CSE1121: Structured & OOP Language

Sumaya Kazary
Assistant Professor
Department of Computer Science and Engineering
DUET, Gazipur.

Acknowledgement
Thanks to the authors of all the books and online tutorials used in this slide.

What is OOP?

- OOP is a powerful way to approach the task of programming.
- OOP encourages developers to decompose a problem into its constituent parts.
- *Each component becomes a self-contained object that contains its own instructions and data that relate to that object.*
- So, complexity is reduced and the programmer can manage larger programs.

What is OOP? (cont.)

- All OOP languages, including C++, share three common defining traits:
  - Encapsulation
    - Binds together code and data
  - Polymorphism
    - Allows one interface, multiple methods
  - Inheritance
    - Provides hierarchical classification
    - Permits reuse of common code and data

C++ Program Structure

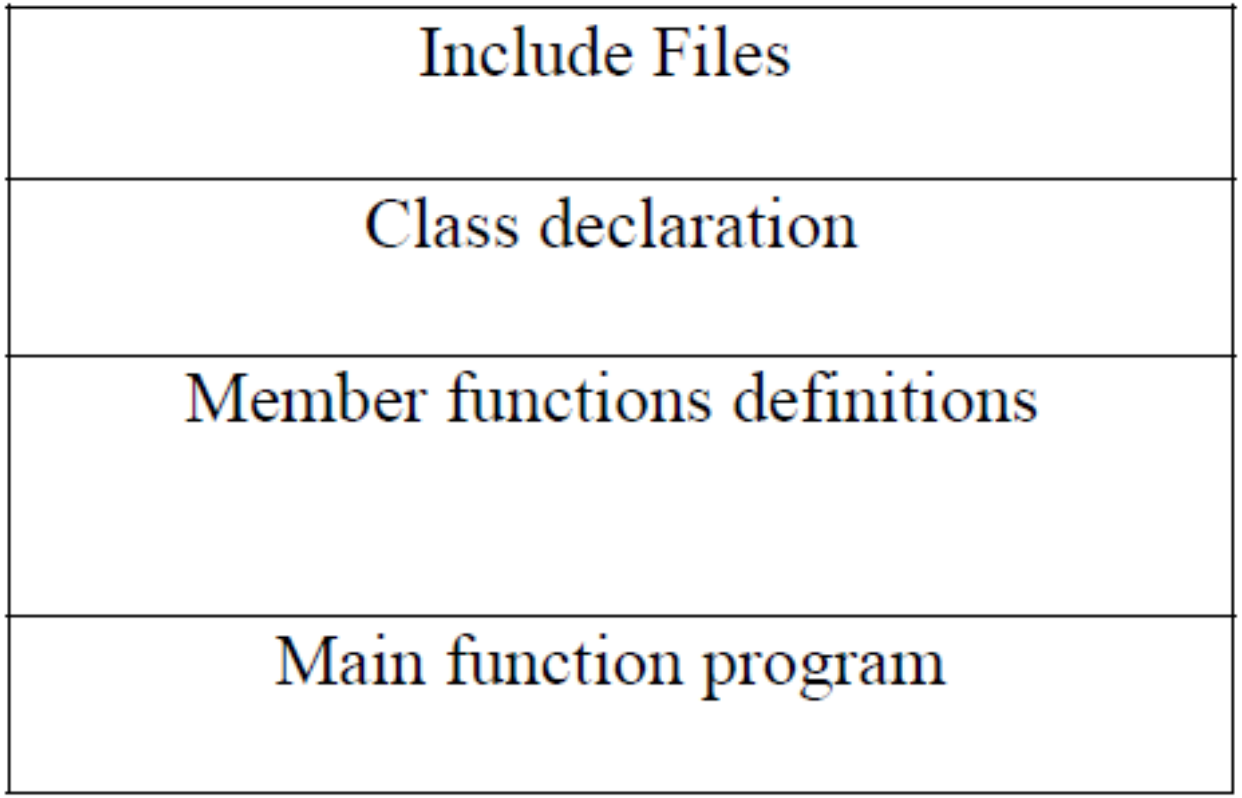


Fig : Structure of a C++ program

Classes: A First Look

- A class declaration is a logical abstraction that defines a new type.
- It determines what an object of that type will look like.
- An object declaration creates a physical entity of that type.
- That is, an object occupies memory space, but a type definition does not.
- **Example:** p-23.cpp, p-26.cpp, stack-test.c

