1. Create a **Person** class that includes fields for last name, first name, and zip code. If no arguments are supplied, Person class includes a default constructor that initializes last name, first name, and zip code to “X”. Also Person class includes a display function. Write a main() program that instantiates and displays two Person objects: one that uses the default values, and one for which you supply your own values.

2. Create a class named **SavingsAccount**. Provide fields for the account number, balance, and interest rate. Provide two constructors. One sets each field to 0; the other takes an account number argument and sets the account’s opening balance to $100 at 3% interest. Include a function that displays an account’s data fields. Write a main() program that instantiates one **SavingsAccount** object of each typed and displays their values.

3. Create a class called **Complex** for performing arithmetic with complex numbers. Write a driver program to test your class.

   Complex numbers have the form
   \[ \text{realPart} + \text{imaginaryPart} \times i \]

   where \( i = \sqrt{-1} \)

   Use floating-point variables to represent the **private** data of the class. Provide a constructor function that enables an object of this class to be initialized when it is declared. The constructor should contain default values in case no initial values are provided. Provide public member functions for each of the following:

   A. Addition of two **Complex** numbers: The real parts are added and the imaginary parts are added, respectively.
   B. Subtraction of two **Complex** numbers: The real part of the right operand is subtracted from the real part of the left operand and the imaginary part of the right operand is subtracted from the imaginary part of the left operand.
   C. Printing **Complex** numbers in the form \((a, b)\) where \(a\) is the real part and \(b\) is the imaginary part.

4. Define and implement a class named **Cylinder** which has members to represent the radius and height of a cylindrical object. Provide application programs with access functions to:

   A. Assign a radius and height to an object
   B. Return the volume of an object of this class (**hrvolume** = \( \pi r^2 h \) where \( r = \text{radius}, h = \text{height} \))
   C. Return the surface area of an object of this class (**hhrarea** = \( 2\pi r^2 + 2\pi rh \) where \( r = \text{radius}, h = \text{height} \))

   Write an application program to demonstrate the class.

5. Create a class **Rectangle**. The class has **height** and **width**, each of which defaults to 1. It has member functions that calculate the perimeter and the area of the rectangle. It has **set** and **get** functions for both height and width. The **set** functions should verify that height and width are each floating-point numbers larger than 0.0 and less than 20.0.

6. Create a more sophisticated **Rectangle** class than the one you create in assignment 4-5. This class stores only the Cartesian coordinates of the four corners of the rectangle. The constructor calls a set function that accepts four sets of coordinates and verifies that each of these is in the first quadrant with no single x or y coordinate larger than 20.0. The set function also verifies that the supplied coordinates do, in fact, specify a rectangle. Member functions calculate the height, width, perimeter, and area. The height is the larger of the two dimensions. Include a predicate function square that determines if the rectangle is a square.

7. An International Standard Book Number (ISBN) is a code of 10 characters separated by dashes (-ashes) such as 0-670-87162-4. An ISBN consists of four parts: a group code, a publisher code, a code that uniquely identifies the book among the publisher’s offerings, and a check character. For the ISBN 0-670-82162-4, the group code is 0, which identifies the book as one from an English-speaking country. The publisher code 670 identifies the book as a Viking Press publication. The code 82162 uniquely identifies the book among the Viking Press publications. The check character is computed as follows:

   A. Compute the sum of the first digit plus two times the second digit plus three times the third digit… plus nine times the ninth digit.
   B. Compute the remainder of this sum divided by 11. If the remainder is 10, the last character is X. Otherwise, the last character is the remainder.

   For example, the sum for ISBN 0-670-82162-4 is
   
   \[ 0 + 2x6 + 3x7 + 4x0 + 5x8 + 6x2 + 7x1 + 8x6 + 9x2 = 158 \]

   The remainder when 158 is divided by 11 is 4, the last character in the ISBN. Implement a class to represent an ISBN. The class should have methods to **set** and **get** the ISBN as a string and to **check** whether the ISBN is valid.

8. Implement a **Book** class that represents pertinent information about a book, including the book’s title, author, publisher, city, date of publication, and price. The class should include the data member
9. Consider the following structure declaration :
   ```
   struct customer {
      char fullname[35];
      double payment;
   }
   ```
   Write a program that adds and removes `customer` structures from an array, represented by a class declaration. Each time a customer is removed, his payment is added to a running total and the running total is reported.

10. Implement an interactive `Calculator` class that accepts as input an arithmetic expression such as “25 / 5 + 4” and then evaluates the expression, printing the value. In this example, the output would be 9. There should be methods to validate the input expression. For example, if the user inputs “25 / 5 +” then the output should be an error message such as “ERROR: operator-operand imbalance”.

11. Create a class, such as apartment number and monthly rent, to hold data for apartments you rent. Create a second class for tenants. Include data members such as name, phone number, and apartment. Include any appropriate functions you think these classes should contain. Create a member function for the tenant class that can access and display the tenant’s apartment number, name, and monthly rent amount. Make the Tenant class a friend of the Apartment class. Write a demonstration program for your classes.