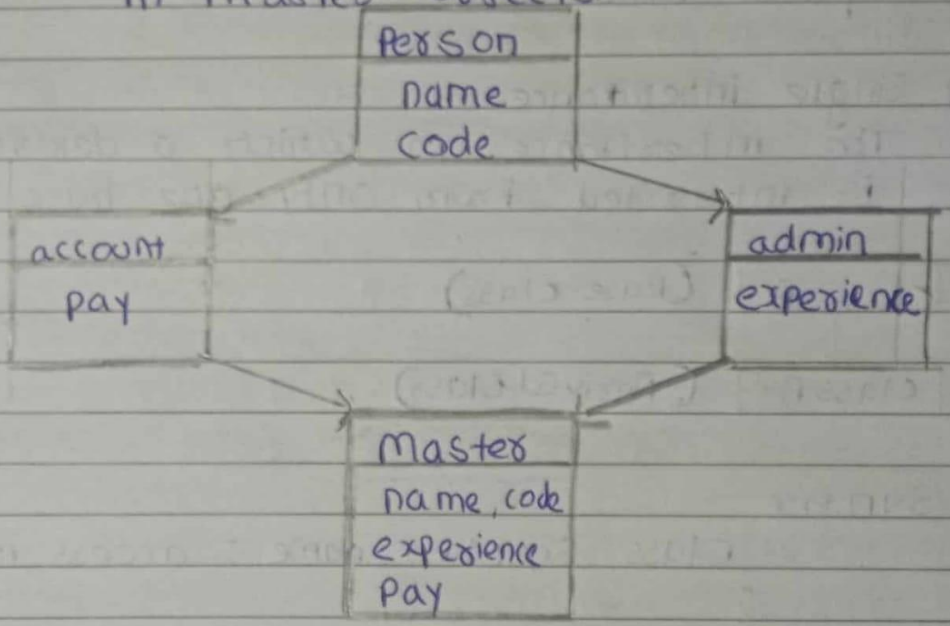
G. H. Raisoni College Of Engineering And Management, Wagholi Pune Assignment no :- 6 2021- 2022			
Department	CE [SUMMER 2022 (Online)]		
Term / Section	III/B	Date Of **submission**	28-10-2021
Subject Name /Code	Object Oriented Programming/ UTIL201/UITP201		
Roll No.	SCOB77	Name	Pratham Rajkumar pitty
Registration Number	2020AC0E1100107		

SCOB77_Prathamp Pithy

Assignment no 6

Aim :→ Consider class network of given figure. The class master derives information from both account and admin classes which is turn derive information from the class person. Define all the four classes and write a program to create, Update and display the information contained in master objects.



Theory :→

Inheritance

The capability of a class to derive properties and characteristics from another class is called Inheritance

In C++

The class which inherits the members of another class is called derived class.

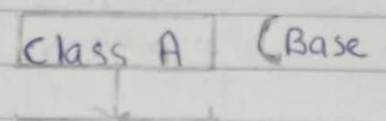
The class whose member is inherited is called as Base class.

Types of inheritance

cpp supports five types of inheritance

(1) Single inheritance
(2) multiple inheritance
(3) Hierarchical inheritance
(4) Multileval inheritance
(5) Hybrid inheritance

- (1) Single inheritance
  The inheritance in which a derived class is inherited from only one base class.



Syntax

```
class Subclass_name : access_mode base
{
  // body of subclass
};
```

extra:→ part about C++

access mode :- There are 3 access mode in

- public mode :→ If we derive a sub class from a public base class. Then the public member of the base class will become in the derived class and protected member of the base class will become protected in derived class.
- Private mode :→ when the member is declared

as private, its accessible within the class only.

