1. Class Design

Create a Class named Vehicle.

a. It must have three member variables.

b. Class must have three member functions.

I. One function would take value from the keyboard (Function name: getValue) II. Another One's value would be assigned by a programmer (Function name: setValue) III. Third function would print the output (Function name: printValue)

Main Function

Create two objects of your created class. One object would take value from the keyboard and another one's value would be assigned. Print the values of two objects.

2. As before, Create a University class with 5 member variables and 2 member functions(getData() and displayData()). Then make an array of 3 University objects and use two functions to get information and show information.

3. Create a class called Reverse. This class must include one integer variable, one getInput () function, and one doReverse() function. Your task is to take an integer input from the keyboard and reverse it.

4. A phone number, such as (212) 767-8900, can be thought of as having three parts: the area code (212), the exchange (767), and the number (8900). Write a program that uses a class (Phone) to store these three parts of a phone number separately. Create two objects of type phone. Initialize one, and have the user input a number for the other one. Then display both numbers. The interchange might look like this: Enter your area code, exchange, and number: 415 555 1212 My number is (212) 767-8900 Your number is (415) 555-1212

5. Create a class called time. Its three members, all type int, should be called hours, minutes, and seconds. Write a program that prompts the user to enter a time value in hours, minutes, and seconds using the getTime() function. The program should print out the total number of seconds using DisplayTimeInSeconds() function.

6. Find out which student got the highest mark among the 5 students. For this, you have to create a Student class. In this class,

- Pass an array of objects as an argument in a function.
- Create necessary constructors and functions.

7. Create a class named **time** that must have three integer data members (hours, minutes, and seconds). Create two constructors, one member function named add_time (return object), and one display function to print the time in 11:59:59 format. The add_time function must take two

objects
as arguments. The main function calls the add_time function to add two-time objects and store the result in a third object. Use the display function to print the result on the console. **Input: 2 55 40, 5 20 30**
Output: 8:16:10

8. Create a class named **Complex** that must have two integer data members (real, and imag). Create two constructors, one Read function to take keyboard input, one Add (return object), function, and one Display function to print results. The Add function must take one object as an argument. The task of the main function is the same as question 2.
Input
Enter real and imaginary numbers respectively: 16 7
Enter real and imaginary numbers respectively: 5 8
Output:
Sum = 21 + 15i

9.

H = 11.3 cm

B = 8.7 cm

Create a class for the above **Triangle**.

a. Create one default constructor.

b. Create four parameterized constructors. They would look like this –

- Constructor_name(double, double)
- Constructor_name(double, int)
- Constructor_name(int, double)
- Constructor_name(int, int)

c. Create one copy constructor.

d. Create a function that would **return** the area of the triangle.

e. Create a destructor.

Call each of the functions from the main functions. Call **Constructor_name(double, int)** using pass-by-value and **Constructor_name(int, double)** by pass-by-reference.

10. K went to a shopping mall. He bought 6 items. At the cash counter, the cashier said if his total purchase was greater than or equal to 500 Taka, he would get a 20% discount. Otherwise, he would get a 2% discount on the total purchase. He gave the cashier a 1000 Taka note. How much change the cashier would return?

Solve this problem using **class**. You must use three functions that will return an **object**. Create the necessary constructors for your code.

N.B: You must set the price of each item greater than 50 Taka to avoid any complexity. The total purchase amount must not exceed 1000 Taka.