Class and Object

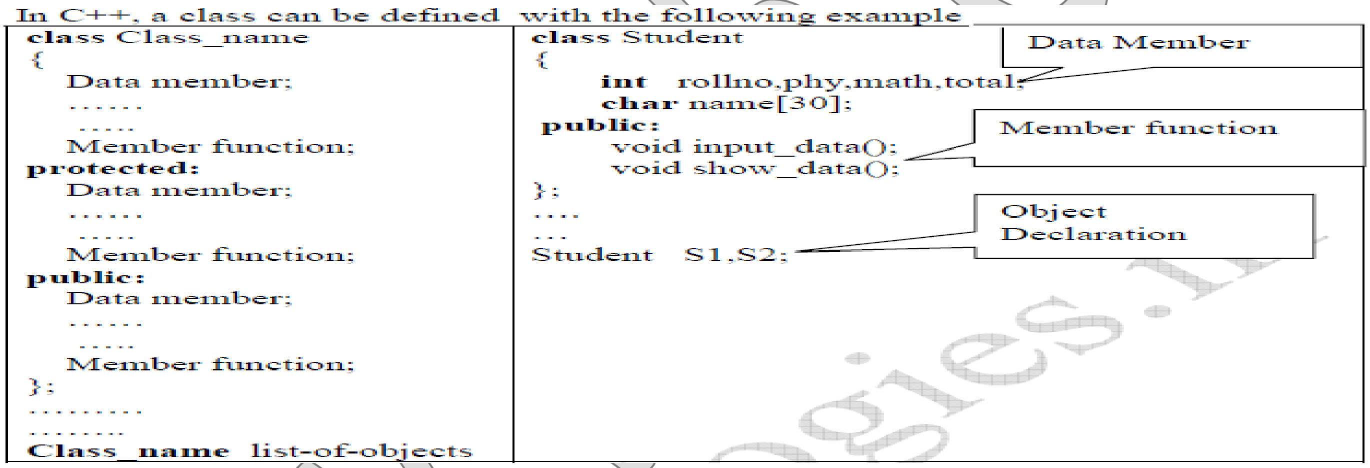
Chapter 4

Important Points(topics)

1. Declare Class
2. Declare Data Members and member function
3. public private and protected (Access modifiers)
4. creating Inner and outer(inside and outside) MF
5. **:: operator(Scope)**
6. Object as local
7. Object as global
8. Call data members and member function as **public**
9. Call data members as **private**
10. **Assign value to DM inside class not allow, using constructor to assign value to DM.**
11. Assign value to data members
12. Static value(Default value) and dynamic value to DM
13. Call Object as local or global
14. **Scope of DM/MF and Class Object in program**
15. **Data Members as Array**
16. **Object Members as Array**
17. **Inline function**
18. **Friend function**
19. Class Object as Function Argument
20. Class Object as Function Return Type
21. Local and Global Class
22. Nested class
23. Nested MF
24. **Calling Static DM and MF**

Important Points (topics) with example

1. Declare Class / Declare Data Members and member function



In C++, a class can be defined with the following example

```
class Class_name
{
   Data member;
   ......
    .....
   Member function;
protected:
   Data member;
   ......
    .....
   Member function;
public:
   Data member;
   ......
    .....
   Member function;
};
.........
........
Class_name  list-of-objects
```

```
class Student
{
      int   rollno,phy,math,total;
      char name[30];
 public:
       void input_data();
       void show_data();
};
....
...
Student   S1,S2;
```

2. Class Reference/Object as global

<class name> <object name>,……; *//outside and inside the class and methods*

Example: A ob1,ob2;

3. public private and protected (Access modifiers)

DM/MF	Own class	Other class	Member function	Inheritance	Other function	Main function
Public	Yes	Yes(By object)	Yes(direct)	Yes(direct)	Yes(By object)	Yes(By object)
Private	Yes	NO	Yes	NO	NO	NO
Protected	Yes	NO	Yes	Yes	NO	NO

4. Scope of Object and class:

Class	Global	This class type is global we can create object anywhere in program
	local	This class type is local, class DM and MF are call inside declaration area
Object	Global	This object type is global we can call object anywhere in program
	local	This object type is local, object call inside declaration area

NOTE : Protected members will be discussed in the inheritance topic

NOTE : if a class defined outside function (main or any other) then it is called global class and it it is defined inside the class then it is called local class.

NOTE : data member (private and public both)are directly accessible through it's member function.

The main purpose of C++ programming is to add object orientation to the C programming language and classes are the central feature of C++ that supports object-oriented programming and are often called user-defined types. **A class is use**d to specify the form of an object and it combines data representation and methods for manipulating that data into one neat package. The data and functions within a class are called members of the class.

C++ Class Definitions: When you define a class, you define a blueprint for a data type. This doesn't actually define any data, but it does define what the class name means, that is, what an object of the class will consist of and what operations can be performed on such an object. A class definition starts with the keyword class followed by the class name; and the class body, enclosed by a pair of curly braces. A class definition must be followed either by a semicolon or a list of declarations. For example, we defined the Box data type using the keyword class as follows:

```
class Box
{
public:
double length; // Length of a box
double breadth; // Breadth of a box
double height; // Height of a box
void InputLBH();
void FindAr();
void DisplayAr();
};
```

The keyword public determines the access attributes of the members of the class that follow it. A public member can be accessed from outside the class anywhere within the scope of the class object. You can also specify the members of a class as private or protected which we will discuss in a sub-section.