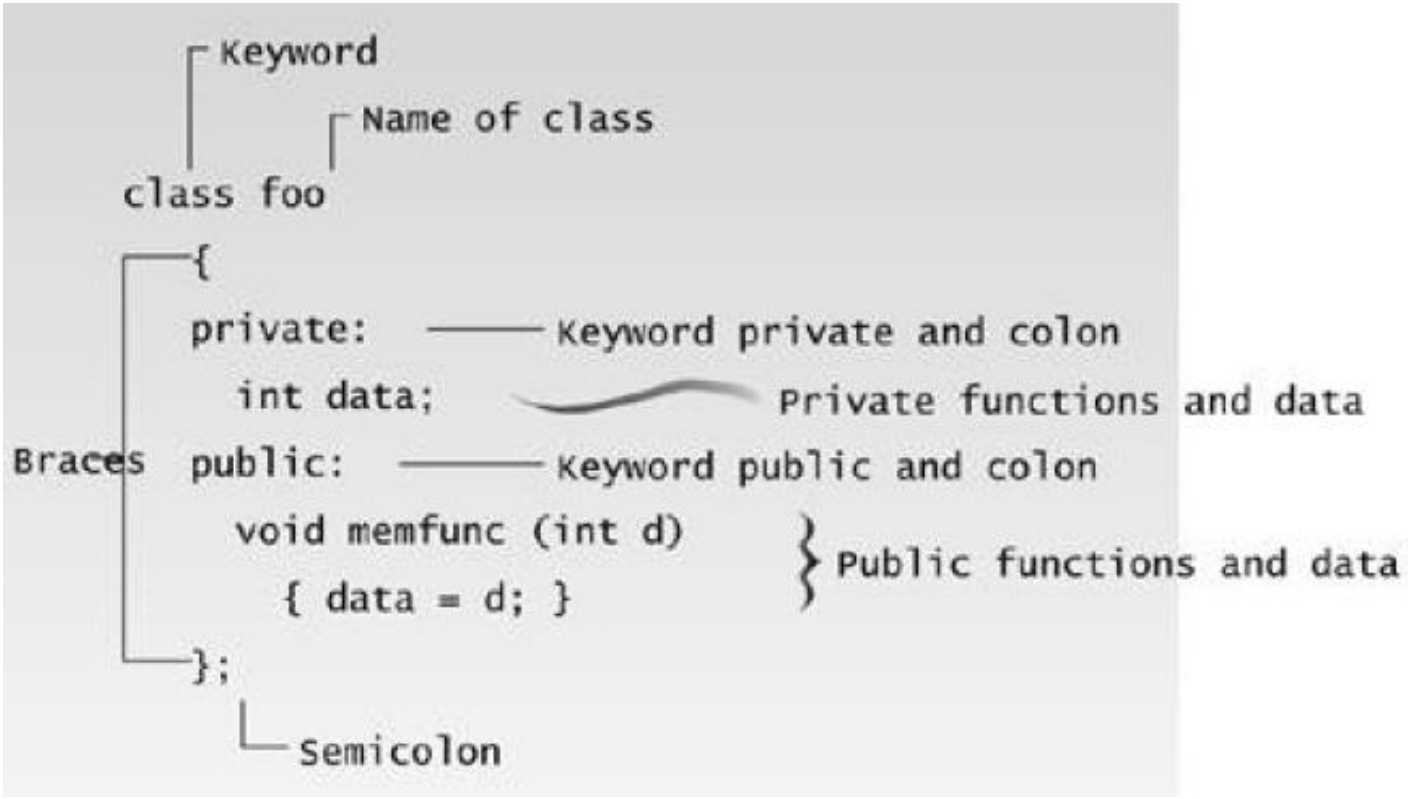
Classes: A First Look

- General syntax -

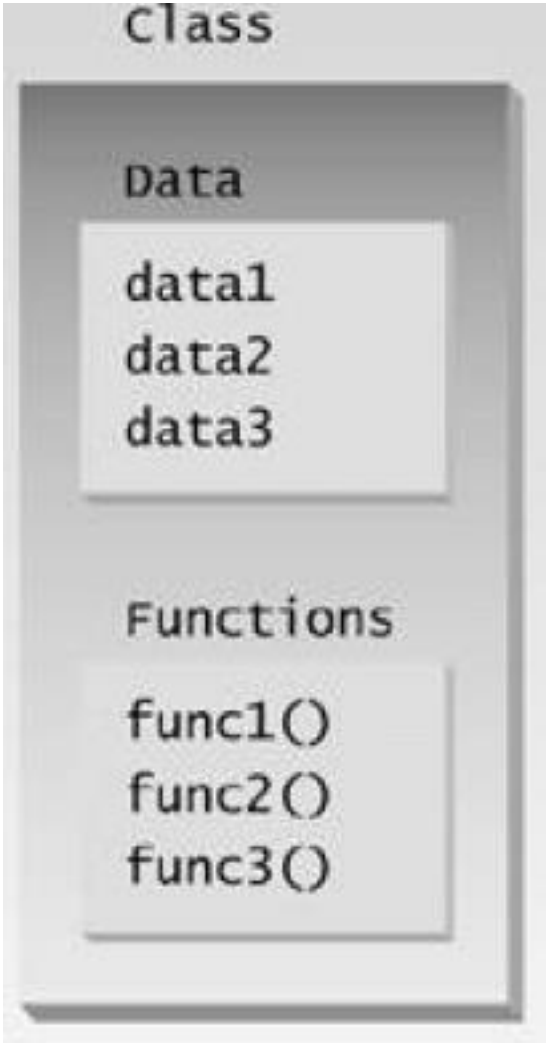
```
class class-name
{
        // private functions and variables
public:

        // public functions and variables
}  object-list (optional);
```

