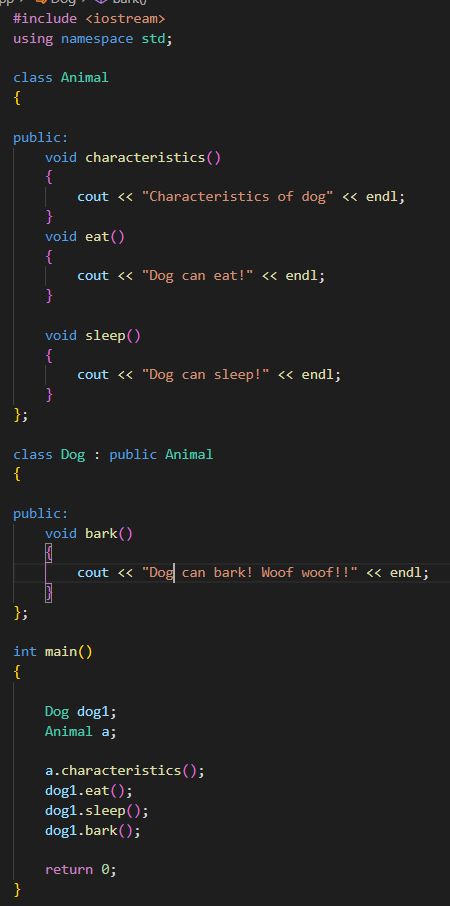
Lab – 5

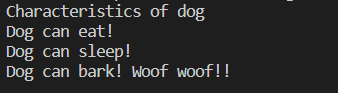
**Inheritance:** Derived Class declaration, Public, Private and Protected Inheritance, friend function and Inheritance, Overriding member function, Forms of inheritance, virtual base class, Abstract class, Constructor and Inheritance, Destructor and Inheritance, Advantage and disadvantage of Inheritance.

1. Write a simple base class, then a derived class and use objects of both of them in the main function. It will be a simple illustration of inheritance.

**Code:**

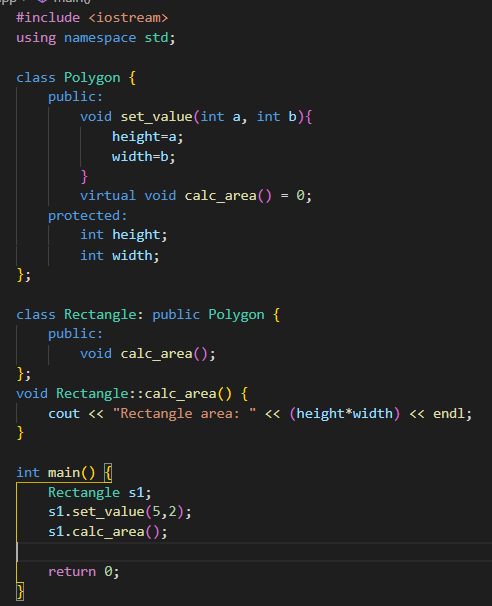
****

**Output:**

****

1. Practice protected access specifier in inheritance. In the base class declare a variable which is protected and access it in the derived class.

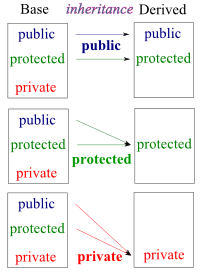
**Code:**

****

**Output:**

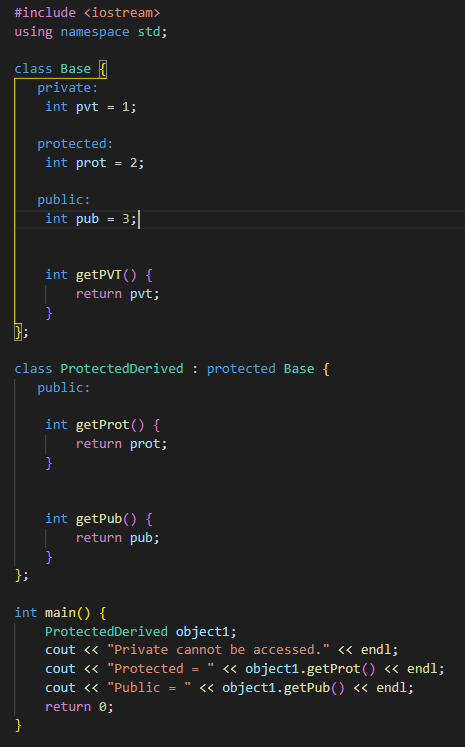
****

1. Most of the time we use public mode of inheritance, for example *class Derived: public Base{};*  Try protected and private access modifiers to understand the difference of various modes of inheritance.

