**Lab 09 [50 Points]**

**Note 1:** Do you own work in lab. You are not allowed to do talking and sharing with class fellows, violation may lead to zero marks in the lab for **both students**.

**Note 2:** Do your own effort as much as possible, you may ask TA's, in case they are busy **wait and carry on to some other problem or to same problem**.

**Note 3:** Read note 2 again, and follow it carefully. The time you spent in finding error will improve your own understanding of the code, which is the ultimate objective. Read code again and again, try changes by commenting previous instructions so that you can reverse changes easily

**Note 4:** Some students are already punished due to violation of sharing through discussion or watching class fellow's code. In future, this punishment will be severe because I am giving warnings again & again.

**Note 5:** There is always partial marking of the tasks, therefore do your own effort as much as possible. For good marks in lab regularly attend lab, do practice tasks, home works & assignments.

**Task 1:** Design a class named **PersonData** with the following member variables: **[25]**

* first name (string)
* last name (string)
* email (string)
* house no (unsigned integer)
* block (a single character A-Z)
* city (string without spaces)

Write the appropriate getter and setter functions for these member variables. Also write stream operator for output. Show output in some reasonable manner.

Next, design a child class of PersonData named **CustomerData**. The CustomerData class should have the following member variables:

* customer number (Assign customer number using customer count)
* emails allowed or not (bool)

The customer number variable will be used to hold a unique positive integer for each customer. The emails allowed or not is a bool type variable. It will be set to true if the customer wishes to be on a mailing list, or false otherwise. Write appropriate getter and setter functions for these member variables. Also write stream operator for output, this operator should use similar operator in parent class.

Write appropriate main function to demonstrate all the member functions in both classes.

**Task 2:** Design a class named **Employee**. The class should keep the following information:

* Employee number (A unique positive number starting from 1)
* Hire date (First, write a class **Date** having day, month & year private members with setters [check valid values], getters. Also write ostream operator for Date classs)

Write two constructors (parameterized & non-parameterized) and the appropriate getter, setter functions for the class.

Next, write a child class of Employee named **ProductionWorker**. The ProductionWorker class should have member variables to hold the following information:

* Shift (a positive integer having value 1 or 2 only)
* Hourly pay rate (positive float value)

The workday is divided into two shifts: day and night. The shift variable will hold an integer value representing the shift that the employee works. The day shift is shift 1, and the night shift is shift 2. Write both parameterized & non-parameterized constructors and the appropriate getter and setter functions for the class. Demonstrate the classes by writing main function calling all the member functions in both classes.