California State Polytechnic University, Pomona **Name___________________**
Computer Science Department
CS141: Introduction to Programming and Problem-Solving
Winter 2017
Instructor: Prof. Tony Diaz

Program 5 - Due Mar. 1

(Total: 100 pts)

Write an abstract class titled "YourLastName_BankAccount" to hold the following data for a bank account:

- Balance
- Number of deposits this month
- Number of withdrawals
- Annual interest rate
- Monthly service charges

The class should have the following methods in addition to the regular access methods for the instance variables (set/get):

constructor: The constructor should accept arguments for the balance and annual interest rate.

deposit: A method that accepts an argument for the amount of the deposit. The method should add the argument to the account balance. It should also increment the variable holding the number of deposits.

withdraw: A method that accepts an argument for the amount of the withdrawal. The method should subtract the argument from the balance. It should also increment the variable holding the number of withdrawals.

calcInterest: A method that updates the balance by calculating the monthly interest earned by the account, and adding this interest to the balance. This is performed by the following formulas:

Monthly Interest Rate = (Annual Interest Rate / 12)
Monthly Interest = Balance * Monthly Interest Rate
Balance = Balance + Monthly Interest

monthlyProcess: A method that subtracts the monthly service charges from the balance, calls the calclnterest method, and then sets the variables that hold the number of withdrawals, number of deposits, and monthly service charges to zero.

Next, design a SavingsAccount class that extends the BankAccount class. The SavingsAccount class should have a status field to represent an active or inactive account. If the balance of a savings account falls below $25, it becomes inactive. (The status field could be a boolean variable.) No more withdrawals may be made until the balance is raised above $25, at which time the account becomes active again. The savings account class should have the following methods:

withdraw: A method that determines whether the account is inactive before a withdrawal is made. (No withdrawal will be allowed if the account is not active.) A withdrawal is then made by calling the superclass version of the method.

deposit: A method that determines whether the account is inactive before a deposit is made. If the account is inactive and the deposit brings the balance above $25, the account becomes active again. A deposit is then made by calling the superclass version of the method.

monthlyProcess: Before the superclass method is called, this method checks the number of withdrawals. If the number of withdrawals for the month is more than 4, a service charge of $1 for each withdrawal above 4 is added to the superclass field that holds the monthly service charges. (Don't forget to check the account balance after the service charge is taken. If the balance falls below $25, the account becomes inactive.)

Create a driver which will create one instance each of the previous two classes and tests them. Create a loop so that the user is repeatedly asked for which account they would like to access, and then is prompted for which action they would like to perform, withdraw, deposit, calculate interest, or monthly processing. For each selection, the program prints out the result or an informative error message if there is a problem (ex. trying to withdraw more than the account balance). The program should continue looping, even if there is a problem, exit the loop when data entered for the bank account type start with characters "q" or "Q". Don't print out an error message in this case, just immediately exit the program.

The following shows an example interaction captured in a file by the command “% script BankAccount.out” (bolded areas represent the user's input):

```
Script started on Thu Sep 26 10:23:58 2013
% java Diaz_BankDriver

Which account would you like to access, regular or savings?: Regular
What action do you wish to perform
(Withdraw, deposit, monthly processing)?: deposit
Enter amount to deposit: 254
Account balance is $254.

Which account would you like to access, regular or savings?: regular
What action do you wish to perform
(Withdraw, deposit, monthly processing)?: Withdraw
Enter amount to withdraw: 100
Account balance is $154.

Which account would you like to access, regular or savings?: regular
What action do you wish to perform
(Withdraw, deposit, monthly processing)?: Withdraw
Enter amount to withdraw: 160
Error - Not enough funds.

Which account would you like to access, regular or savings?: savings
What action do you wish to perform
(Withdraw, deposit, monthly processing)?: monthly processing
Account balance is $15.
Savings account is now inactive.

Which account would you like to access, regular or savings?: savings
What action do you wish to perform
(Withdraw, deposit, monthly processing)?: deposit
Enter amount to deposit: 20
Account balance is $35.
Savings account is now active.

Which account would you like to access, regular or savings?: Quit
```

```
% exit
script done on Thu Sep 26 10:24:17 2013
```

Run the program three times with different input each time and capture all interaction in a file using the script command.

What to turn in:

- Soft copy of the script command (using Blackboard)
- Soft copy of the program (submit your .java file using Blackboard)