

Assignment 6

Submission Date- 02/12/2024

Title – Study of Inheritance in C++.

Objective- 1. To understand concept of Inheritance and their usage in C++.

2. To Create bio-data using multiple inheritance using C++

Problem Statement-

Create employee bio-data using following classes:

- i) Personal record
- ii) Professional record
- iii) Academic record Assume appropriate data members and member function to accept required data & print bio-data.

Software & Hardware requirements- any Text editor and Terminal in Linux/
Turbo C++ Compiler installed on PC.

Theory-

C++ Inheritance

In C++, inheritance is a process in which one object acquires all the properties and behaviors of its parent object automatically. In such way, you can reuse, extend or modify the attributes and behaviors which are defined in other class.

In C++, the class which inherits the members of another class is called derived class and the class whose members are inherited is called base class. The derived class is the specialized class for the base class.

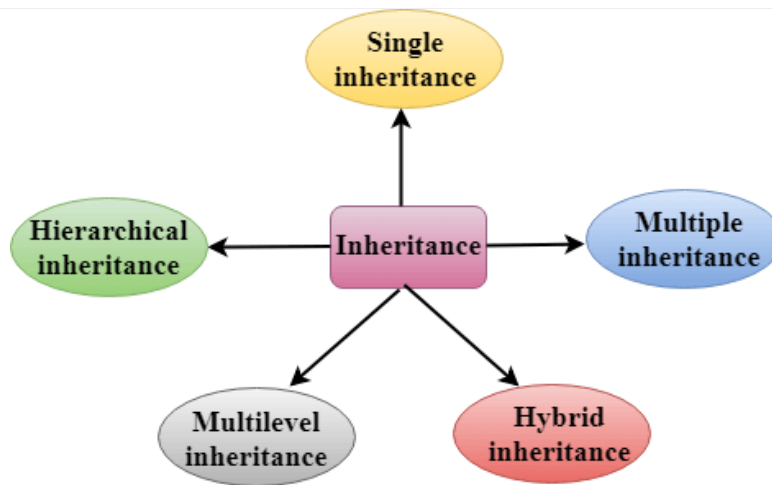
Advantage of C++ Inheritance

Code reusability: Now you can reuse the members of your parent class. So, there is no need to define the member again. So less code is required in the class.

Types of Inheritance

C++ supports five types of inheritance:

- Single inheritance
- Multiple inheritance
- Hierarchical inheritance
- Multilevel inheritance
- Hybrid inheritance



Derived Classes

A Derived class is defined as the class derived from the base class.

The Syntax of Derived class:

```
class derived_class_name :: visibility-mode base_class_name
{
    // body of the derived class.
}
```

Where,

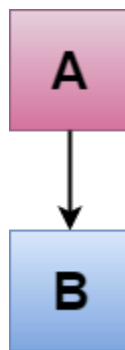
derived_class_name: It is the name of the derived class.

Visibility mode: The visibility mode specifies whether the features of the base class are publicly inherited or privately inherited. It can be public or private.

base_class_name: It is the name of the base class.

C++ Single Inheritance

Single inheritance is defined as the inheritance in which a derived class is inherited from the only one base class.



Where 'A' is the base class, and 'B' is the derived class.

When one class inherits another class, it is known as single level inheritance.

Example of single level inheritance which inherits the fields only.

```
#include <iostream>
using namespace std;
class Account {
public:
float salary = 60000;
```

```
};  
  
class Programmer: public Account {  
public:  
    float bonus = 5000;  
};  
  
int main(void) {  
    Programmer p1;  
    cout<<"Salary: "<<p1.salary<<endl;  
    cout<<"Bonus: "<<p1.bonus<<endl;  
    return 0;  
}
```

C++ Multilevel Inheritance

Multilevel inheritance is a process of deriving a class from another derived class.



When one class inherits another class which is further inherited by another class, it is known as multi-level inheritance in C++. Inheritance is transitive so the last derived class acquires all the members of all its base classes.

Example of multi-level inheritance in C++.

```
#include <iostream>

using namespace std;

class Animal {
    public:
    void eat() {
        cout<<"Eating..."<<endl;
    }
};

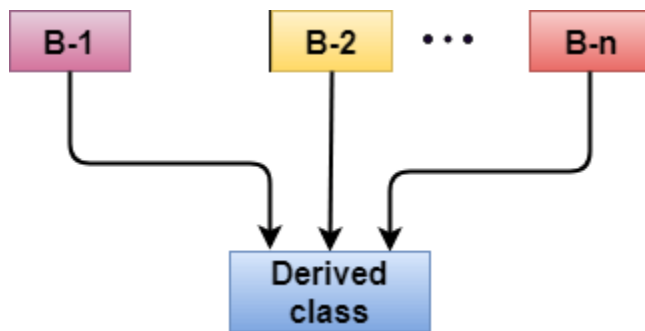
class Dog: public Animal
{
    public:
    void bark(){
        cout<<"Barking..."<<endl;
    }
};

class BabyDog: public Dog
{
    public:
    void weep() {
        cout<<"Weeping...";
    }
};
```

```
int main(void) {  
    BabyDog d1;  
    d1.eat();  
    d1.bark();  
    d1.weep();  
    return 0;  
}
```

C++ Multiple Inheritance

Multiple inheritance is the process of deriving a new class that inherits the attributes from two or more classes.

