

## Fun with Bank Accounting Assignment #2 - Analysis

The problem is to develop a class hierarchy of bank accounts, checking accounts, and savings accounts, and then start interacting with transactions of adding money, withdrawing, etc. At the end of the process, a summary of the final balances for the four accounts that need to be created (a checking account, three savings accounts for primary savings, student loan repayment, and auto loan repayment) is generated for all customers. We are using JOptionPane dialogue input, summary, and error boxes. A Customer class is necessary to interact with the bank accounts, and has an ID, a name, and an address. A driver will run the program.

Bank accounts have account numbers, owners ID, and current balance. Actions are depositing money, withdrawing, and getting the account balance. A negative balance is not an option. Savings accounts have the ability to add an interest rate so interest can be added. Checking accounts have the additional option of overdraft protection by drawing from the primary savings account, if the balance of the savings account can support it, and the savings account is charged \$20 for each transfer. No fees for checking account, no minimum balance and no interest, kind of like real life.

For savings accounts, interest is gained via an interest action is posted, but only if \$1000 is the minimum balance. The interest for all savings accounts is 4%, and it's only for savings accounts.

Finally, illegal transactions with illegal customer ID, illegal transaction type, negative input values for the amounts, and illegal account type codes. Don't quit though, process the error and go to the next transaction.

Input is as follows: Customer ID#, transaction-type, amount, account-type, account-type 2 (for transfer of funds from account 2).

## Assignment #2 Design IPO Model

Process add, withdrawal, interest if applicable, and  
get balance transactions, and then display final  
account balances at the end.

Customer ID# (single digit)

Transaction type (D, W, I, T, G)

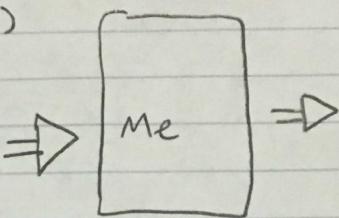
D  
W  
I  
T  
G

Amount (\$)

Account Type (C, S, L, A)

C  
S  
L  
A

Account Type 2



Option pane dialogue boxes with summary of transaction and another with "continue?" message