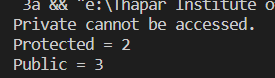


**1)Protected**

**Code:**

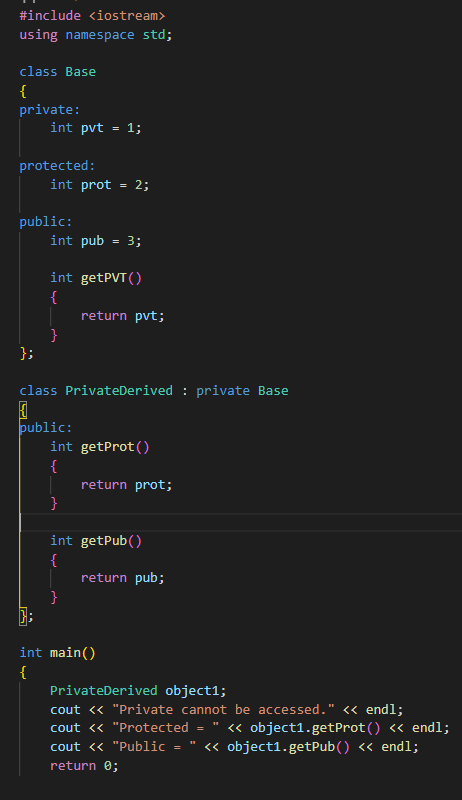
****

**Output:**

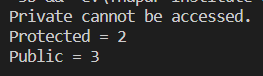
****

**2)Private:**

**Code:**

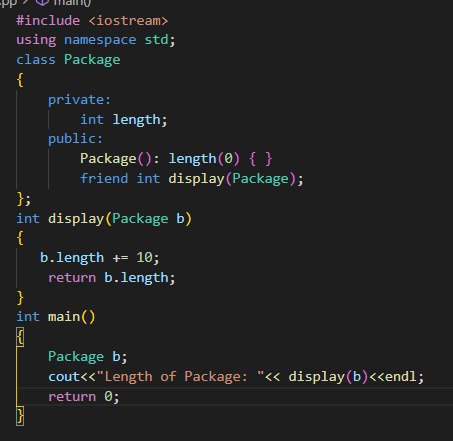
****

**Output:**

****

1. Write a C++ program to practice (i) friend function (ii) friend class.

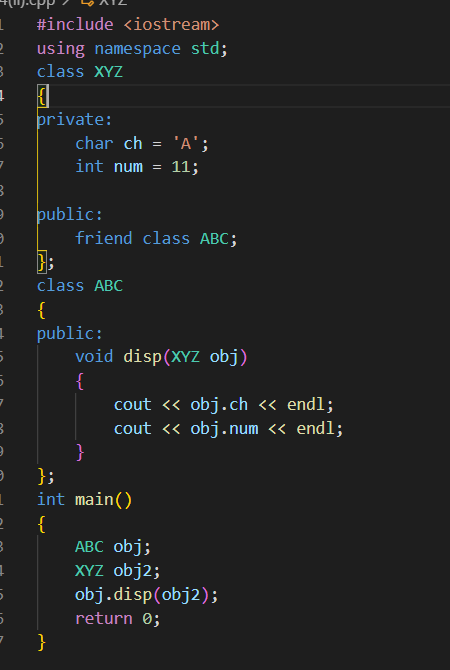
**(i)Code:**

****

**Output:**

****

**(ii)Code:**

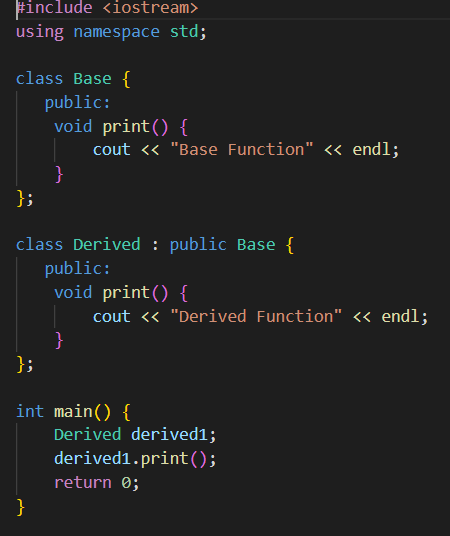
****

**Output:**

****

1. Illustrate the application of overriding the base class function through the derived class member function.

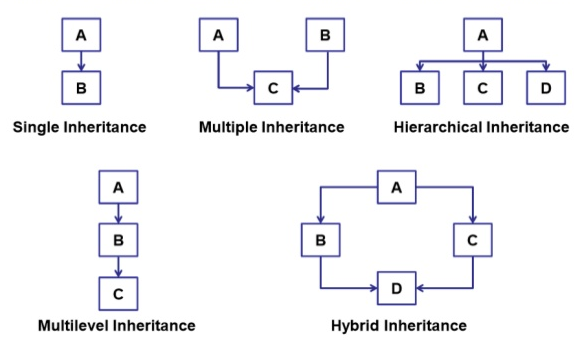
**Code:**

****

**Output:**

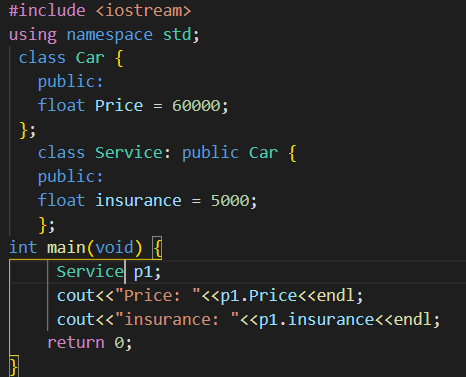
****

1. Various types of inheritances are as shown below. Write small C++ codes for each inheritance type.



