Code 1 – Classes

Create three classes that should be related to each other using inheritance:

(1) class Product, which has three data elements - the name, price and weight of the product (the base class for all classes);

(2) class Buy, containing data of the quantity of purchased goods (units), the price for all purchased goods and of the weight of the goods (derived class from Product class and base class for Check class);

(3) class Check, not containing any data element. This class should display information about the product and purchase (derived class from Buy class);

To interact with class data, create set- and get-methods. All class data elements are declared private.

Draw the class hierarchy

Code 2 – Member access

Consider the following class declarations in C++:

```cpp
class C {
    protected: int x;
    public: void f(){...};
};

class C1: public C {
    protected: int x1;
    public: void h(C *obj){...};
};

class C2: public C1 {
    public: int x2;
};

class C3: public C {
    public: void f(){...};
};
```

(1) Draw the class hierarchy

(2) Assume that main contains the following declaration:

C1 obj1;

For each of the following expressions, say whether it is allowed in the code of main or not (they can be forbidden either because they violate the class definition or the protection mechanism)

obj1.x , obj1.f() , obj1.x1 , obj1.x2

(3) Assume that the body of C1::h contains the following declarations

```cpp
C2 *obj2;
C3 *obj3;
```

For each of the following expressions, say whether it is allowed in the body of `C1::h` or not

```
obj->x , obj2->x , obj3->x
```

(4) Assume that the body of `C1::h` contains the following statement

```
obj->f();
```

(5) Assume that we call `C1::h` with a parameter (pointing to an object) of class `C3`. What is the method f executed, `C::f()` or `C3::f()`? And what would be the method executed if `f` were declared virtual in C++?

………

Consider the following class declarations in Java:

```
class C {
    protected int x;
    public void f(){...};
}


class C1 extends C {
    protected int x1;
    public void h(C obj){...};
}


class C2 extends C1 {
    public int x2;
}


class C3 extends C {
    public void f(){...};
}
```

Draw the class hierarchy

(1)Assume that main contains the following declaration:

   C1 obj1;

(2) For each of the following expressions, say whether it is allowed in the code of main or not (they can be forbidden either because they violate the class definition or the protection mechanism)

   obj1.x , obj1.f() , obj1.x1 , obj1.x2

(3) Assume that the body of h(C obj) method from C1 class contains the following declarations

   C2 obj2;

   C3 obj3;

(4) For each of the following expressions, say whether it is allowed in the body h(C obj) method from C1 class or not

   obj.x , obj2.x , obj3.x

Assume that the body of h(C obj) method from C1 class contains the following statement

   obj.f();

Assume that we call h(C obj) method from C1 class with a parameter (pointing to an object) of class C3. What is the method f executed, f() from C or f() from C3? And what would be the method executed if f() were declared virtual in C++?