Define C++ Objects: A class provides the blueprints for objects, so basically an object is created from a class. We declare objects of a class with exactly the same sort of declaration that we declare variables of basic types. Following statements declare two objects of class Box:

```
void main()
{
Box Box1; // Declare Box1 of type Box
Box Box2; // Declare Box2 of type Box
}
```

Both of the objects Box1 and Box2 will have their own copy of data members.

The public members: A **public** member is accessible from anywhere outside the class but within a program. You can set and get the value of public variables without any member function

The private members: A **private** member variable or function cannot be accessed, or even viewed from outside the class. Only the class and friend functions can access private members.

The protected members: A **protected** member variable or function is very similar to a private member but it provided one additional benefit that they can be accessed in child classes which are called derived classes.

1. There is no need to pass them in the member function.
2. creating Inner and outer(inside and outside) Member function

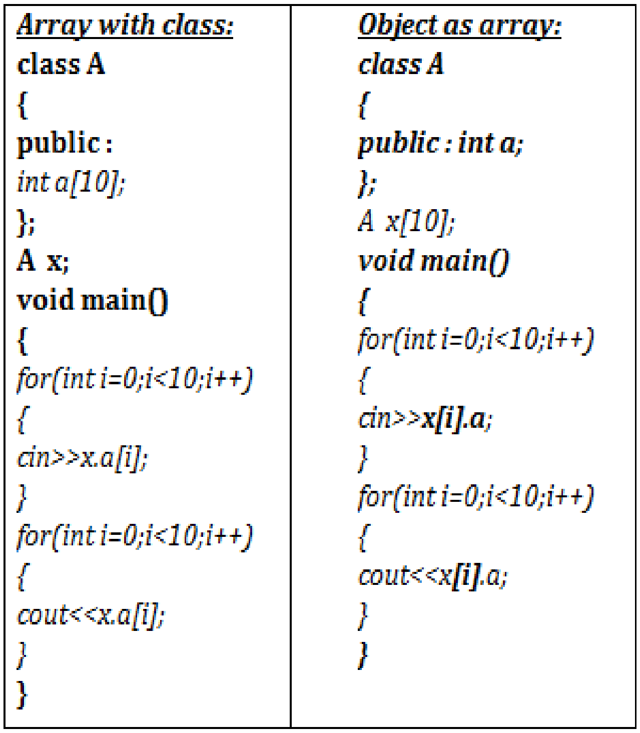
Member Function	
Inner	**Outer**
Declaration **NO**	Declaration **(in side class)**
Definition **(in side class)**	Definition**(Outside class with scope ::)**
Calling **(in side Main Method/other method)**	Calling **(in side Main Method/ other method)**

Example: (Inner)
#include<iostream>
#include<string>
#include<stdio.h>
class employee
{
char name[80];
public:
void putname()//**definition**
{
	Example:(outer)
	#include<iostream>
	#include<string>
	#include<stdio.h>
	class employee
	{
	char name[80];
	public:
	void putname();//**declaration**
	void getname();//**declaration**

```
puts(name);
}
void getname()//definition
{
gets(name);
}
};
void main()
{
employee ted;
ted.getname();//calling
ted.putname();//calling
}
```

```
};
void employee::putname()//definition
{
puts(name);
}
void employee::getname()//definition
{
gets(name);
}
void main()
{
employee ted;
ted.getname();//calling
ted.putname();//calling
}
```

3.

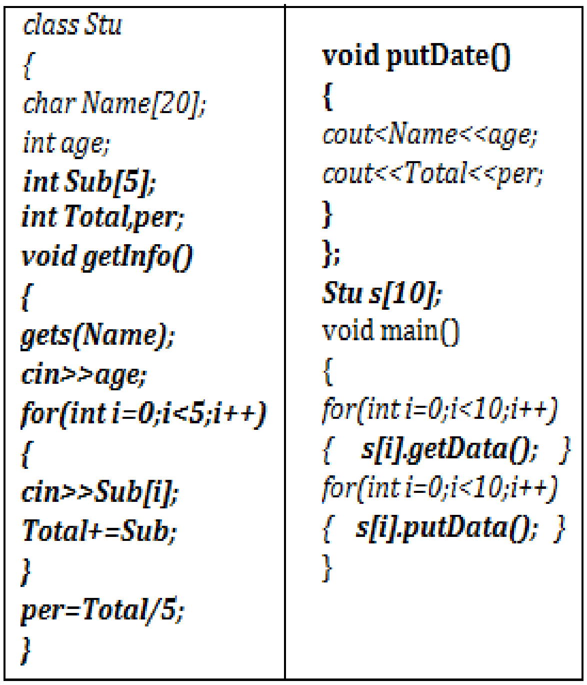


Array with class:

```
class A
{
public :
int a[10];
};
A x;
void main()
{
for(int i=0;i<10;i++)
{
cin>>x.a[i];
}
for(int i=0;i<10;i++)
{
cout<<x.a[i];
}
}
```

Object as array:

```
class A
{
public : int a;
};
A x[10];
void main()
{
for(int i=0;i<10;i++)
{
cin>>x[i].a;
}
for(int i=0;i<10;i++)
{
cout<<x[i].a;
}
}
```



```
class Stu
{
char Name[20];
int age;
int Sub[5];
int Total,per;
void getInfo()
{
gets(Name);
cin>>age;
for(int i=0;i<5;i++)
{
cin>>Sub[i];
Total+=Sub;
}
per=Total/5;
}
```

```
void putDate()
{
cout<Name<<age;
cout<<Total<<per;
}
};
Stu s[10];
void main()
{
for(int i=0;i<10;i++)
{   s[i].getData();  }
for(int i=0;i<10;i++)
{   s[i].putData();  }
}
```

4.