Syntax of a class specifier



Classes contain data and functions.



Simple your first OOP

```
class smallobj {      //declare a class
 private:
        int somedata;     //class data
public:
       void setdata(int d)       //member function to set data
              { somedata = d; }
      void showdata()         //member function to display data
              { cout << "\nData is " << somedata; }
};
int main( ){
   smallobj s1, s2;
   s1.setdata(1066);
   s2.setdata(1776);
    s1.showdata();
   s2.showdata();
   return 0;
}
```

Two objects of class smallobj

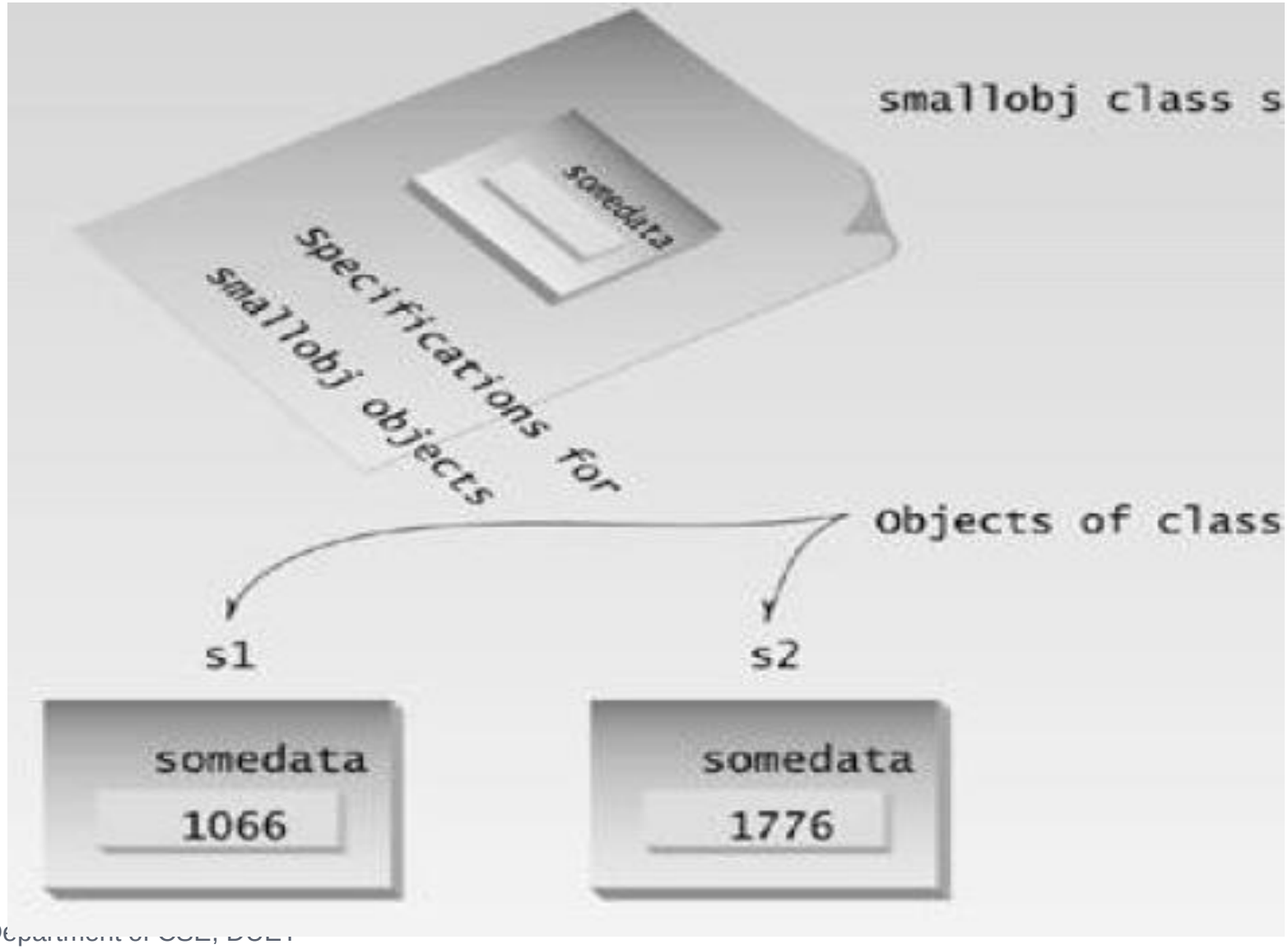
