1 Instructions

When you are finished submit all your work through the MyClasses page for this class. Create a directory called Lab05, put each programming exercise into its own subdirectory of this directory, zip the entire Lab05 directory up into the file Lab05.zip, and then submit this zip file to Lab #5.

Make sure that you:

- Follow the coding and documentation standards for the course as published in the MyClasses page for the class.
- Check the contents of the zip file before uploading it. Make sure all the files are included.
- Make sure that the file was submitted correctly to MyCLasses.

All class structures are to have their own guarded specification file (.h) and implementation file (.cpp) that has the same name as the class. No inline coding in the .h files. In addition, you must create a make file that compiles and links the project on a Linux computer with a Debian or Debian branch flavor.

2 Programming Exercise

Design a class named `PersonData` with the following member variables:

- lastName
- firstName
- address
- city
- state
- zip
- phone

Write the appropriate accessor and mutator functions for these member variables. Next, design a class named `CustomerData`, which is derived from the `PersonData` class. The `CustomerData` class should have the following member variables:

- customerNumber
- mailingList

The customerNumber variable will be used to hold a unique integer for each customer. The mailingList variable should be a bool. It will be set to true if the customer wishes to be on a mailing list, or false if the customer does not wish to be on a mailing list. Write appropriate accessor and mutator functions for these member variables.

In particular, both classes should have the functions `PrintRecord` and `getRecord`. Neither should take any parameters. The `PrintRecord` will print the data to the console in a nice format, example is below. The `getRecord` function will return a string of the data with each field separated by a “/”, example below. Also, these two functions must be set up for dynamic binding. The CustomerData class should call the PrintRecord and getRecord functions from the parent class to do its portion of the printing or string creation, you do not want to rewrite the code you already wrote for the base class.

Create the following main and run it with your classes. The output is below the main code.

```cpp
#include <iostream>

#include "PersonData.h"
#include "CustomerData.h"

using namespace std;

int main() {
    PersonData P1;
    P1.setName("Don", "Spickler");
    P1.setAddress("1101 Camden Ave.", "Salisbury", "MD", "21801");
    P1.setPhone("410-543-6148");

    P1.PrintRecord();
    cout << endl;
    cout << P1.getRecord() << endl;
    cout << endl;

    PersonData *P2 = new PersonData();
    P2->setName("John", "Doe");
    P2->setAddress("Nowhere Ave.", "Podunk", "NB", "66666");
    P2->setPhone("123-456-7890");

    P2->PrintRecord();
    cout << endl;
    cout << P2->getRecord() << endl;
    cout << endl;

    CustomerData *P3 = new CustomerData();
    P3->setName("Jane", "Doe");
    P3->setAddress("Nowhere Ct.", "Podunk", "NB", "66666");
    P3->setPhone("987-798-1111");
    P3->setCustomerNumber(12345);
    P3->setMailingList(false);

    P3->PrintRecord();
    cout << "\n" << P3->getRecord() << "\n\n";

    PersonData *P4;
    P4 = P3;
    P3->PrintRecord();
    cout << endl;
    P4 = P2;
    P4->PrintRecord();

    delete P2;
    delete P3;
    P2 = nullptr;
    P3 = nullptr;

    return 0;
}
```

Output:

```
Spickler, Don
1101 Camden Ave.
Salisbury, MD 21801
410-543-6148

Don Spickler / 1101 Camden Ave. / Salisbury, MD 21801 / 410-543-6148

Doe, John
Nowhere Ave.
Podunk, NB 66666
123-456-7890

John Doe / Nowhere Ave. / Podunk, NB 66666 / 123-456-7890

Doe, Jane
Nowhere Ct.
Podunk, NB 66666
987-798-1111
Customer Number: 12345
On the Mailing List: No

Jane Doe / Nowhere Ct. / Podunk, NB 66666 / 987-798-1111 / C#: 12345 / ML: No

Doe, Jane
Nowhere Ct.
Podunk, NB 66666
987-798-1111
Customer Number: 12345
On the Mailing List: No

Doe, John
Nowhere Ave.
Podunk, NB 66666
123-456-7890
```