Object and DM both are array:

Global class

```
#include<iostream.h>
class A
{
DM/MF   // as public ,private, protected
};
A  x , y;
void main()
{
///
.
}
```

Local class

```
#include<iostream.h>
void main()
{
        class A
        {
        DM/MF// as public ,private, protected
                };
A  x , y;
}
```

5. **Nested Class:**

```
#include <iostream.h>
class Nest
{
public:
        int a;
        class Display
        {
        private:
        int s;
        public:
        void sum( int a, int b)
        {
         s =a+b;
         }
        void show( )
        {
 cout << "\nSum of a and b is:: " << s;
        }
        };
};
void main()
{
Nest::Display x;
x.a=9;
x.sum(12, 10);
x.show();
}
Sum of a and b is::22
```

6. **Nested Member Function:**

```
#include<iostream.h>
class A
{
        int a,b;
public:
void input()
{
cin>>a>>b;
}
void output()
{
input();
cout<<(a+b);
}
};
void main()
{
A x;
x.output();
}
```

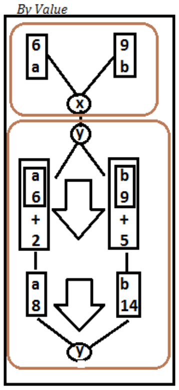
7. **Member function as By Value /By Reference**

```
#include<iostream.h>
class A
{
public: int a,b;
void chang(A y)
{
y.a+=2;
y.b+=5;
cout<< “Y:-> ”<<y.a<< “ : ”<<y.b;
}
};
void main()
{
A x;
x.a=6;
x.b=9;
chang(x);
cout<< “X:-> ”<<x.a<< “ : ”<<x.b;
}
```

Result:

Y: -> 8:14

X:-> 6:9 [No change in original value]



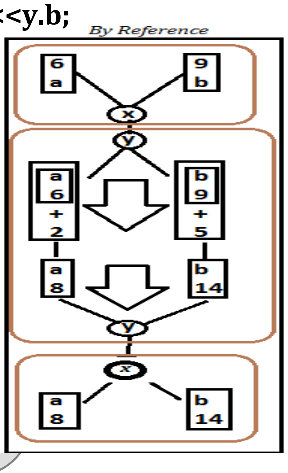
```
#include<iostream.h>
class A
{
public: int a,b;
void chang(A &y)
{
y.a+=2;
y.b+=5;
cout<< “Y:-> ”<<y.a<< “ : ”<<y.b;
} };
void main()
{
A x;
x.a=6;
x.b=9;
cout<< “Before X:-> ”;
cout<<x.a<< “ : ”<<x.b;
chang(x);
cout<< “After X:-> ”;
cout<<x.a<< “ : ”<<x.b;}
```

Result:

Y: -> 8:14

Befor **X:-> 6:9**

After X:-> 8:14 [change in original value]



8. Class Object as return type:

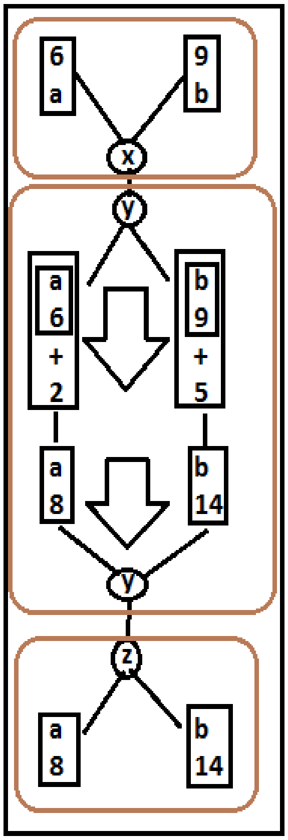
```
#include<iostream.h>
class A
{
public: int a,b;
A chang(A y)
{
y.a+=2;
y.b+=5;
cout<< “Y:-> ”<<y.a<< “ : ”<<y.b;
return y;
}
};
void main()
{
A x,z;
x.a=6;
x.b=9;
cout<< “X:-> ”<<x.a<< “ : ”<<x.b;
z= chang(x);
cout<< “Z:-> ”<<z.a<< “ : ”<<z.b;
}
```

Result:

X:->6:9

Y: -> 8:14

Z:-> 8:14



9. Static member (DM/MF):
   1. static <datatype> VN = value;
   2. call only by class name and scope like (<class name>::VN} for **I/O/P.**

Example:

```
class A
{
public:
static int a;
static void dis();…
.
.
};
void main()
{
A::a=7;
A::dis();
}
```

Solve Question Set 1

Q1. Define Class Student

<u>Private Member:-</u>

Data Member	**Type/Size**
Roll_No	Integer
Name	Character 20 Or String
Class1	Character 20 Or String
Tmarks	Integer
Sub1	Integer
Sub2	Integer
Sub3	Integer

calTmarks() to calculate total marks as Tmarks=Sub1 +Sub2+Sub3

Public Member:

getDate() to **input/accept/get** information of student like Roll_No, Name, Class1,Sub1, Sub2, Sub2.

putData() to display record.