**1)Single inheritance:**

**Code:**

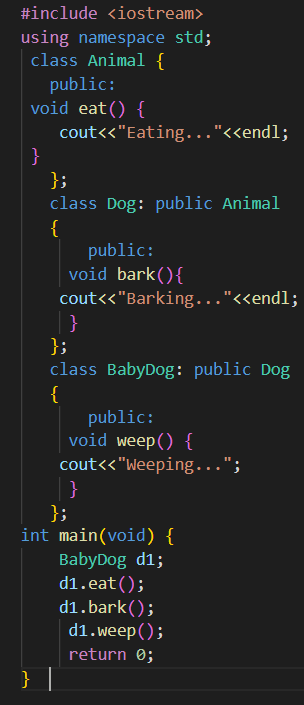
****

**Output:**

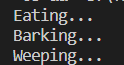
****

**2)Multilevel Inheritance:**

**Code:**

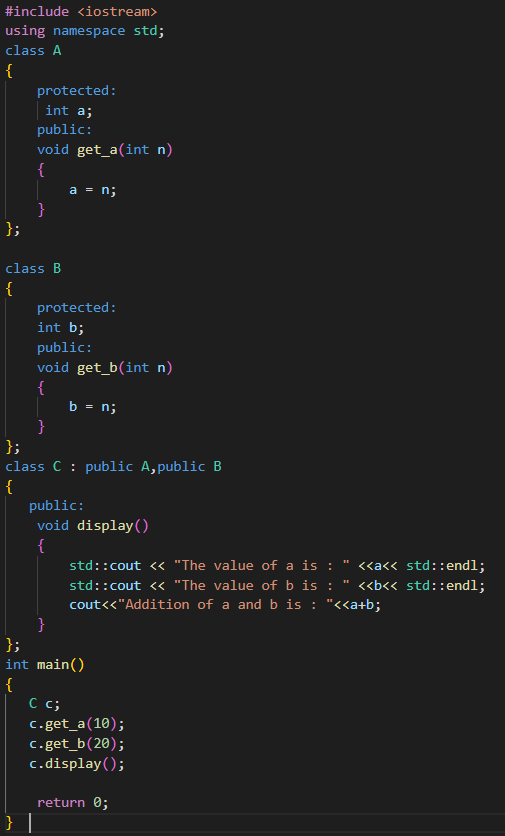
****

**Output:**

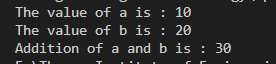
****

**3)Multiple Inheritance:**

**Code:**

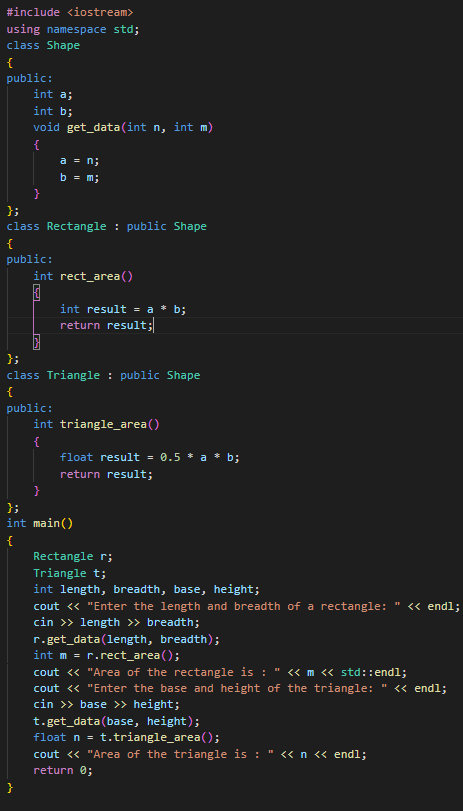
****

**Output:**

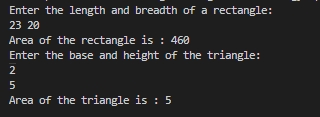
****

**4)Hierarchical Inheritance:**

**Code:**

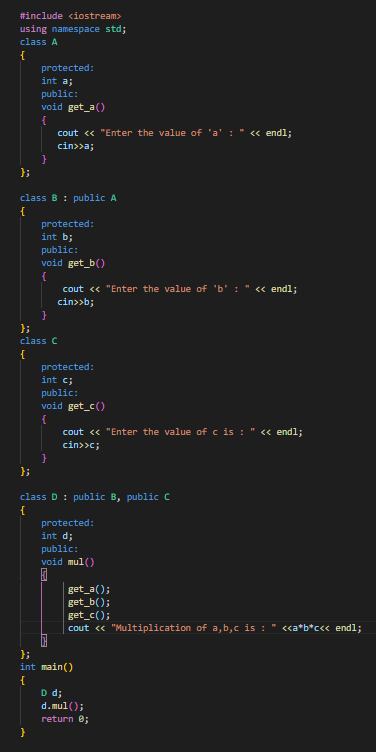
****

**Output:**

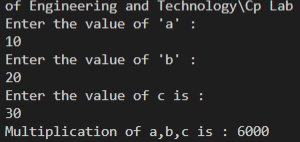
****

**5)Hybrid Inheritance:**

**Code:**

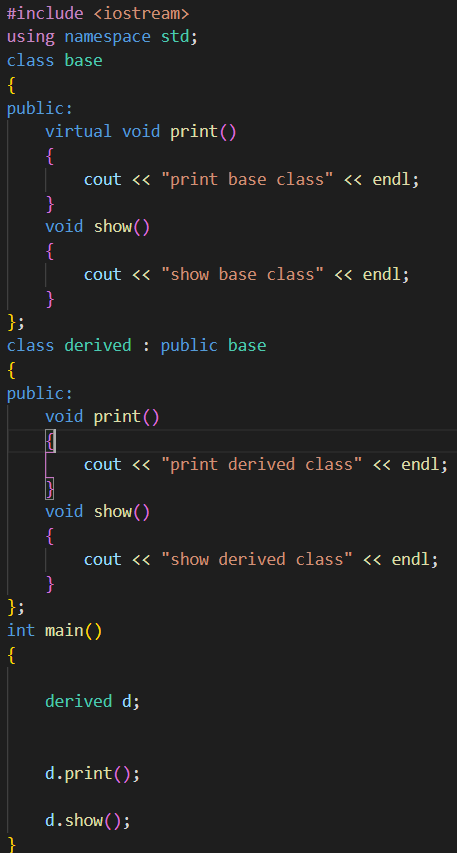
****

**Output:**

****

1. Write a program to create a virtual function in the base class and override (define) it in the child class.

**Code:**

****

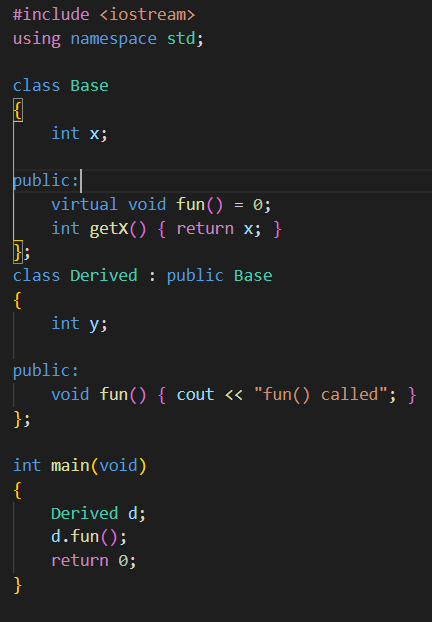
