Algorithm Workbench 34 - 37

34. Write the first line of the declaration for a `Poodle` class. The class should be derived from the `Dog` class with public base class access.

```
class Poodle : public Dog {
```

35. Write the first line of the declaration for a `SoundSystem` class. Use multiple inheritance to base the class on the `CDPlayer` class, the `Tuner` class, and the `CassettePlayer` class. Use public base class access in all cases.

```
class SoundSystem : public CDPlayer, public Tuner, public
CassettePlayer {
```

36. Suppose a class named `Tiger` is derived from both the `Felis` class and the `Carnivore` class. Here is he first line of the `Tiger` class declaration:

```
  class Tiger : public Felis, public Carnivore
```

Here is the function header for the `Tiger` constructor:

```
  Tiger(int x, int y) : Carnivore(x), Felis(y)
```

Which base class constructor is called first, `Carnivore` or `Felis`?

`Felis` is called first (order of inheritance).

37. Write the declaration for class B. The class's members should be

- `m`, an integer. The variable should not be accessible to code outside the class or to member functions in any class derived from class B.
- n, an integer. This variable should not be accessible to code outside the class, but should be accessible to member functions in any class derived from class B.
- `setM`, `getM`, `setN`, `getN`. These are set and get functions for the member variables `m` and n. These functions should be accessible to code outside the class.
- `calc`, a public virtual member function that returns the value of `m` times n.

```
class B {
  int32_t m;
  int32_t n;
public:
  int32_t getM() { return m; };
  int32_t getN() { return n; };
  void setM(int32_t _m) { m = _m; };
  void setN(int32_t _n) { n = _n; };

  virtual int32_t calc() const { return m * n; }
};
```

Next write the declaration for class D, which is derived from class B. The class's members should be

- q, a float. This variable should not be accessible to code outside the class but should be accessible to member functions in any class derived form class D.
- r, a float. This variable should not be accessible to code outside the class, but should be accessible to member functions in any class derived from class D.
- setQ, getQ, setR, getR. These are the set and get functions for the member variables q and r. These functions should be accessible to code outside the class.
- calc, a public member function that overrides the base calc function. This function should return the value of q times r.

```
class D : public B {
  float q;
  float r;
public:
  float getQ() { return q; }
  float getR() { return r; }
  void setQ(float _q) { q = _q; }
  void setR(float _r) { r = _r; }

  int32_t calc() const { return q * r; }
};
```