Ans:

```
class Student
{
int Roll_No;
char Name[20],Class1[20];
int Tmarks,Sub1,Sub2,Sub3;
void calTmarks()
{
Tmarks=Sub1 +Sub2+Sub3;
}
public:
void getData()
{
cout<< "Enter Information of Student Roll_No, Name, Class, Marks of Sub1, Sub2, Sub3: "
cin>>Roll_No;
gets(Name);
gets(Class1);
cin>>Sub1>>Sub2>> Sub3;
calTmarks();
}
void putData()
{
cout<< "The Information of Student is"<< Roll_No<< Name<< Class1<<TMarks;
}
};
```

Q2. Define Class Student

Private Member:-

Data Member	**Type/Size**
Roll_No	Integer
Name	Character 20 Or String
Class1	Character 20 Or String
Tmarks	Integer
Sub1	Integer
Sub2	Integer
Sub3	Integer

calTmarks() to calculate total marks as Tmarks=Sub1 +Sub2+Sub3 and return Tmarks

Public Member:

getDate() to **input/accept/get** information of student like Roll_No, Name, Class1,Sub1, Sub2, Sub2.

putData() to display record.

Ans:

```
class Student
{
int Roll_No;
char Name[20],Class1[20];
int Tmarks,Sub1,Sub2,Sub3;
int calTmarks()
{
Tmarks=Sub1 +Sub2+Sub3;
return Tmarks;
}
public:
void getData()
{
cout<< "Enter Information of Student Roll_No, Name, Class, Marks of Sub1, Sub2, Sub3: "
cin>>Roll_No;
gets(Name);
gets(Class1);
cin<< Sub1<<Sub2<< Sub3;
}
void putData()
{
cout<< "The Information of Student is"<< Roll_No<< Name<< Class1<< calTmarks();
}
};
```

Solve Question Set 2

Q1. Define Class Student

Private Member:-

Data Member	**Type/Size**
Roll_No	Integer

Name	Character 20 Or String
Class1	Character 20 Or String
per	integer
Grade	Character

calGrade() to Assign Grade as

Per	Grade
Grater 60	*A*
60 to 50	*B*
50 to 40	*C*
Less 40	*D*

Public Member:

getDate() to **input/accept/get** information of student like Roll_No, Name, Class1 and per

putData() to display record.

Ans:

```
class Student
{
int Roll_No;
char Name[20],Class1[20],Grade
int per
void calGrade(){
if(per>=60) Grade= 'A';
else if(per>=50&&per<60) Grade= 'B';
else if(per>=40&&per<50) Grade= 'C';
else  Grade = 'D'
}
public:
void getData()
{
cout<< "Enter Information of Student Roll_No, Name, Class, Per: "
cin>>Roll_No;
gets(Name);
gets(Class1);
cin>>per;
calGrade();}
void putData()
{
cout<< "The Information of Student is"<< Roll_No<< Name<< Class1<<Grade;}
};
```

Q2. Define Class Student

Private Member:-

Data Member	Type/Size
Roll_No	Integer
Name	Character 20 Or String
Class1	Character 20 Or String
per	integer
Stream	Character 20 Or String

calStream() to Assign Stream as

Per	Stream
Grater 60	*PCM*
60 to 50	*PCB*
50 to 40	*COMM*
Less 40	*ART*

Public Member:

getDate() to **input/accept/get** information of student like Roll_No, Name, Class1 and per

putData() to display record.

Ans:

```
class Student
{
int Roll_No;
char Name[20],Class1[20],Grade
int per
void calSream(){
if(per>=60) strcpy(Stream,"PCM");
else if(per>=50&&per<60) strcpy(Stream,"PCB");
else if(per>=40&&per<50) strcpy(Stream,"COMM");
else  strcpy(Stream,"ART");}
public:
void getData(){
cout<< "Enter Information of Student Roll_No, Name, Class, Per: "
cin>>Roll_No;
gets(Name);
gets(Class1);
cin>>per;
calStream();
}
void putData()
{
cout<< "The Information of Student is"<< Roll_No<< Name<< Class1<<Stream;
}};
```

Q3. Define Class Student

Private Member:-

Data Member	Type/Size
Roll_No	Integer
Name	Character 20 Or String
Class1	Character 20 Or String
per	integer
Stream	Character 20 Or String
Section	Character 20 Or String

calStream() to Assign Section according to

Stream	Section
PCM	**XII Sci A**
PCB	**XII Sci B**

COMM	**XII Com A**
ART	**XII Art A**

Public Member:

getDate() to **input/accept/get** information of student like Roll_No, Name, Class1, **per and Stream**

putData() to display record.

Ans:

```
class Student
{
int Roll_No;
char Name[20],Class1[20],Stream[20],Section[20];
int per
void calSection()
{
if(strcmp(Stream, “PCM”)==0) strcpy(Section,”XII Sci A”);
// if(strcmpi(Stream, “PCM”)==0) ignore  Capital and small Letter
else if(strcmp(Stream, “PCB”)==0)
strcpy(Section,”XII Sci A”);
else if(strcmp(Stream, “COMM”)==0)
strcpy(Section,”XII ComA”);
else  strcpy(Section,”XII Art A”);}
public:
void getData(){
cout<< “Enter Information of Student Roll_No, Name, Class, Per and Stream:”;
cin>>Roll_No;
gets(Name);
gets(Class1);
cin>>per;
gets(Stream);
calSection();}
void putData(){
cout<< “The Information of Student is”<< Roll_No<< Name<< Class1<<per<<Stream<<Section;}
};
```

Rewrite the following program after removing syntactical error(s) if any. Underline each correction(in Paper).

