SCS 1209 - Object Oriented Programming

Practical 09

1. Create a class named Shape with a function that prints "This is a shape". Create another class named Polygon inheriting the Shape class with the same function that prints “Polygon is a shape". Create two other classes named Rectangle and Triangle having the same function which prints "Rectangle is a polygon" and "Triangle is a polygon “respectively. Again, make another class named Square having the same function which prints "Square is a rectangle”. Now, try calling the function by the object of each of these classes.

2. We want to calculate the total marks of each student of a class in Database, Statistics, and Data structures and the average marks of the class. The user enters the number of students in the class. Create a class named Marks with data members for roll number, name, and marks. Create three other classes inheriting the Marks class, namely Database, Statistics, and Data structures, which are used to define marks in individual subjects of each student. The roll number of each student will be generated automatically.

3. Check the output of the code. Indicate the reason if there is an error.

```
 #include<iostream>
using namespace std;
class base
{
public:
void fun_1() { cout << "base-1\n"; }
virtual void fun_2() { cout << "base-2\n"; }
virtual void fun_3() { cout << "base-3\n"; }
virtual void fun_4() { cout << "base-4\n"; }
};
class derived : public base
{
public:
void fun_1() { cout << "derived-1\n"; }
void fun_2() { cout << "derived-2\n"; }
void fun_4(int x) { cout << "derived-4\n"; }
};
```

```
int main()
{
base *p;
derived obj1;
p = &obj1;
// Early binding because fun1() is non-virtual
// in base
p->fun_1();
// Late binding (RTP)
p->fun_2();
// Late binding (RTP)
p->fun_3();
// Late binding (RTP)
p->fun_4();
}
```

4. Make a class named sweet with a data member to calculate the number of sweets in a basket. Create two other classes named Chocolate and Ice cream to calculate the number of Chocolate and Ice cream in the basket. Print the number of sweets of each type and the total number of sweets in the basket. (Multiple inheritances)

5. Circle and Rectangle are two base classes from which the class Cylinder is being inherited. The data members of both the base classes are declared in protected mode.

   Thus, the class Cylinder can access the data member radius of class Circle and data member length, breadth of the class Rectangle, but the objects of the class Cylinder cannot access these protected data members.

   The volume of the cylinder is equal to 22/7*(radius*radius*length). Thus, instead of defining these data again, they can be inherited from the base classes Circle and Rectangle (radius from class Circle and length from class Rectangle).

   Write a C++ program find the Area and Volume using multiple inheritance.