Assignment-1 (Object Oriented Paradigm)

P1: Write a function using reference variables as arguments to swap the values of a pair of integers.

P2: Write a program to print the following outputs using for loops
1
22
333
4444
55555

P3: Write a C++ program to input a number and check and print whether it is a 'Pronic' number or not. Use a function int Pronic(int n) to accept a number. The function returns 1, if the number is 'Pronic', otherwise returns zero (0). (Hint: Pronic number is the n1umber which is the product of two consecutive integers)
Examples:
12 = 3 * 4
20 = 4 * 5
42 = 6 * 7

P4: Write a program in C++ to find area and perimeter of circle by creating objects.

P5: Write a program in C++ to find areas of triangle, rectangle and square by creating objects.

Assignment-2 (Class, Object and Function Overloading)

P6: Write a program in C++ to add two complex numbers by creating objects.

P7: Write a program in C++ to implement class ADD that will have 3 overloaded functions. 1st one will add two integer numbers. 2nd one will add two floating and one integer number. 3rd one will take two complex objects as parameter and will add two complex numbers.

P8: Write a C++ program that will ask for a temperature in Fahrenheit and display it in Celsius using a class called temp and member functions.

A1: Write a simple program that converts the temperature in degree Celsius to degree Fahrenheit and vice versa using the basic concept of class and object. Make separate classes for Centigrade and Fahrenheit which will have the private member to hold the temperature value and make conversion functions in each class for conversion from one to other. For example, you will have a function to Fahrenheit() in class Celsius that converts to Fahrenheit scale and returns the value.

A2: Write a program in C++ to implement DATE class which will have 3 data members (day, month, and year) and some member functions. Use a function to get date. Create another function to validate a given date. (e.g., 30.02.2013 is not a valid date)

A3: Assume that object represents an employee report that contains the information about employee id, total bonus, and total overtime in a particular year. Use four objects to represent four employees reports. Write a program that displays report information. Use setpara() overloaded member functions to set report attributes by passing/without passing the arguments and member function displayreport() to show the reports according to parameter passe[1]d.

Assignment-3 (Constructor-Destructor and Static Member)

P9: Implement stack where each stack object deals with different array sizes as per the user's given size.

P10: Create a complex class to perform addition, subtraction, multiplication and division of two Complex Numbers. Your complex class should contain:
i. Constructors to initialize data members
ii. Copy constructor
iii. Member functions for addition, subtraction, multiplication and division of two Complex Numbers

P11: Create a class called Time that has a separate integer member data for hours, minutes, and seconds. One method initializes it to fixed values. A member function should display it in HH:MM:SS format. The final two member functions should add and subtract two objects of time passed as arguments. A main() program should create two initialized time objects. It should then add the two initialized objects, storing the result in a third time object. Finally, it should display the value of the third time object.

P12: Write down a program on C++ to define your own Stringclass. Your String class should contain:
i. Dynamic constructor(s) to allocate memory space for a string.
ii. Copy constructor
iii. Member function to merge two strings and store it in another string.
iv. Destructor

A4: Write down a program on C++ to define a class Matrix which uses a 2D array and two variables rlimit and climit to hold the row size and column size of the matrix. Your Matrix class should contain:
i. Copy constructor
ii. Dynamic constructor to allocate memory space for the matrix.
iii. Member function for addition of two matrices and store it in another matrix.
iv. Destructor

A5: Write an object-oriented program to enter and display Employee information. Enter the following information about students:

- Name
- Age
- Department
- Salary

Use a constructor to allocate memory for n employees (show both static and dynamic initialization of objects). Also define a destructor to de-allocate memory.

Assignment-4 (Operator overloading, Friend Class and Function)

P13: Create a class String then implement the following operations:
i. Overload + operator to concatenate two strings.
ii. Overload assignment operator = to assign one string into another.
iii. Overload comparison operators <, == to compare two strings.

P14: Create a complex class to perform addition, subtraction, multiplication and division of two Complex Numbers. Your class should contain:
i. Constructors to initialize data members
ii. Copy constructor
iii. Overload operators +, -, *, / for addition, subtraction, multiplication and division of two ComplexNumbers respectively. Then find the expression a - b * c + d (where a, b, c, and d are complex objects).

A7: Create a class called HEIGHT that stores the height of a student in feet and inches in two private instance variables. Include a constructor that sets these values. Define a function into_cm(), which returns the height in cm. Overload the operator - to perform the difference of two students' heights. Also overload comparison operators <, == to compare two students' heights.

A8: Create a class complex that contains two double data members. Overload +, -, and * arithmetic operators using a friend function, so that they can operate on the object of complex. Then find the expression a - b * c + d (where a, b, c, and d are complex objects).

A9: Create two classes DM & DB which store the value of distances. DM stores it in meters & centimetres. DB stores it in feet & inches. Write a program that can read values for the class objects & add one object of DM with another object of DB. Use a friend function to carry out the addition operation. The objects that store the result may be of either type depending on the units in which the results are required. The display function should act accordingly.

Assignment-5 (Inheritance)

P15: Write three derived classes inheriting functionality of base class person (should have member functions). That ask to enter name and age and with added unique features of student, employee, and functionality to assign, change and delete records of student and employee. And make one member function for printing the address of the objects of classes (base and derived) using the this pointer. Create two objects of the base class and derived classes each and print the addresses of individual objects. Using a calculator, calculate the address space occupied by each object and verify this with address spaces printed by the program.

P16: Write a base class that asks the user to enter a complex number and a derived class adds its own complex number with the base. Finally, make a third class that is a friend of the derived and calculates the difference of the base complex number and its own complex number.

A10: This is a database of the employees of an Educational Institute. Specify all the classes & define functions to create a database & retrieve individual information when required.

Assignment-6 (Polymorphism (Virtual Functions))

P17: Write a program to create a class shape with functions to find the area of the shapes and display the name of the shape and other essential components of the class. Create derived classes circle, rectangle, and trapezoid each having overridden functions area and display. Write a suitable program to illustrate virtual functions and a virtual destructor.

A10: Create a class Person and two derived classes Employee and Student, inherited from class Person. Now create a class Manager which is derived from two base classes Employee and Student. Show the use of the virtual base class.

A11: Create a base class called shape. Use this class to store two double-type values that could be used to compute the area of figures. Derive two specific classes called triangle and rectangle from the base shape. Add to the base class, a member function get_data() to initialize base class data members and another member function display_area() to compute and display the area of figures. Make display_area() as a **virtual function** and redefine this function in the derived classes to suit their requirements. Using these three classes, design a program that will accept dimensions of a triangle or a rectangle interactively, and display the area. Remember the two values given as input will be treated as lengths of two sides in the case of rectangles and as base and height in the case of triangles, and used as follows:

- Area of rectangle = x * y
- Area of triangle = 1/2 * x * y