1. include<iostream.h>

```
class FLIGHT
{
Long FlightCode;
Char Description[25];
public
void addInfo()
{
cin>>FlightCode;
 gets(Description);
}
void showInfo()
{
cout<<FlightCode<<”:”<<Description<<endl;
} };
void main( )
{
 FLIGHT F;
addInfo.F();
showInfo.F; }
```

2. #include **“iostream.h”**

```
class MEMBER
{
int Mno;
float Fees;
PUBLIC:
void Register ( )
{
cin>>Mno>>Fees;}
void Display( )
{
cout<<Mno<<" : "<<Fees<<endl;}
};
void main( )
{
MEMBER delete;
Register();
Display().delete;
}
```

3. #include<iostream.h>

```
class MyStudent
{
int StudentId = 101;
char Name[20];
public:
MyStudent (){}
void Register()
{
cin>> StudentId;
gets name
}
void Display(){
cout << StudentId<<”,”<<name<<endline;
};
void main()
{
MyStudent MS
Register.MS;
MS.display;
}
```

4. include <iostream.h>

```
class TRAIN
{
```

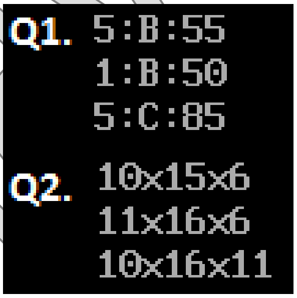
```
long TrainNo;
char Description[25];
public
void Entry ( )
{
cin >>TrainNo; gets(description);
}
Void Display ( )
{
cout<<TrainNo<<“:”<<Description<<endl;
}};
void main( )
{
TRAIN t;
Entry. T( ); Display. T( );
}
```

```
    5.  #include “iostream.h”
    class MEMBER
    {
    int Mno;
    float Fees;
    PUBLIC:
    void Register ( )
    {cin>>Mno>>Fees;}
    void Display( )
    {cout<<Mno<<" : "<<fees<<endl;}
    };
    void main( )
    {
    MEMBER Delete;
    register();
    delete.Display();
    }
```

```
    6.  #include<iostream.h>
#include<stdio.h>
class MyStudent;
{
int StudentId = 101;
char Name[20]= “Rajat Sharma”;
Public
MyStudent (){};
void Register()
{
cin>> StudentId:
gets name}
void Display()
{
 cout << StudentId<<’,’<<name<<endline;
};
void main()
{
MyStudent Ms
Register.MS;
MS.Display(); }
```

Find output

```
    1.  #include<iostream.h>
#include<conio.h>
#include<ctype.h>
class Class
{
int Cno,total;
char section;
public:
Class(int no=1)
{
Cno=no;
section='A';
total=30;
}
void addmission(int c=20)
{
section++;
total+=c;
}
void ClassShow()
{
cout<<Cno<<":"<<section<<":"<<total<<endl;
}} ;
void main()
{
Class C1(5),C2;
C1.addmission(25);
C1.ClassShow();
C2.addmission();
C1.addmission(30);
C2.ClassShow();
C1.ClassShow();
}
```



```
Q1. 5:B:55
    1:B:50
    5:C:85

Q2. 10x15x6
    11x16x6
    10x16x11
```

```
    2.  #include<iostream.h>
struct MyBox
{
int Length, Breadth, Height;
};
void Dimension (MyBox M)
{
cout<<M.Length<<"x"<<M.Breadth<<"x";
cout<<M.Height<<endl;
}
void main()
{
MyBox B1={10,15,5}, B2, B3;
++B1.Height;
Dimension(B1);
B3 = B1;
++B3.Length;
B3.Breadth++;
Dimension(B3);
B2 = B3;
B2.Height+=5;
B2.Length--;
Dimension(B2);
}
```

Define a Class

Q1. Define a class RESORT

Private members:

roomno-	int,
name-	string,
charges-	float,
days-	int,
amount-	float.

compute( ) - To calculate and return amount as days * charges and if the value of days * charges is more than 2100 then as 1.5 * days * charges.

Public members:

enterdetails ( )– to input data and invoke compute( ) function.

display ( ) – to display the details of the customer

Q2. Define a class Travel

Private members:

plancode	of type long
place	characters array
number_of_travellers	of type integer
number_of_buses	of type integer

Public members:

A function newplan( ) which allows user to enter plancode , place and number_of_travellers and also assign the number_of_buses as per the following conditions:

Number Of Travellers	Number Of Buses
less than 20	2
equal to and more than 20 and less than 40	3
equal to and more than 40	4

A function show( ) to display the contents of all the data members on the screen.

Q3. Define a class named Cricket

Private members

Target_scope	int
Overs_bowled	int
Extra_time	int
Penalty	int

cal_panalty() a member function to calculate penalty as follows :

if Extra_time <=10 ,	penalty =1
if Extra_time >10 but <=20,	penalty =2
otherwise,	penalty =5

Public members

- a function extradata() to allow user to enter values for target_score,overs_bowled , extra_time.
- a function dispdata() to follow user to view the contents of all data members.

Q4. Define a class in C++ with following description:

Private Members

Flight number	of type integer
Destination	of type string
Distance	of type float
Fuel	of type float

A member function CALFUEL() to calculate the value of Fuel as per the following criteria

Distance	Fuel
<=1000	500
more than 1000 and <=2000	1100
more than 2000	2200

Public Members

- A function FEEDINFO() to allow user to enter values for Flight Number, Destination, Distance & call function CALFUEL() to calculate the quantity of Fuel
- A function SHOWINFO() to allow user to view the content of all the data members

Q5. Define a class PhoneBill

Private members:

CustomerName	of type character array
PhoneNumber	of type long
No_of_units	of type int
Rent	of type int
Amount	of type float.

calculate( ) This member function should calculate the value of amount as Rent+ cost for the units.Where cost for the units can be calculated according to the following conditions.