**Output:**



1. What is an abstract class? Write a program to illustrate and discuss with the lab instructor.

**Ans:**  **abstract class in C++** is a class that has at least one pure virtual function i.e. a function that has no definition. The classes inheriting the abstract class must provide a definition for the pure virtual function; otherwise, the subclass would become an abstract class itself.

**Code:**

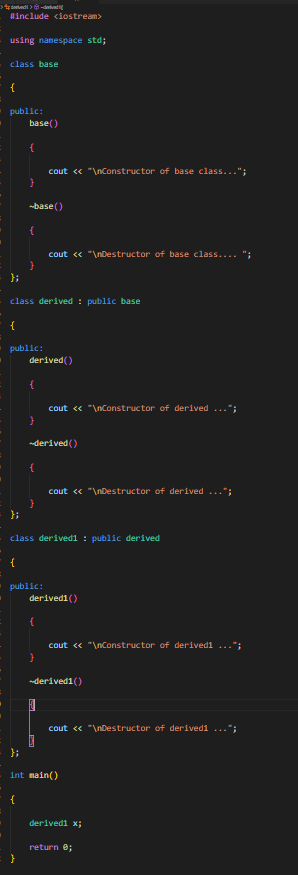
****

**Output:**

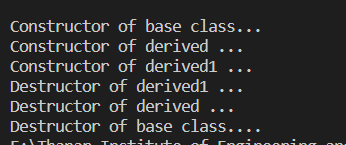
****

1. How can you override constructors and destructors in C++ inheritance? Write a program to illustrate.

**Code:**



**Output:**

****

1. Discuss the advantages and disadvantages of inheritance in the lab.

**Ans:**

**Disadvantages:**

* Inherited functions work slower than normal function as there is indirection.
* Improper use of inheritance may lead to wrong solutions.
* Often, data members in the base class are left unused which may lead to memory wastage.
* Inheritance increases the coupling between base class and derived class. A change in base class will affect all the child classes.

**Advantages:**

* Inheritance promotes reusability. When a class inherits or derives another class, it can access all the functionality of inherited class.
* Reusability enhanced reliability. The base class code will be already tested and debugged.
* As the existing code is reused, it leads to less development and maintenance costs.
* Inheritance makes the sub classes follow a standard interface.
* Inheritance helps to reduce code redundancy and supports code extensibility.
* Inheritance facilitates creation of class libraries.