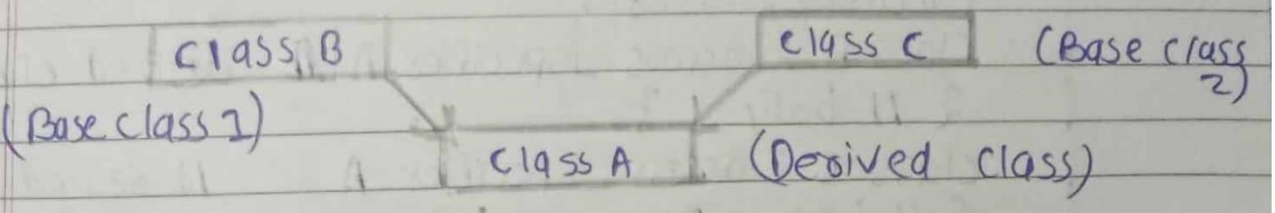
- protected:→ When the member is declared as protected it is accessible with its own class as well as the class immediatly derived from it.

In short,

Base class member access specifier	Types of inheritence		
	Public	Protected	Private.
Public	Public	protected	Private
Protected	Protected	Protected	Private
Private	Not accessible (Hidden)	Not accessible (Hidden)	Not accesible (Hidden)

- (2) multiple inheritance

Multiple inheritance is a feachere of c++ where a class can inherit from more than one classes.
ie one sub class is inherited from more than one base classes.

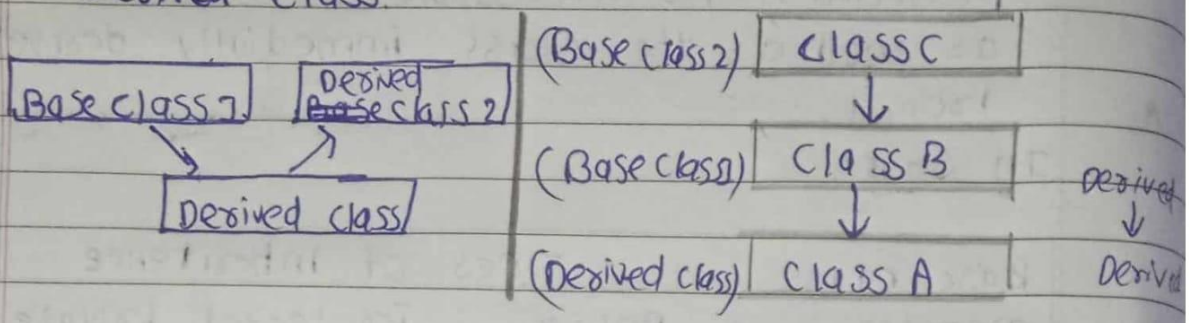


Syntax :-

```
class subclass_name: access_mode base_class1,
                     access_mode base_class2,....
{
  // body of sub class
};
```

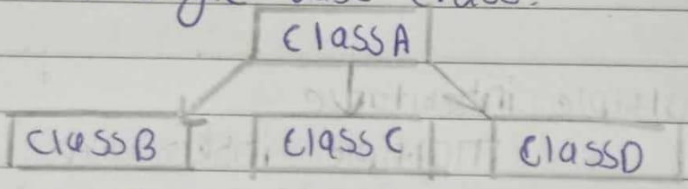
(3) Multilevel Inheritance

A derived class is created from another derived class.



(4) Hierarchical inheritance

In this inheritance more than one sub class is inherited from a single base class i.e. more than one derived class is created from a single base class.