No of units	Cost
First 50 calls	Free
Next 100 calls	0.80 @ unit
Next 200 calls	1.00 @ unit
Remaining calls	1.20 @ unit

Public members:

A function accept ( ) which allows user to enter CustomerName, PhoneNumber, No_of_units And Rent and should call function calculate ( ).

A function Display ( ) to display the values of all the data members on the screen.

Q6. Define a class Garments in C++

Private Members:

GCode	of type string
GType	of type string
GSize	of type integer
GFabric	of type string
GPrice	of type float

A function Assign( ) which calculates and assigns the value of GPrice as follows:

For the value of GFabric "COTTON",

GType	GPrice(Rs)
TROUSER	1300
SHIRT	1100

For GFabric other than "COTTON" the above mentioned GPrice gets reduced by 10%

Public Members :

A constructor to assign initial values of GCode, GType and GFabric with the word "NOT ALLOTTED" and GSize and GPrice with “0”.

A function Input( ) to input the values of the data members GCode, GType, GSize and GFabricl and invoke the Assign( ) function.

A function Display( ) which displays the content of all the data members for a Garment

Q7. Define Class Student

Private Member:-

DM	Type/Size
Roll No	Integer
Name	Character 20
Class1	Character 20
per	integer
Stream	Character 20
Section	Character 20

calStream() to Assign Section according to

Stream	Section
PCM	*XII Sci B*
PCB	*XII Sci A*
COMM	*XII Com A*
ART	*XII Art A*

Public Member:

getDate() to input/accept/get information of student like Roll_No, Name, Class1, per and Stream

putData() to display record.

Q8. Define a class employee with the following specification:

Private members

empno	integer type
ename	20 character
basic,hra,da	float
netpay	float

ctotal( ) A function to calculate the total basic+hra+da with float return type.

Public members:

takedata ( ) function to read empno, ename, basic, hra,da and invoke ctotal( ) to calculate total.

showdata ( ) to display all the data members on the screen.

Q9. Declare a class taxpayer with the following specifications:

Private members/Member Function:

int pan - to stores the personal account number

char name[20] - to store the name of a person

float taxincome - to store the total annual taxable income

float tax - to store the tax that is calculated

computetax()-to compute tax for a taxpayer

The tax is calculated according to the following rules:

Total annual income	Rate of taxation
Upto 60000	*0%*
60000 to 150000	*5%*
150000 to 500000	*10%*
above 500000	*15%*

Public Member functions:

inputdata() - to enter the data for a taxpayer

displaydata()- to display the data for a taxpayer

Q10. Define a class CONTEST in C++ with the following description :

Private Data Members

Eventno integer

Description char(30)

Score integer

qualified char

Public Member functions

● A constructor to assign initial values Eventno as 11,Description as ‘‘School level’’, Score as 100, qualified as ‘N’.

●Input() – To take the input for Eventno, description andscore.

● Award (intcutoffscore) – To assign qualified as ‘Y’, if score is
more than the cutoffscore that is passed as argument to the function, else assign qualified as ‘N’.

●Displaydata() – to display all data members.

Q11. Define a class Customer with the following specifications.

Private Members :

Customer_no	integer
Customer_name	char (20)
Qty	integer
Price, TotalPrice, Discount, Netprice	float

Public members:

* A constructer to assign initial values of Customer_no as 111,Customer_name as “Leena”, Quantity as 0 and Price, Discount and Netprice as 0.

*Input( ) – to read data members(Customer_no, Customer_name, Quantity
and Price) call Caldiscount().

* Caldiscount ( ) – To calculate Discount according to TotalPrice and NetPrice is TotalPrice = Price*Qty

TotalPrice	Discount
TotalPrice>=50000	*25% of TotalPrice*
TotalPrice>=25000 and <50000	*15% of TotalPrice*
TotalPrice<250000	*10% of TotalPrice*

Netprice= TotalPrice-Discount

*Show( ) – to display Customer details.

MODEL 1b: Define a class (Using Constructors) 4 Marks

1)Define a class Garments in c++ with following descriptions. (2008 D)

private members :

GCode of type string

GType of type string

Gsize of type intiger

Gfabric of type istring

Gprice of type float

A function **Assign()** which calculate and the value of GPrice as follows.

For the value of GFabric “COTTON” ,

GType	**GPrice(RS)**
TROUSER	1300
SHIRT	1100

For GFabric other than “COTTON”, the above mentioned GPrice gets reduced by 10%

public members:

A constructor to assign initial values of GCode,GType and

GFabric with the a word “NOT ALLOTED”and Gsize and Gprice with 0.

A function Input ()to the values of the data membersGCode,

GType,Gsize and GFabric and invoke the Assign() function.

