CL-217
Object Oriented Programming
Lab # 12

Objectives:

- Composition
- Friend Functions
- Friend Classes

Note: Carefully read the following instructions (*Each instruction contains a weightage*)

1. There must be a block of comments at start of every question's code by students; the block should contain brief description about functionality of code.
2. Comment on every function about its functionality.
3. Use understandable name of variables.
4. Proper indentation of code is essential
5. Write a C++ statement(s) for each of the following task one after the other, in the same order.
6. Make a Microsoft Word file and paste all of your C++ code with all possible screenshots of every **task outputs in MS word and do not submit .cpp file with word file**.
7. First think about statement problems and then write/draw your logic on copy.
8. After copy pencil work, code the problem statement on MS Studio C++ compiler.
9. At the end when you done your tasks, attached C++ created files in MS word file and make your submission on Microsoft teams. (Make sure your submission is completed).
10. Please submit your file in this format 19F1234_L9.
11. Do not submit your assignment after deadline.
12. Do not copy code from any source otherwise you will be penalized with negative marks.

Problem 1: Friend Function

- Let us assume you are going to print event cards of an upcoming event in your university. *Eventcard* is a class and has the following **private** attributes:
- 1. Event Name
- 2. Date of event
- 3. Time of event
- 4. Venue (address of event)
- Use friend function to print all three attributes on console.

- The print function will be a function that is not a part of class itself but a friend of event card class.

Problem 2: Friend Classes

- Make a class ***myclass*** and declare private member *secret* of type int.
- Initiliaze it with 0
- It has a member function *print* which prints the value of *secret*.
- Make ***secondclass*** its friend.
- Create class ***secondclass*** which has a method change and it changes the value of secret to any user desired value. (You may take input of new value or set it by yourself, both of these should be done from main).
- Now display the changed value by using *print* function of ***myclass***.

Problem 3:

Write a class Person, having following private data members:

1. Name
2. Data of birth (a constant data member)
3. Count (a static data member)
4. CNIC (a constant data member)

Count should keep track of how many person objects are created. Set the value of count before any object is created. Display it at the end of the program.

Public member functions:

1. Constant member function to access the Date of Birth of each person (get DoB)
2. Constant member function to access the CNIC of each person (get CNIC)
3. Display function for Person record output (const)

Display a person record Mr. X with DoB 1st January 2001.

Problem 4: Friend Functions and Constant data members

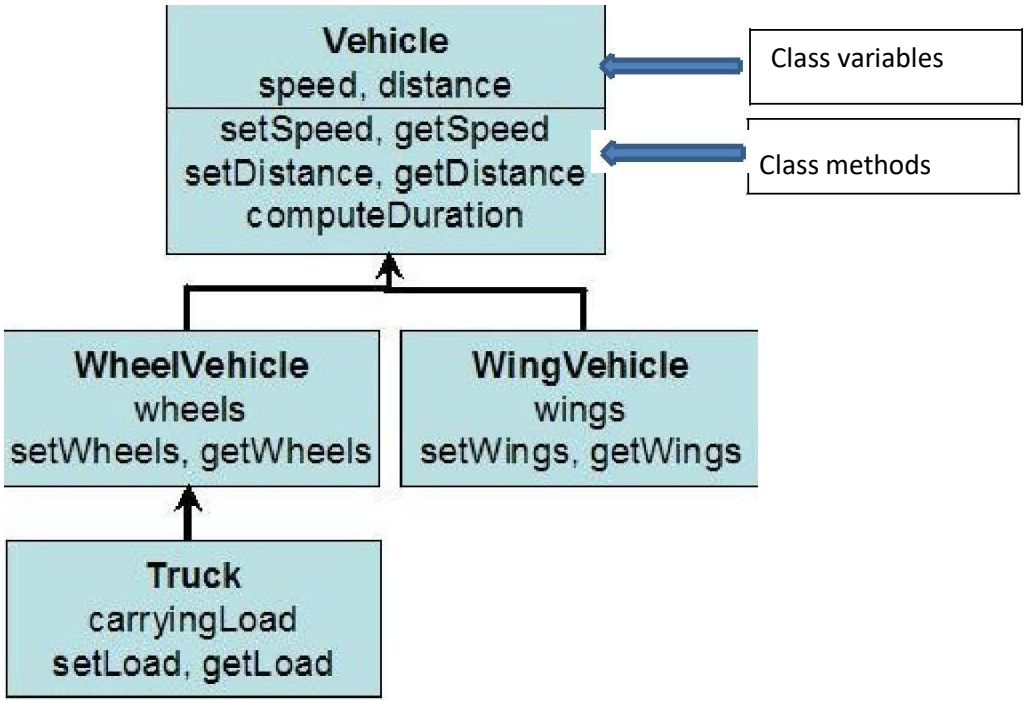
Modify the 2nd problem as follows:
Add Input function to the class.
Make all member functions private.

Task:

Write friend functions to access the member functions of the Person class separately (one friend function for getting DoB, one for getting CNIC and one for displaying Person's record).

Problem 5: (Classes, Inheritance, Friend functions)

Design the following classes with given variables and methods using inheritance.



- **All four classes data members will be protected.**
- **Inheritance type is protected.**
- You have to write default constructors for each class. Child's class constructor should call parent class constructor.
- You have to write overloaded constructors for each class. Child's class overloaded constructor should call parent class overloaded constructor.
- You have to write **friend input** member function for each **leaf classes**. This function will input all the data member values by accessing them. i.e. friend function for Truck class will be InputTruck(Truck T).
- You have to write **friend print** member function for each **leaf classes** to print the data members to console. This function will print all the data member values by accessing them.
- **Create objects of each leaf class and demonstrate your work with proper output.**

Problem 6: Composition

Create a class Time with following data members and member functions

```
public:
	Time();
	Time(int, int);
	void setTime(int, int);
	void getTime(int&, int&);
	void printTime();
	void incrementHours();
	void incrementMinutes();
private:
	int hr;
	int min;
```

Create a class Date with following data members and member functions

```
public:
	Date();
	Date(int, int, int);
	void setDate(int, int, int);
	void getDate(int&, int&, int&);
	void printDate();

private:
	int month;
	int day;
	int year;
```

Create a class Event with following data members and member functions

```
public:
	Event(int hours = 0, int minutes = 0, int m = 1,
		int d = 1, int y = 1900, string name = "Christmas");
	void setEventData(int hours, int minutes, int m, int d, int y, string
name);
	void printEventData();

private:
	string eventName;
	Time eventTime;
	Date eventDay;
```

Write a Main and create events using the above classes to demonstrate composition.