

1> When class contains objects of other class, constructor of member classes are called first

```
Eg class A{....};
    class B{....};
    class C{
        A a;
        B b;.....};
    C c //invoke in order A, B, C
```

2> when multiple objects of a class are declared, cons. are called in order of declaration

```
Eg Student S1, S2, S3; //order of cons is S1, S2 S3
```

3> Destructor are called in reverse order of constructor

4> Cons. and Dest. cannot be inherited, though derived class can call them

5> By default, visibility mode of base class is private

6> Private members of base class are never inherited

7> If vis is public, public mem. of base become public of derived, and protected become protected of derived

8> If vis is private, public and protected of base become private of derived

9> if vis is protected, they become protected of derived

10> First constructor of base is called and then cons. of derived. Order of destructor is opposite of it

11> When derived class has a function with same name as a function of base class, the derived class function hides the base class function. this is function overriding

12> In multiple base classes, cons are called in order in which base class appear in derived class declaration

13> If base class is inherited as virtual by 2 different classes D1 and D2 and D3 inherits D1 and D2, then only one copy of base class is present. Else, 2 copies will be there if its not virtual

14> If base class or a nested class constructor need arguments, the outer or derived class has to pass them explicitly

```
Eg struct TreeNode {
*   int val;
*   TreeNode *left;
*   TreeNode *right;
```

```

*     TreeNode(int x) : val(x), left(NULL), right(NULL) {}
* };
    TreeNode passes arguments for val, left and right explicitly
    better way to understand
    TreeNode(int x) : val(x), left(NULL), right(NULL){
    }

```

15> OOP concept

Data abstraction: Representing only key features

Data Encapsulation: Wrapping up data and associated function in one unit

Modularity: #include files

Inheritance: Inherit base class

Polymorphism: Function overloading, function overriding