Heaven's Light is Our Guide Rajshahi University of Engineering & Technology Department of Computer Science & Engineering

Lab Manual

Course Code: **CSE 1204 (Sec A & B)**Course Title: Sessional based on CSE 1203

Module 3 [Inheritance]: (for Week 4 [14-18/6/2025)

Problem Statement: You have the create an inheritance among **Father**-->**Son** -->**GrandSon** class. The **father** class has the following data members

```
class Father{
  private:
    int money;
  protected:
    int gold;
  public:
    int land;
};
```

Now write the **Son** and **GrandSon** classes with **private/protected/public** access modifier and do the following:

- i) Try to access **money**, **gold** and **land** from Son class
- ii) Try to access **money**, **gold** and **land** from GrandSon class
- iii) Find the values of money, gold and land when different access modifier is used in the following table
- iv) Display the sum of money, gold and land in Grandson class.

Class		In Son class			In GrandSon class		
Son	GrandSon	money	gold	land	money	gold	land
public	public	5	?	?	?	?	?
protected	public	?	?	?	?	?	?
private	public	?	?	?	?	?	?
public	protected	?	?	?	?	?	?
protected	protected		?	?	?	?	?
private	protected	?	?	?	?	?	?
public	private	?	?	?	?	?	?
protected	private	?	?	?	?	?	?
private	private	?	?	?	?	?	?

Topic 2 [**Types of Inheritance**]: Learn and Test different types of inheritance in C++. In each inheritance draw the class diagram with class chain and try to access the data members of bases classes from child classes.

i) Single inheritance

```
class A{
               class B:public A{
                                      int main(){
                 //write public
  private:
                                       B b;
               method to //access
    int x;
                                      //call methods of
 protected:
                                      class B
               x,y & z
  int y;
                                      return 0;
public:
                                      }
  int z;
```

ii) Multi-level inheritance

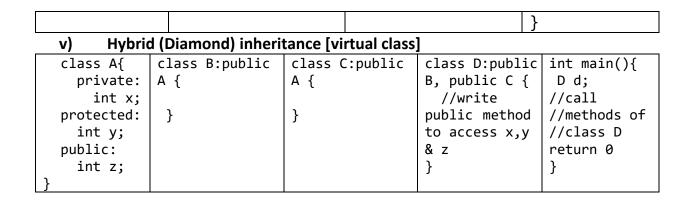
class A{	class B:public	class C:public	<pre>int main(){</pre>		
private:	Α{	B{	C c;		
<pre>int x;</pre>	}	//write	//call		
protected:		<pre>public //method</pre>	//methods of		
<pre>int y;</pre>		to //access x,y	//class C		
<pre>public:</pre>		& z	return 0		
int z;		}	}		
}					

iii) Multiple inheritance

ing wattiple interitance					
class A{	class B{	class C:public A,	<pre>int main(){</pre>		
private:	private:	Public B{	C c;		
int x;	int p;	//write public	//call		
protected:	protected:	method //to	//methods of		
int y;	int q;	access	//class C		
public:	<pre>public:</pre>	//x,y,z,p,q & r	return 0		
int z;	int r;	}	}		
}	}				

iv) Heirarchical inheritance

```
int main(){
class A{
            class B:public A
                                class C:public A
  private:
                                                    B b;
                                {
                                  //write method
            //write public
                                                    C c;
    int x;
            method to access
protected:
                                public to access
                                                   //call
                                x,y & z
                                                   //methods of
  int y;
            x,y & z
public:
                                }
                                                   //class B &
              }
  int z;
                                                   return 0
```



Topic 3 [Constructor & Destructor in inheritance]: Write the constructors & destructors for different types of inheritance are given as follows. Also follow and write the sequence of their execution.

i) Single inheritance

```
class A{
                    class B:public A{
                                                 int main(){
  private:
                       private:
                                                  B b;
                                                 //call methods of
                         int bx;
    int ax;
                      public:
                                                 class B
public:
  //write
                    //write constructor to
                                                 return 0;
                    initialize bx
  constructor to
  initialize ax
                    //Write method to sum ax
  //Write
                    and bx
                    //Write destructor
  destructor
```

ii) Multi-level inheritance

```
private:
                   class B:public A {
                                       class C:public B {
                                                             int main(){
   int ax;
                     private:
                                         private:
                                                             C c;
public:
                                                             //call
                        int bx;
                                             int cx;
 //write
                                                             //methods of
                     public:
                                          public:
 constructor to
                       //write
                                           //write
                                                             //class C
 initialize ax
                       constructor to
                                           constructor to
                                                             return 0
 //Write
                       initialize bx
                                           initialize cx
                                                             }
 destructor
                       //Write
                                        //Write method to
                       destructor
                                       sum ax, bx and cx
 }
                                       //Write destructor
```

iii) Multiple inheritance

iii) ividitiple iiiileritanee					
private:	class B{	class C:public A,	<pre>int main(){</pre>		
<pre>int ax;</pre>	private:	Public B{	C c;		
<pre>public:</pre>	int bx;	private:	//call		
//write	<pre>public:</pre>	int cx;	//methods of		
constructor	//write	<pre>public:</pre>	//class C		
to initialize	constructor	//write	return 0		
ax	to initialize	constructor to	}		
//Write	bx	initialize cx			
destructor	//Write	//Write method to sum			
}	destructor	ax, bx and cx			
	}	//Write destructor			
		}			

iv) Heirarchical inheritance

```
class A{
                                   class C:public A { | int main(){
                 class B:public A
private:
                                     private:
                                                         B b;
    int ax;
                  private:
                                         int cx;
                                                         C c;
public:
                      int bx;
                                      public:
                                                        //call
//write
                                       //write
                                                        //methods of
                   public:
                                                        //class B & C
constructor to
                    //write
                                       constructor to
initialize ax
                    constructor
                                       initialize cx
                                                        return 0
//Write
                    to initialize
                                   //Write method to
                                   sum ax, bx and cx
destructor
                    bx
 }
                    //Write
                                    //Write destructor
                    destructor
                                    }
```

v) Hybrid (Diamond) inheritance [virtual class]

	r '-			1 .
class A{	class	<pre>class C:public A {</pre>	class D:public B,	int
private:	B:public A {	private:	<pre>public C {</pre>	<pre>main(){</pre>
int ax;	private:	<pre>int cx;</pre>	private:	D d;
<pre>public:</pre>	int bx;	<pre>public:</pre>	int dx;	//call
//write	public:	//write	<pre>public:</pre>	//methods
constructo	//write	constructor	//write	of
r to	constructo	to	constructor to	//class D
initialize	r to	initialize	initialize dx	return 0
ax	initialize	CX	//Write method	}
//Write	bx	//Write	to sum ax, bx	
destructor	//Write	* *	cx and dx	
}	destructor	destructor		
			//Write	
	}	}	destructor	
			}	