Same submission requirement as before. Your program must contain proper documentation. Every function must have function documentation header (see example from printMeFirst() function.

You must separate .h file from .cpp file. You must use Makefile. You .h file must use safeguard (use #ifndef ….. #endif)

Below is details of this lab:

Bank Accounts

Implement a class Account. An account has a balance, functions to add and withdraw money, and a function to query the current balance.

```
class Account
{
public:
   Account();
   Account(double bal);
   void deposit(double amount);  // need to implement member function program
   void withdraw(double amount);
   double getBalance() const;

private:
   double balance;
};
```

Implement a class Bank. This bank has two objects, checking and savings, of the type Account. Implement member functions:

```
 class Bank
{
public:
   Bank();
   Bank(double checkingAmount, double savingsAmount);
   void deposit(double amount, string account);
   void withdraw(double amount, string account);
   void transfer(double amount, string account);
   void printBalances() const;

private:
   ????? checking;   // change ????? with correct data type variable checking
   ??????? savings;  // use correct data type for variable savings
};
```

Charge a $5 penalty if an attempt is made to withdraw more money than available in the account.

Here the account string is "S" or "C". "S" is for savings account. "C" is for checking account. For the deposit or withdraw, it indicates which account is affected. For a transfer it indicates the account from which the money is taken; the money is automatically transferred to the other account.

Use global CONSTANT for **penalty (PENALTY)**. Don't hard code the number in your code.

You may need to add more member variables and functions to the classes than those listed above, and you need to implement these member functions.

You must store class Account and class Bank in a .h file. You must use safe guard (#ifndef) in the .h file. Implementation of the class member functions must be in a separate cpp file.

You need to use Makefile to compile the programs. Below is a sample of Makefile (save the code below to Makefile in the folder for this lab. You can use make to compile the program).

```
# begin of Makefile
#
CC=g++
#
CFLAGS = -c -Wall -I/usr/include/mysql
#LFLAGS = -L/usr/lib/mysql -lmysqlclient
LFLAGS =

all: bankmain

bankmain: bankmain.o account.o  printMeFirst.o
   $(CC) bankmain.o account.o  printMeFirst.o -o bankmain $(LFLAGS)

bankmain.o: bankmain.cpp
   $(CC) $(CFLAGS) bankmain.cpp

account.o: account.cpp account.h
   $(CC) $(CFLAGS) account.cpp

printMeFirst.o: printMeFirst.cpp
   $(CC) $(CFLAGS) printMeFirst.cpp
```

```
clean:
    rm *.o bankmain

run:
    ./bankmain
#end of Makefile
```

Use the c++ main program file below to run the program.

/* include all headers files */

```
int main()
{

   printMeFirst("Ron Sha", "CS-116 - 2020 Spring");  // you must call this function 1st

   Bank myBank;
   cout << "\nInital bank balances: \n";
   myBank.printBalances();   /* set up empty accounts */

   cout << "\nAdding some money to accounts: \n";

   cout << "\nAdding $1000 to saving \n";
   cout << "Adding $2000 to checking \n";
   myBank.deposit(1000, "S");  /* deposit $1000 to savings */
   myBank.deposit(2000, "C");  /* deposit $2000 to checking */
   myBank.printBalances();

   cout << "\nTaking out $1500 from checking,and moving $200 from";
   cout << " savings to checking.\n";

   myBank.withdraw(1500, "C"); /* withdraw $1500 from checking */
   myBank.transfer(200, "S");  /* transfer $200 from savings */
   myBank.printBalances();


   cout << "\ntrying to withdraw $900 from Savings.\n";
   myBank.withdraw(900,"S");
   myBank.printBalances();

   cout << "\ntrying to withdraw $400 from Checking.\n";
   myBank.withdraw(400,"C");
   myBank.printBalances();
```

```
    return 0;
}
```

Your program submission (must include 2 files):

1. **Include zip file contains all the files you used for this program. You should create a folder for each lab, and then just zip (compress) the folder directory. Please NOTE that you must separate all class definition (.h), all class implementation (c++ code for the class definition) and other programs into a separate .h and .cpp files.**

2. **Single pdf file contains all of the following section:**
    a. **Program description**
        i. Describe the purpose of this program
        ii. Some high-level algorithm (logic) of your program
    b. Source Code
        i. Include the source codes for each of your cpp file (put cpp file name before the cpp source code – screenshots of the source codes is ok)
        ii. Your source code for each function must be properly documented with the following information: (Use PrintMeFirst function as example)
            - You can use block comments for section of codes and/or line comments
            - Each function, at minimum, must have function header documentation which includes the following:
                - Description:
                - @Parameters
                - @Return

    c. Screen Shots
        i. Include the screen shots of the program output when you test the program. You must use the test cases I specified in the assignment.
        ii. Your program must print out the information from PrintMeFirst function.

Your program output should be similar to the one below:

```
Program written by: Ron Sha
Course Info: CS-116 - 2017 Spring
Date: Mon Feb 13 17:38:36 2017


Inital bank balances:
Savings account balance:  $    0.00
Checking account balance: $    0.00

Adding some money to accounts:

Adding $1000 to saving
Adding $2000 to checking
Savings account balance:  $ 1000.00
Checking account balance: $ 2000.00

Taking out $1500 from checking,and moving $200 from savings to checking.
Savings account balance:  $  800.00
Checking account balance: $  700.00

trying to withdraw $900 from Savings.
Only 800.00 is available.  But tring to withdrawn 900.00.  Deduct 5 from account
Savings account balance:  $  795.00
Checking account balance: $  700.00

trying to withdraw $400 from Checking.
Savings account balance:  $  795.00
Checking account balance: $  300.00
cs:lab2-bank$
```