Error dialogue box for illegal transaction or input

Updating account balance variables based on transaction inputs

Summary dialogue box with final account balances

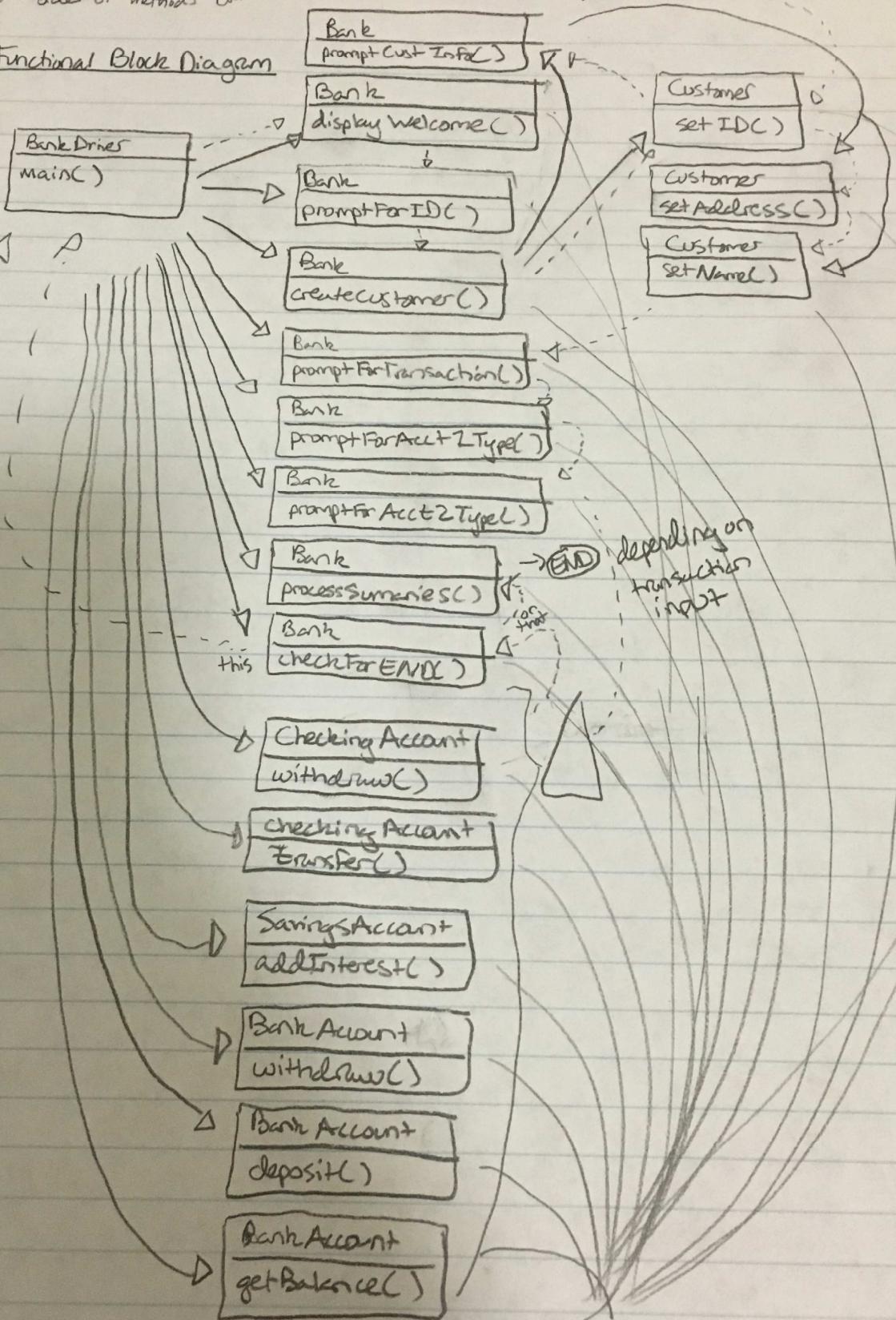
### Requirement

Banking Account

Checking Account      Savings Account

Solid arrows = calling relationships  
dotted arrows = order of methods called by drivers

## Functional Block Diagram



### Driver Logic Pseudocode

Display welcome message

Ask for customer ID # and make sure it's valid (positive), otherwise just skip to asking if there's another transaction

Ask for customer info if they're a new customer (name and address)

If invalid input, we don't care. They don't have to provide a name and address, just leave it blank.

Ask for transaction type and make sure it's either T(transfer), G(get balance), I(interest), D(deposit), or W(withdraw), otherwise ask if another transaction is made

Ask for amount, if applicable, and make sure it's valid

Ask for account type, and make sure it's valid

Ask for second account type if a transfer occurs and make sure it's valid

If all inputs were satisfied, proceed with transaction and I/O

If deposit occurs, then deposit amount into account and display result

If withdraw occurs, either it succeeds or doesn't.

If the account is a checking account, check to see if overdraft occurs, and if it can be cleared by withdrawing from primary savings. If it can't go through on its own or with savings, display it failed. Otherwise displays success.

If savings account, display that withdrawal worked or didn't.

If interest occurs, add interest to a savings account if minimum balance is met and display result. If account is checking, display transaction isn't supported.

If get balance occurs, get balance of account and display it.

If transfer occurs, try to withdraw money from first account (following overdraft protections for checking accounts) and if successful, deposit it into the second account, and display results; if it failed, succeeded, or succeeded with help from the primary savings account in the instance of withdrawal from a checking account

Ask user if more transactions are to be made

If yes, go back to asking for customer ID and repeat rest of program

Otherwise, for each customer, display their name, ID, address, and account balances for all their accounts

End of driver.