C++ Programming – Lecture 5

Class organization

- Two ways:

1) Define member functions inside the class

```
class Sample
{
        public :
        void getData( )
        {
        }
} ;
```

2) Declare the prototype of member function inside the class and define the member function outside the class.

```
class Sample
{
        public :
                void getData( ) ;
} ;
Sample :: getData( )    // :: is scope resolution operator
{
}
```

this Pointer

- Whenever an object method is called, address of the object is passed to the method. The method collects it in a this pointer.
- this is a constant pointer and it cannot be modified during execution of the method
- this pointer dies once control returns from the method

Object Initialization Methods

- Three ways to initialize an object :

Method 1 : By declaring data members public
Limitation 1 - Data is liable for manipulation
Limitation 2 - Validation required at multiple places

Method 2 : Using member function like getData( ) / setData( )
Benefit 1 - Data is protected from manipulation
Benefit 2 - Better validation as it is done at one place
Benefit 3 - Validation done by class designer

Method 3 : Using special member function – Constructor
Benefit 1 - Program is better organized

Benefit 2 - Guaranteed initialization through constructor (Ctor)

- When an object is created, space is allocated in memory and Ctor is called
- Name of Ctor must be same as name of class
- Ctor is a function
- Ctor doesn't return any value
- Ctor gets called automatically when an object is created
- Ctor is called only once during entire lifetime of an object
- Ctor can be overloaded
- If we don't define a Ctor, compiler inserts a 0-arg Ctor
- A class may have Ctor as well as setData( )
- Ctor – To initialize object
- setData( ) – To modify object
- A destructor method is called when an object is about to be destroyed

Manager Functions

- Every C++ class has 4 manager functions:

  0-Arg constructor
  Destructor
  Copy constructor
  Overloaded = operator function

- 0-Arg constructor is called when an object is created without any parameters
- Destructor function is called when an object goes out of scope
- Copy constructor is call when an object is being created and is being initialized with another object of same type

  Complex c1 ;
  Complex c2 (c1) ;

- Copy constructor is also called when an object is passed to a function or object is returned from a function.
- Overloaded assignment operator function is called when an existing object is assigned to another existing object.