A function Display () which displays the content of all the data

members for a garment.

```
#include<iostream.h>
#include<string.h>
#include<conio.h>
#include<stdio.h>
class Garments
{ char GCode[21],GType[21];
int Gsize;
char Gfabric[21];
float Gprice;
void Assign( )
{
if(strcmp(strupr(Gfabric),"COTTO
N")==0)
{
if(strcmp(strupr(GType),"TROUSE
R")==0)
Gprice=1300;
if(strcmp(strupr(GType),"SHIRT")
==0)
Gprice=1100;}
else
{if(strcmp(strupr(GType),"TROUS
ER")==0)
Gprice=1300*0.90;
if(strcmp(strupr(GType),"SHIRT")
==0)
Gprice=1100*0.90;}}
public:
Garments( )
{
strcpy(GCode,"NOT ALLOTED");
strcpy(GType,"NOT ALLOTED");
Gsize=0;
strcpy(Gfabric,"NOT ALLOTED");
Gprice=0;
}
void Input( )
{ cout<<"\nEnter the Grament
Code: ";
gets(GCode);
cout<<"\nEnter the Garment Type:
";
gets(GType);
cout<<"\nEnter the Garment Size:
";
cin>>Gsize;
cout<<"\nEnter the Garment
Fabric: ";
gets(Gfabric);
Assign( ); }
void display( )
{ cout<<"\nThe Garment Code:
"<<GCode;
cout<<"\nThe Garment Type:
"<<GType;
cout<<"\nThe Garment Size:
"<<Gsize;
cout<<"\nThe Garment Fabric:
"<<Gfabric;
cout<<"\nThe Garment Price:
"<<Gprice; }};
void main( )
{ Garments G;
G.Input( );
G.display( );}
```

2) Define a class clothing in c++ with the following descriptions : (2008OD)

private members :

code of type string

type of type string

size of type intiger

material of type string

price of type float

A function **calc_price( )**which calculates and assigns the value of

GPrice as follows ;

For the value of material as “COTTON” :

Type price (Rs)

TROUSER 1500.

SHIRT 1200.

for material other than “COTTON”, the above mentioned GPprice

price gets reduced by 25%

public members :

* A constructor to assign initial values of code ,type and material

with the word “NOT ASSIGNED “and size and price with 0.

* A function enter() to input the values of the data members

code, type, size and material and invoke the caclPrice () function.

* A function show which displays the content of all the data

members for a clothing.

```
#include<iostream.h>
#include<string.h>
#include<conio.h>
#include<stdio.h>
class clothing
{ char Code[21],Type[21];
int size;
char material[21];
float price;
void calc_price( )
{
if(strcmp(strupr(material),"COTTON")==0)
{
if(strcmp(strupr(Type),"TROUSER")==0)
price=1500;
if(strcmp(strupr(Type),"SHIRT")==0)
price=1200;
}
else
{
if(strcmp(strupr(Type),"TROUSER")==0)
price=1500*0.75;
if(strcmp(strupr(Type),"SHIRT")==0)
price=1200*0.75;
}
}
public:
clothing( )
{ strcpy(Code,"NOT ALLOTED");
strcpy(Type,"NOT ALLOTED");
size=0;
strcpy(material,"NOT ALLOTED");
price=0;
}
void enter( )
{
cout<<"\nEnter the Cloth Code: ";
gets(Code);
cout<<"\nEnter the Cloth Type: ";
gets(Type);
cout<<"\nEnter the Cloth Size: ";
cin>>size;
cout<<"\nEnter the cloth material: ";
gets(material);
calc_price( );
}
void show( )
{
cout<<"\nThe Cloth Code: "<<Code;
cout<<"\nThe Cloth Type: "<<Type;
cout<<"\nThe Cloth Size: "<<size;
cout<<"\nThe Cloth Material: “ <<material;
cout<<"\nThe Cloth Price: "<<price;}};
void main( )
{
clothing C;
C.enter( );
C.show( );
}
```

3)Define a class Tour in C++ with the description given below. (2007 D)

Private Members:

TCode of type string
No of Adults of type integer
No of Kids of type integer
Kilometers of type integer
TotalFare of type float

Public Members:

A constructor to assign initial values as follows:
TCode with the word “NULL”
No of Adults as 0
No of Kids as 0
Kilometers as 0
TotalFare as 0

A function AssignFare() which calculates and assigns the value of the data member Totalfare as follows
For **each** Adult

XII Computer Chap 4 to6 4 5

Fare (Rs)	***For Kilometers***
500	>=1000
300	<1000 & >=500
200	<500

For **each** Kid the above Fare will be 50% of the Fare mentioned in the above table
For Example:
If Kilometers is 850, Noofadults =2 and NoofKids =3
Then TotalFare should be calculated as
Numof Adults *300+ NoofKids *150
i.e., 2*300+ 3 *150 =1050

A function EnterTour() to input the values of the data members TCode, NoofAdults, NoofKids and Kilometers ; and invoke the AssignFare() function.

A function ShowTour() which displays the content of all
the data members for a Tour.

Ans:

```
#include<conio.h>
#include<stdio.h>
#include<string.h>
#include<iostream.h>
class Tour
{ char TCode[21];
int
NoofAdults,NoofKids,Kilometres;
float TotalFare;
public:
Tour( )
{ strcpy(TCode,"NULL");
NoofAdults=NoofKids=Kilometres=TotalFare=0;
}
void AssignFare( )
{ if(Kilometres>=1000)
TotalFare=NoofAdults*500+NoofKids*250;
else if(Kilometres>=500)
TotalFare=NoofAdults*300+NoofKids*150;
else
TotalFare=NoofAdults*200+NoofKids*100;
}
void EnterTour( )
{ cout<<"\nEnter the Tour Code: ";
gets(TCode);
cout<<"\nEnter the Number of Adults: ";
cin>>NoofAdults;
cout<<"\nEnter the Number of Kids: ";
cin>>NoofKids;
cout<<"\nEnter the Number of Kilometres: ";
cin>>Kilometres;
AssignFare( );
}
void ShowTour( )
{ cout<<"\nThe Tour Code: "<<TCode;
cout<<"\nThe Number of Adults:”<<NoofAdults;
cout<<"\nThe Number of Kids: "<<NoofKids;
cout<<"\nThe Number of
Kilometres: “
<<Kilometres;
cout<<"\n\nThe Total Fare:
"<<TotalFare; }};
void main( )
{ clrscr();
Tour T;
T.EnterTour( );
T.ShowTour( );
getch();
}
```