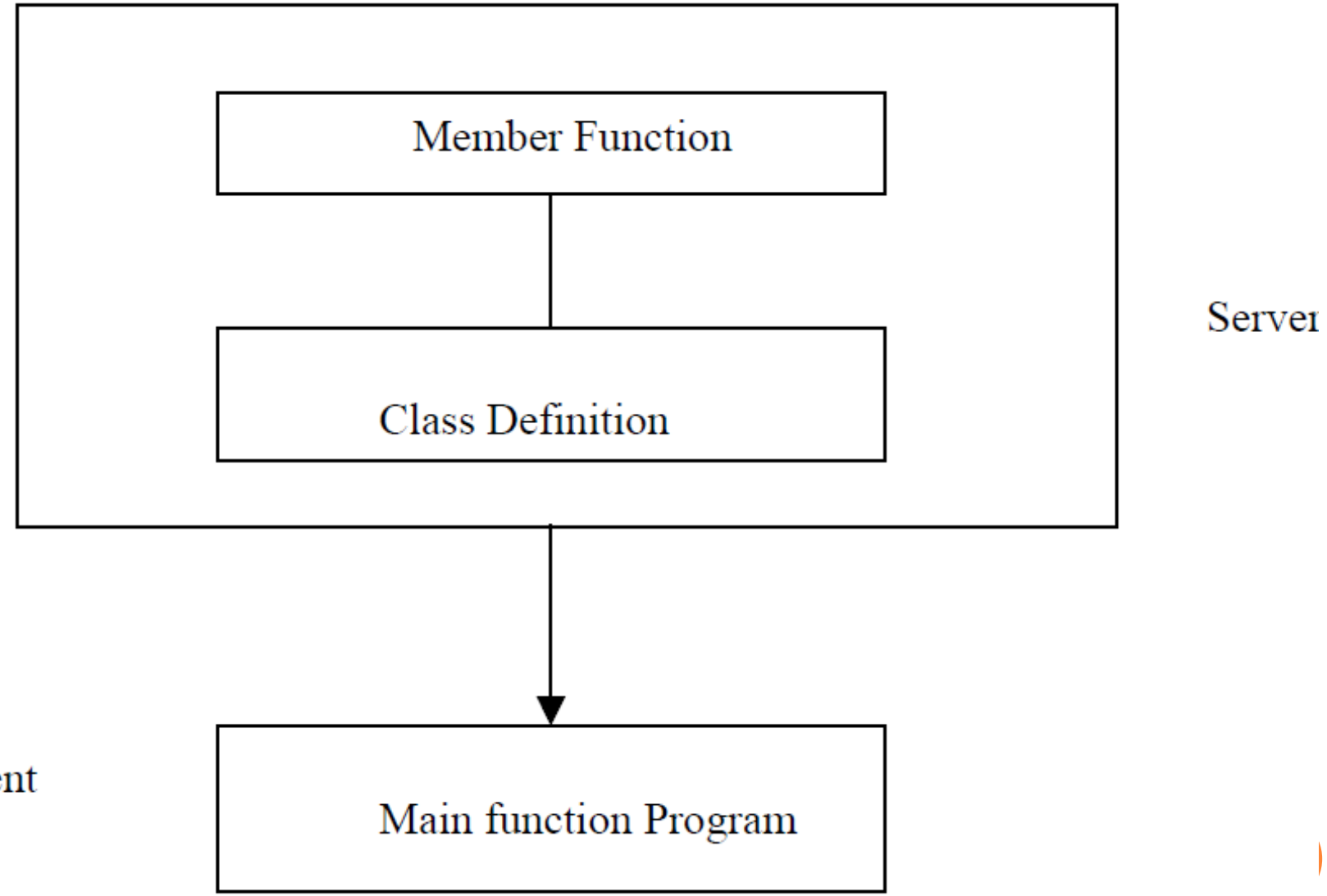


Fig. 1.10 *The client-server model*



Classes: A First Look (cont.)

- Each object of a class has its own copy of every variable declared within the class (except static variables which will be introduced later), but they all share the same copy of member functions.
  - How do member functions know on which object they have to work on?
    - The answer will be clear when "***this***" pointer is introduced.

Lecture Contents

- Teach Yourself C++
  - Chapter 1 (1.5 Full, with exercises)

- Object oriented Programming by C++ [R.Lafore]
  - Chapter 1 (With excercises)