**Assignment number:6**

**Subject: Object Oriented** Programming

**INHERITANCE**

Name: ***RIA MITTAL***

Class: ***SECOND YEAR ENGINEERING***

Division: ***B***

Roll no: ***222008***

Batch: ***B1***

**PROBLEM STATEMENT:**

Employee bio data using following classes: -

1. Personal record.

2. Professional record.

3. Academic record.

Assume appropriate data members and member functions to accept and display required data and at the end display the complete bio data of employee. Use multiple inheritances in C++.

**CONCEPTS:**

Inheritance

**THEORY:**

Inheritance in Object Oriented Programming can be described as a process of creating new classes from existing classes. New classes inherit some of the properties and behavior of the existing classes. An existing class that is "parent" of a new class is called a **base class**. New class that inherits properties of the base class is called a **derived class**. Inheritance is a technique of code reuse. It also provides possibility to extend existing classes by creating derived classes.

The basic syntax of inheritance is:

Class Derived\_Class : Access\_Specifier Base\_Class

There are 3 access specifiers:

Namely public, private and protected.

**public:**

This inheritance mode is used mostly. In this the protected member of Base class becomes protected members of Derived class and public becomes public.

**protected:**

In protected mode, the public and protected members of Base class becomes protected members of Derived class.

**private:**

In private mode the public and protected members of Base class become private members of Derived class.

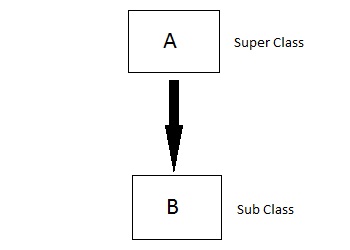
**TYPES OF INHERITANCE:**

In C++, we have 5 different types of Inheritance. Namely,

1. Single Inheritance
2. Multiple Inheritance
3. Multilevel Inheritance
4. Hybrid Inheritance
5. Hierarchical Inheritance

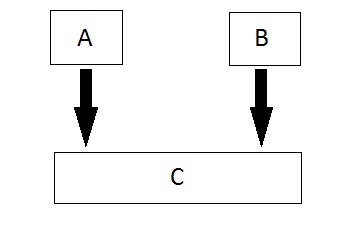
**1. SINGLE INHERITANCE**

In this type of inheritance one derived class inherits from only one base class. It is the simplest form of Inheritance.



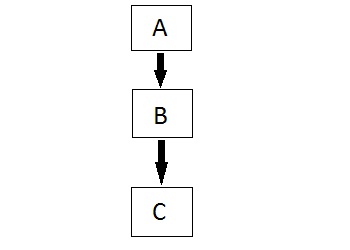
**2. MULTIPLE INHERITANCE**

In this type of inheritance a single derived class may inherit from two or more than two base classes.



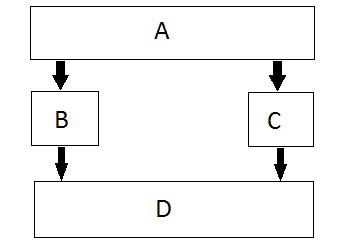
**3. MULTILEVEL INHERITANCE**

In this type of inheritance the derived class inherits from a class, which in turn inherits from some other class. The Super class for one is sub class for the other.



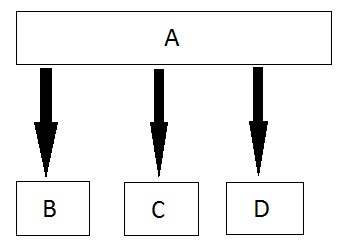
**4. HYBRID INHERITANCE**

Hybrid Inheritance is combination of any 2 or more types of inheritances.



**5. HIERARCHICAL INHERITANCE**

In this type of inheritance, multiple derived classes inherit from a single base class.



**ALGORITHM:**

1. Start.

2. Create a class named personal\_record.

3. Declare data members name, add, dob, email, phone as private members of the class.

4. Declare public members: get\_per\_rec() to accept values from user, put\_per\_rec() to display the values.

5. Create another class named professional\_record.

6. Declare data members empid desig, join, depart, salary, exp as private members of the class.

7. Declare public members : get\_prof\_rec() accept values from user, put\_prof\_rec() to display the values.

8. Create another class named academic\_record which privately inherites data from class personal\_record and professional\_record.

9. Declare data members empid, desig, join, depart, salary, exp as private members of the class.

10. Declare public members : get\_aca\_rec() accept values from user, put\_aca\_rec() to display the values.

11. In main function create object of class academic\_record obj.

12. Using obj access get\_per\_rec(), get\_prof\_rec() and get\_aca\_rec() to accept input from user.

13. Using obj access put\_per\_rec(), put\_prof\_rec() and put\_aca\_rec() to display the inputted data.

14. Stop.

**STATE TRANSITION DIAGRAM:**

Q0

Qe

Q6

Q5

Q4

Q3

Q2

Q1

**Q0** – Start program, initialize variables.

**Q1** –call function get\_per\_rec() which originally belonged to personal\_record using object of academic\_record.

**Q2** –call function get\_prof\_rec() which originally belonged to professional\_record using object of academic\_record.

