OOP Practice Problems

1) Create a base class Shape. Each Shape has a 2D location (Point). There are three inherited classes from Shape: Circle, Square and Triangle. Each Circle has a radius member variable. Provide appropriate constructors, accessor and mutator functions for it. Square has a size member variable that represents its size. Provide appropriate constructors, accessor and mutator functions for it. Triangle has two member variables: base and height. Provide appropriate constructors, accessor and mutator functions.

   There are two pure virtual functions in the Shape class. The first function intersect is declared as follows:

   ```
   virtual bool intersects(const Point& point)=0;
   ```

   This function returns true if the given point intersects a given Shape. The second function getArea returns the area of the given Shape. It is declared as follows:

   ```
   virtual float getArea()=0;
   ```

   This function returns the area of the given shape. Note that for calculating the area of triangle, you can use the formula ½ * base * height, for Circle the area can be calculated using $PI*r^2$ and for square its size*size.

   Inherit three classes from the Shape class, Circle, Square, Triangle and implement the intersects and getArea functions in all of them. Provide appropriate default, two parameters, copy constructor and assignment operator in Shape class. For all other classes, provide an appropriate constructor depending on the additional member variables that are needed to be assigned. Overload the stream insertion operator to output the details about the Shape that is for parent class Shape, you should output Shape [x: {x pos}, y: {y pos}]

   For child classes only output additional attributes stored for each class for e.g. for Circle lets

   In main, create a vector of Shape objects dynamically and populate it with an object of each child class that is Circle, Square, Triangle. Then polymorphically call the intersect function using the Shape pointer.

2) Use public inheritance to define the hierarchy of classes A, B, and C, where the A class is the base. Each class should contain only an integer private member and the public functions get() and show().

   Define the classes in such a way that the following program works.

   ```
   int main() {
      C c;
      c.get(); /* The program should read sequentially three integers and store them in
                  the respective class members. */
      c.show(); /* The program should display sequentially the values of the three class
                   members. */
      return 0;
   }
   ```

   Suppose that the user enters the values 10, 20, and 30. The program should display:

   ```
   Class_A input: 10
   Class_B input: 20
   Class_C input: 30
   Class_C output: 30
   Class_B output: 20
   Class_C output: 10
   ```

3) Define the classes in such a way that the following program works.

```
class Circle {
   private:
      float rad; ...
};
class Sphere : public Circle {
   private:
      string col; ...
};
int main() {
  Sphere sph("Blue", 2); /* Make appropriate calls to constructors, so that Blue is
                            stored in the col member of Sphere and 2 in the rad
                            member of Circle. */
  Circle& c = sph;
  c.show();              /* The program should display the colour of the sph object
                            and its volume according to the type 4/3*pi*r3. In this
                            example, the program should display Blue and 33.49. */
  return 0;
}
```

4) Define the abstract class named Product, which contains the code and the price of the product as protected members. Define the Book class derived with public access from the Product with 716 Introduction to C++ the private member the publishing firm (of type string). Also, define the Car class derived with public access from the Product with private member the horsepower. Add appropriate functions in the classes so that the following program works.

```
int main() {
   Book b(1, 2, "pub"); /* code should become 1, the price 2, and the name
                           of the publisher pub. */
   Car c(4, 5, 6);   /* should become 4, the price 5, and the horsepower 6.*/
   Product& p1 = b;
   Product& p2 = c;
   p1.show();     /* should display all the data of the book, i.e., 1, 2, and pub. */
   p2.show();     /* should display all the data of the car, that is, 4, 5, and 6. */
  return 0;
}
```