Class B, C, D are derived from same Base class A

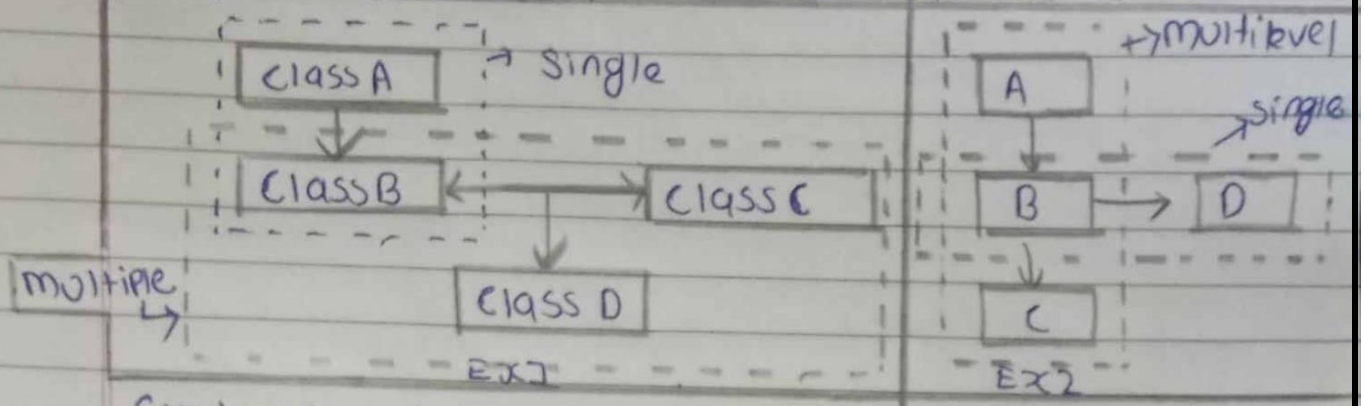
Syntax :-

```
class A        // base class
{ //body ..... } ;
Class B : access_specifier A // derived class from
{ // body .... };
class C : access_specifier A      // derived class
{ // body ... };                      from A
Class D : access_specifier A      // derived Class
{ // body... };
```

(5) Hybrid ~~class~~ inheritance

The inheritance in which the derived derivation of class involves more than one ~~class~~ Form

of inheritance. like combination of multiple, Simple, hierarchical, multilevel inheritance.



Syntax :- ex1

Ex1

```
class A
{ //Statement};
Class B : Public A
{ // Statement..};
Class C :
{ // Statement..};
Class D : Public B, Public C
{ // Statement..};
```

Ex2

```
Class A
{ // Statement...};
Class B : Public A
{ // Statement...};
Class C : Public B
{ // Statement...};
Class D : Public B
{ // Statement...};
```

Program code

```
/*
Consider class network of given figure. The class master derives information from both account
and admin classes which in turn derive information from the class person. Define all the four classes
and write a program to create, update and display the information contained in master objects
*/

#include<iostream>
#include<string.h>
using namespace std;

class person
{
protected:
	char name[20];
	int code;
public:
	// void get_num(int c)
	void get_num(int c, char * s)
	{
		code=c;
		strcpy(name,s);
	}
	void put_num()
	{
		cout<<"\ncode : "<<code;
		cout<<"\nName : "<<name;
	}
};
```

```
class account : public virtual person
{
protected:
	float pay;
public:
	void get_pay(float p)
	{
		pay=p;
	}
	void put_pay()
	{
		cout<<"\nPay amount : "<<pay;
	}
};

class admin : public virtual person
{
protected:
	int experience;
public:
	void get_experience(int e)
	{
		experience=e;
	}
	void put_experience()
	{
		cout<<"\nexperience : "<<experience;
	}
};

class master : public account,public admin
```

```
{
public:
        void display()
        {
                put_num();
                put_pay();
                put_experience();
        }
};

int main()
{
        master m1;
   // we are calling it directly form master class
        m1.get_num(77,(char*)"pratham pitty");
        m1.get_pay(79845.50);
        m1.get_experience(3);
        m1.display();
        return 0;
}
```

Output


"C:\Users\prath\OneDrive\Desktop\code block 1\SCOB77_Pratham_pitty_OOP_Assignment_no_6\bin\Debug\SCOB77_Pratham_pitty_OOP_Assignment_no_6.exe"
code : 77
Name : pratham
Pay amount : 79845.5
experience : 3
Process returned 0 (0x0)   execution time : 0.026 s
Press any key to continue.