**Q3** –call function get\_aca\_rec() belonging to academic\_record using object of academic\_record.

**Q4** –call function put\_per\_rec() which originally belonged to personal\_record using object of academic\_record.

**Q5** –call function put\_prof\_rec() which originally belonged to professional\_record using object of academic\_record.

**Q6** –call function put\_aca\_rec() which belongs academic\_record using object of academic\_record.

**Qe** –end of program.

**PROGRAM:**

#include<iostream>

using namespace std;

//class 1

class personal\_record

{

char name[100],address[100];

float height;

int age;

long long int mob\_no;

public:

int accept1();

int display1();

};

//class 2

class professional\_record

{

char gr[7];

char depart[100],project[100];

long long int emp\_no;

public:

int accept2();

int display2();

};

//class3

class academic\_record

{

char degree[100],clg\_name[100];

float per1,per2;

public:

int accept3();

int display3();

};

// class 4

class biodata:public personal\_record,public professional\_record ,public academic\_record

{

public:

int display();

};

int biodata::display()

{

cout<<"\n\n\t\tWELCOME TO STUDENT PROFILE SYSTEM\n\n";

cout<<"\n\t\t\t\tSTUDENT BIODATA\t\t\t\n\n\n";

}

int personal\_record::accept1()

{

cout<<"\nENTER YOUR NAME\n";

cin.getline(name,100);

cout<<"\nENTER YOUR AGE\n";

cin>>age;

cout<<"\nENTER YOUR HEIGHT\n";

cin>>height;

cout<<"\nENTER YOUR MOBILE NUMBER\n";

cin>>mob\_no;

cout<<"\nENTER YOUR ADDRESS\n";

ws(cin);

cin.getline(address,100);

}

int personal\_record::display1()

{

cout<<"\nNAME : "<<name;

cout<<"\nAGE : "<<age;

cout<<"\nHEIGHT : "<<height;

cout<<"\nMOBILE NUMBER : "<<mob\_no;

cout<<"\nADDRESS : "<<address;

}

int professional\_record::accept2()

{

cout<<"\nENTER YOUR EMPLOYEE NUMBER\n";

cin>>emp\_no;

cout<<"\nENTER YOUR 7 DIGIT GR.NO.\n";

cin>>gr;

cout<<"\nDEPARTMENT NAME\n";

ws(cin);

cin.getline(depart,100);

cout<<"\nPROJECT NAME\n";

ws(cin);

cin.getline(project,100);

}

int professional\_record::display2()

{

cout<<" \nEMPLOYEE NUMBER : "<<emp\_no;

cout<<"\nG.R NUMBER : "<<gr;

cout<<" \nDEPARTMENT NAME : "<<depart;

cout<<"\nPROJECT NAME : "<<project;

}

int academic\_record::accept3()

{

cout<<"\nENTER DEGREE NAME\n";

ws(cin);

cin.getline(degree,100);

cout<<"\nENTER 10TH PERCENTAGE\n";

cin>>per1;

cout<<"\nENTER 12TH PERCENTAGE\n";

cin>>per2;

cout<<"\nENTER COLLEGE NAME\n";

ws(cin);

cin.getline(clg\_name,100);

}

int academic\_record::display3()

{

cout<<"\nDEGREE : "<<degree;

cout<<"\n10TH PERCENTAGE : "<<per1;

cout<<"\n12TH PERCENTAGE : "<<per2;

cout<<"\nCOLLEGE NAME : "<<clg\_name<<endl;

}

//main

int main()

{

biodata obj3;

obj3.accept1();

obj3.accept2();

obj3.accept3();

cout<<"\n------------------------------------------------------------------------";

obj3.display();

cout<<"------------------------------------------------------------------------";

obj3.display1();

cout<<"\n------------------------------------------------------------------------";

obj3.display2();

cout<<"\n------------------------------------------------------------------------";

obj3.display3();

cout<<"\n------------------------------------------------------------------------";

cout<<"\n\n\nTHANK YOU FOR VISITING STUDENT PROFILE\n\n";

return 0;

}

**OUTPUT:**

ENTER YOUR NAME

ria mittal

ENTER YOUR AGE

19

ENTER YOUR HEIGHT

5.8

ENTER YOUR MOBILE NUMBER

8237529877

ENTER YOUR ADDRESS

98/838,abc def

ENTER YOUR EMPLOYEE NUMBER

123

ENTER YOUR 7 DIGIT GR.NO.

u1610212

DEPARTMENT NAME

computer

PROJECT NAME

oop

ENTER DEGREE NAME

be

ENTER 10TH PERCENTAGE

94

ENTER 12TH PERCENTAGE

76

ENTER COLLEGE NAME

viit

------------------------------------------------------------------------

WELCOME TO STUDENT PROFILE SYSTEM

STUDENT BIODATA

------------------------------------------------------------------------

NAME : ria mittal

AGE : 19

HEIGHT : 5.8

MOBILE NUMBER : 8237529877

ADDRESS : 98/838,abc def

------------------------------------------------------------------------

EMPLOYEE NUMBER : 123

G.R NUMBER : u161021computer

DEPARTMENT NAME : computer

PROJECT NAME : oop

------------------------------------------------------------------------

DEGREE : be

10TH PERCENTAGE : 94

12TH PERCENTAGE : 76

COLLEGE NAME : viit

------------------------------------------------------------------------

THANK YOU FOR VISITING STUDENT PROFILE