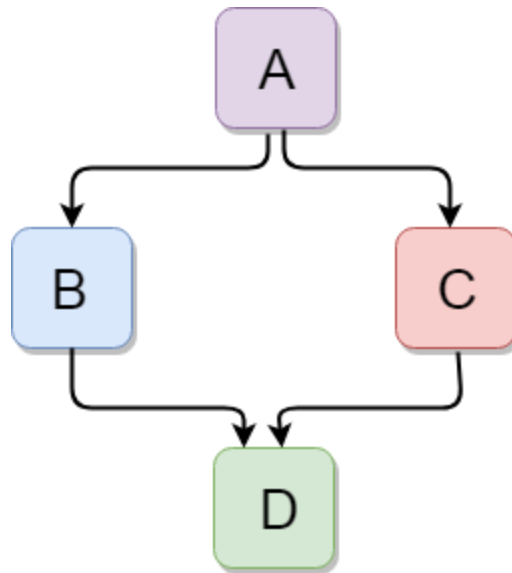


Syntax of the Derived class:

```
class D : visibility B-1, visibility B-2, ?  
{  
    // Body of the class;  
}
```

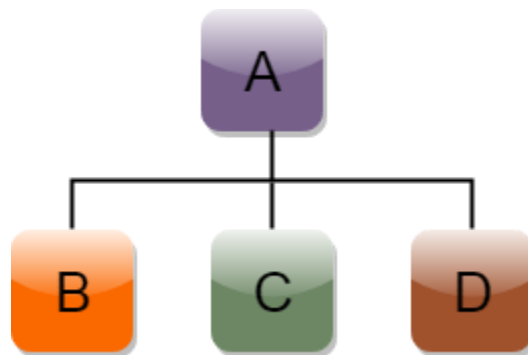
C++ Hybrid Inheritance

Hybrid inheritance is a combination of more than one type of inheritance.



C++ Hierarchical Inheritance

Hierarchical inheritance is defined as the process of deriving more than one class from a base class.



Syntax of Hierarchical inheritance:

```
class A
{
    // body of the class A.
}
class B : public A
```

```
{
    // body of class B.
}
class C : public A
{
    // body of class C.
}
class D : public A
{
    // body of class D.
}
```

Code

```
#include<iostream>
using namespace std;
class personal
{
protected:
char name[50];
char address[50];
char birthdate[50];
char gender;
public:
void get_personal()
{
cout<<"\nEnter name";
cin>>name;
cout<<"\nEnter Address";
cin>>address;
```



```

cout<<"\nEnter Birthdate(dd/mm/yyyy)";
cin>>birthdate;
cout<<"\nEnter gender(M/F)";
cin>>gender;
}
};
class professional
{
protected:
int expinnoofyear;
char orgname[50];
char projname[50];
char projdetails[50];
public:
void get_professional()
{
cout<<"\nEnter number of years of exp";
cin>>expinnoofyear;
cout<<"\nEnter organization name";
cin>>orgname;
cout<<"\nEnter project name";
cin>>projname;
cout<<"\nEnter project Details";
cin>>projdetails;
}
};
class academic
{
protected:
int year;
int marks;
int percentage;
char clas[50];
public:
void get_academic()

```

```

{
cout<<"\nEnter academic year";
cin>>year;
cout<<"\nEnter total marks";
cin>>marks;
cout<<"\nEnter percentage";
cin>>percentage;
cout<<"\nEnter class";
cin>>clas;
}
};
class biodata: public personal, public academic,public professional
{
public:
void display()
{
cout<<"\n-----Employee Biodata-----"<<endl;
cout<<"-----"<<endl;
cout<<"Personal Details"<<endl;
cout<<"Name:"<<name<<endl;
cout<<"Address:"<<address<<endl;
cout<<"Birthdate:"<<birthdate<<endl;
cout<<"Gender:"<<gender<<endl;
cout<<"-----"<<endl;
cout<<"-----Academic Details-----"<<endl;
cout<<"Academic
Year\t"<<"marks\t"<<"percentage\t"<<"class\t"<<endl;
cout<<year<<"\t"<<marks<<"\t"<<percentage<<"\t"<<clas<<endl;
cout<<"-----"<<endl;
cout<<"-----Professional
Details-----"<<endl;
cout<<"Organization Name:"<<orgname<<endl;
cout<<"Years of Experince:"<<expinnoofyear<<endl;
cout<<"Project Done:"<<projname<<endl;
cout<<"Project Details:"<<projdetails<<endl;

```

```

}
};
int main()
{
biodata b;
b.get_personal();
b.get_academic();
b.get_professional();
b.display();
return 0;
}

```

```
/*student@student-OptiPlex-3010:~$ g++ groupa13.cpp
```

```
student@student-OptiPlex-3010:~$ ./a.out
```

Enter nameXYZ

Enter AddressPUNE

Enter Birthdate(dd/mm/yyyy)15/5/2016

Enter gender(M/F)F

Enter academic year2016

Enter total marks500

Enter percentage70

Enter classFirst

Enter number of years of exp5

Enter organization namePQR

Enter project namePPP

Enter project DetailsIII

-----Employee Biodata-----

Personal Details

Name:XYZ

Address:PUNE

Birthdate:15/5/2016

Gender:F

-----Academic Details-----

Academic Year marks percentage class

2016 500 70 First

-----Professional Details-----

Organization Name:PQR
Years of Experince:5
Project Done:PPP
Project Details:III
student@student-OptiPlex-3010:~\$
*/

Conclusion-

Well Understanding concept of Inheritance and their usage in C++.