**CSCI 1411 – Lab 13–Vectors of Objects**

**Goals:**

* Understand how to build and use Classes
* Understand how to use vectors of objects built with classes

**Development Environment:** (all students must use Clion)

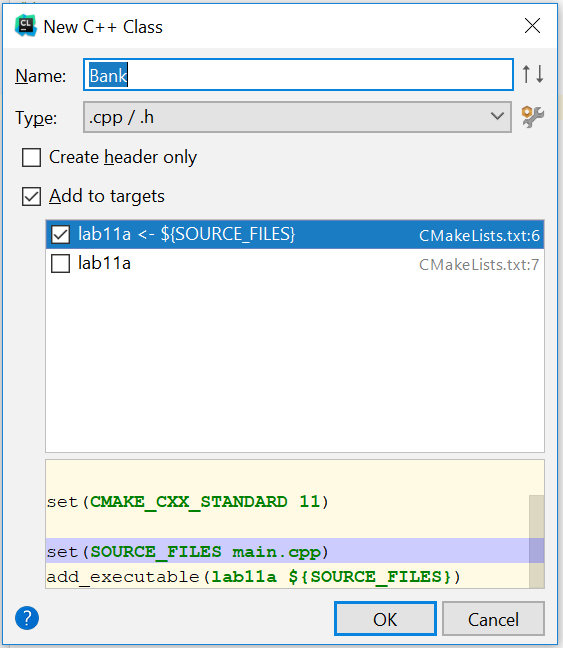
**Skills:** Classes, Default Constructor, Constructor, member functions, vectors of objects.

**Reading**: Chap 13

**Deliverables:** 1) This lab with 2 screen shots 2) lastnameFirstLab12.cpp, Scores.h, Scores.cpp

**Part I – Skills Practice (10 points)**

* Open a new project in CLion. Call the project Lab13a.
* For this project will make vectors of objects and use the built in member functions to add, and erase records.
* New->New C++ Class



* Add the name Bank. And accept the defaults Add to targets and Lab11a<-$(Source\_Files)
* This will create both a Bank.h file and a Bank.cpp file
* Double Click on the Bank.h file. Replace the entire text with this

**#ifndef BANK\_H  
#define BANK\_H  
  
#include <string>  
#include <iostream>  
using namespace std;  
class Bank  
{  
private:  
 string name;  
 int number;  
 float balance;  
public:  
 Bank();  
 Bank(string \_name, int \_number, float \_balance);  
 string getName(){return name;}  
 void setName(string \_name){name=\_name;}  
 int getNumber() {return number;}  
 void setNumber(int \_number){number = \_number;}  
 float getBalance(){return balance;}  
 void increaseBalance(float amount);  
  
};  
#endif**

* Now double click on the Bank.cpp. Replace the entire code with the code below:

**#include "Bank.h"  
Bank::Bank()  
{  
 name = "";  
 number = 0;  
 balance = 0;  
}  
  
Bank::Bank(string \_name, int \_number, float \_balance)  
{  
 name = \_name;  
 number = \_number;  
 balance = \_balance;  
}  
  
void Bank::increaseBalance(float amount)  
{  
 balance = balance + amount;  
}**

**New->New C++ class. Call it functions**

* **In functions.h**

**#ifndef FUNCTIONS\_H  
#define FUNCTIONS\_H  
#include "Bank.h"  
#include <vector>  
void bubbleSort(vector<Bank> & acctVector);  
ostream& operator << (ostream &out, Bank &tempBank);  
#endif**

* **In functions.cpp**
* Now double click on the main.cpp. Replace the entire code with the code below:

**#include "functions.h"  
void bubbleSort(vector<Bank> & acctVector)  
{  
 *//Bubble Sort records by name* int maxElement;  
 int index;  
 Bank tempBank3;  
  
 for (maxElement = acctVector.size() - 1; maxElement > 0; maxElement--)  
 {  
 for (index = 0; index < maxElement; index++)  
 {  
 if (acctVector[index].getName() > acctVector[index + 1].getName())  
 {  
*//swap the entire record (name, number, balance)* tempBank3 = acctVector[index];  
 acctVector[index] = acctVector[index + 1];  
 acctVector[index + 1] = tempBank3;  
 }*//if* }*//for* }*//for*}  
  
ostream& operator << (ostream &out, Bank &tempBank)  
{  
 cout << tempBank.getName() << endl;  
 cout << tempBank.getNumber()<<endl;  
 cout << tempBank.getBalance()<<endl<<endl;  
 return out;  
}**

* In main.cpp

**#include <iostream>  
#include <vector>  
#include "Bank.h"  
#include "functions.h"  
using namespace std;  
int main()  
{  
 string tempName;  
 int tempNumber, size;  
 float tempBalance;  
 Bank tempBank;  
  
 vector<Bank> acctVector;  
 cout << "How many records do you want to add?\n";  
 cin >> size;  
 cin.ignore();  
 for (int i=0; i<size; i++)  
 {  
 cout << "Name for customer " << i+1 <<endl;  
 getline(cin,tempName);  
 cout << "Number for customer " << i+1 <<endl;  
 cin >> tempNumber;  
 cout <<"Balance for customer " << i+1 <<endl;  
 cin >> tempBalance;  
 cin.ignore(); *//removes '/n' in buffer for next getline* cout <<endl;  
 tempBank.setName(tempName);  
 tempBank.setNumber(tempNumber);  
 tempBank.increaseBalance(tempBalance);  
 acctVector.push\_back(tempBank);  
 }  
  
 bubbleSort(acctVector);  
  
 for (int i=0;i<acctVector.size(); i++)  
 {  
 cout << acctVector[i]<<endl;  
 }  
  
return 0;**

* Now if you did everything correctly it should compile. If it has errors, try to fix the errors by reading the error and clicking on the link to the error.
* Once it successfully compiles, go to Run->Run Lab 13a (Note the Clion terminal window repeats all input from the user, so don’t worry about trying to fix that)
* Take a **screenshot** of the successful output below: