/Design

At a minimum, provide an algorithm, pseudo code, or diagram you designed to help solve the problem. In addition, include pictures or flow charts you used to help you devise your plan, as well as any other design decisions you made such as how to manage your time, how to decompose the problem, where to start first, etc. You can scan any handwritten work and attach it to the document as needed.

Looking Back/Self-Reflection

Report any checking or self-reflection you did while solving the problem. For instance, how did you make sense of the output from the implementation? This includes things such as using a calculator to make sure the output is correct, testing to make sure your code executes correctly and behaves the way you expect under specific circumstances, using external sources of information such as the internet to make sense of the results, etc. Also, include a statement about what you learned from the assignment.

(75 pts) Problem Statement:

Write a program to:

1. Display a menu for users to select a game to play (listing at least the provided guessMyCard game and the first part of the blackjack program described below)
2. implement a working blackjack program as follows:

(<http://en.wikipedia.org/wiki/Blackjack>)

- a. your program implements the **Game interface** similar to the way the provided GuessMyCard program does (be sure to use the keyword implements and be sure to actually implement the methods that are required by the interface),
- b. your blackjack program has a **Card class** that can be used to create objects that represent cards (a suit, a face value, and whether it is face up or face down) that includes the following method:

public String showing()

This method will return either the card's value and suit if the card is showing, or a string indicating that the card is face down,

- c. the play method (as required by the Game interface) will deal two cards to a player and deal two cards to the dealer, where one of the dealer's cards is face down from the point of view of the player **and allow the player to decide to hit or stay, only disallowing the hit option when the player busts (over 21 total) or decides to stay.**
3. **Note:** Now your blackjack game **has** to play blackjack, it needs to be able to deal two cards to a player, deal two cards to a dealer, and show the appropriate values to the user (the visible cards), **only showing the value of the pair of cards the dealer was dealt after the player has decided to stay or has gone bust,**
4. So your program needs to do the following:
 - a. Your program should allow users to play games repeatedly,
 - b. deal cards to the player and dealer (only show the face-up card of the dealer to the user, or you can choose not to display these at all until after the player has gone),
 - c. allow the player to decide whether to hit or stay (a player is allowed to hit repeatedly),
 - d. your blackjack program displays the total of the cards (remember that Aces can be 1 or 11),

Extra credit considerations:

1. (up to 10 points) Allow there to be multiple players at the table (having each one decide to hit or stand and play the dealer's hand after all players have decided),
2. (up to 10 points) Create a game (or multiple games) of your own design (implementing the Game interface) and include it as an option for users to play as well as Blackjack.

(10 pts) In your implementation, make sure that you include a program header in your program, in addition to proper indentation/spacing and other comments!

```
/******  
** Program: LargestInt.java  
** Author: <Your Name>  
** Date: <today's date>  
** Description: <what your program does>  
** Input: <what input is necessary for the program to execute properly>  
** Output: <what output is expected when your program is executing properly>  
*****/  

```