4) Define a class Travel in C++ with the description given below : (2007 OD)

Private Members:

T_Code of type string
No_ of_ Adults of type integer
No _of _Children of type integer
Distance of type integer
TotalFare of type float

Public Members:

A constructor to assign initial values as follows:
TCode with the word “NULL”
No _of_ Adults as 0
No_ of_Children as 0
Distance as 0
TotalFare as 0

A function AssignFare() which calculates and assigns the value of the data member Totalfare as follows

For **each** Adult

Fare (Rs)	*For Kilometers*
500	>=1000
300	<1000 & >=500
200	<500

For **each** Child the above Fare will be 50% of the Fare mentioned
in the above table
For Example:
If Distance is 750, No_of_adults =3 and No_of_Children =2
Then TotalFare should be calculated as
Num_of _Adults *300+ No_of_Children *150
i.e., 3*300+ 2 *150 =1200

A function EnterTour() to input the values of the data members T_Code, No_of_Adults, No_of_Children and Distance ; and invoke the AssignFare() function.

A function ShowTravel() which displays the content of all the data members for a Travel.

```
#include<conio.h>
#include<stdio.h>
#include<string.h>
#include<iostream.h>
class Travel
{ char T_Code[21];
int
No_of_Adults,No_of_Children,Dista
nce;
float TotalFare;
public:
Travel( )
{ strcpy(T_Code,"NULL");
No_of_Adults=No_of_Children=Dist
ance=
TotalFare=0;
}
void AssignFare( )
{
if(Distance>=1000)
TotalFare=No_of_Adults*500+No_
of_Children*250;
else if(Distance>=500)
TotalFare=No_of_Adults*300+No_
of_Children*150;
else
TotalFare=No_of_Adults*200+No_
of_Children*100;
}
void EnterTravel( )
{ cout<<"\nEnter the Travel Code:
";
gets(T_Code);
cout<<"\nEnter the Number of
Adults: ";
cin>>No_of_Adults;
cout<<"\nEnter the Number of
Children: ";
cin>>No_of_Children;
XII Computer Chap 4 to6 4 6
cout<<"\nEnter the Distance in
Kilometres: ";
cin>>Distance;
AssignFare( );
}
void ShowTravel( )
{ cout<<"\nThe Travel Code: “
<<T_Code;
cout<<"\nThe Number of Adults: “
<<No_of_Adults;
cout<<"\nThe Number of Children:
“<<No_of_Children;
cout<<"\nThe Distance in
Kilometres: "<<Distance;
cout<<"\n\nThe Total Fare:
"<<TotalFare;}};
void main( )
{
clrscr();
Travel T;
T.EnterTravel( );
T.ShowTravel( );
getch();
}
```

5) Define a class Travelplan in C++ with the following descriptions: (2005 D)

Private Members:

Plancode of type long
Place of type character array(string)
Number_of_travellers of type integer
Number_of_buses of type integer

Public Members:

*A constructer to assign initial values of PlanCode as 1001,
Place as “agra”,Number_of_travellers as 5,Number_of_buses as 1
* A function NewPlan() which allows user to enter PlanCode,
Place and Number_of travelers. Also, assign the value of Number_of_buses as per the following conditions:

Number_of_travellers Number_of_buses 1
less than 20 1
Equal to or more than 20 and less than 40 2
Equal to 40 or more than 40 3

* A function ShowPlan() to display the3 content of all the data members on the screen.

Ans:

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
#include<string.h>
class TravelPlan
{ long PlanCode;
char Place[21];
int
Number_of_travellers,Number_of_
buses;
public:
TravelPlan( )
{ PlanCode=1001;
strcpy(Place,"Agra");
Number_of_travellers=5;
Number_of_buses=1;
}
void NewPlan( )
{ cout<<"\nEnter the Plan Code: ";
cin>>PlanCode;
cout<<"\nEnter the Place to
Travel: ";
gets(Place);
cout<<"\nEnter the Number of
Travellers: ";
cin>>Number_of_travellers;
if(Number_of_travellers>=40)
Number_of_buses=3;
else if(Number_of_travellers>=20)
Number_of_buses=2;
else
Number_of_buses=1;
}
void ShowPlan( )
{ cout<<"\nThe Plan Code:
"<<PlanCode;
cout<<"\nThe Place of Travel:
"<<Place;
cout<<"\nNumber of Travellers: “
<<Number_of_travellers;
cout<<"\nNumber of Buses: “
<<Number_of_buses;}};
void main( )
{ clrscr( );
TravelPlan T;
T.NewPlan( );
T.ShowPlan( );
getch();
}
```

6) Define a class Travel in C++ with the following descriptions: (2005 OD)

Private Members:

Travelcode of type long

Place of type character array(string)

Number_of_travellers of type integer

Number_of_buses of type integer

Public Members:

* A constructer to assign initial values of TravelCode as 201,
Place as “Nainital”, Number_of_travellers as 10,
Number_of_buses as 1

* A function NewTravel() which allows user to enter TravelCode, Place and Number_of travelers. Also, assign the
value of Number_of_buses as per the following conditions:

Number_of_travellers Number_of_buses

less than 20 1

Equal to or more than 20 and less than 40 2

Equal to 40 or more than 40 3

* A function ShowTravel() to display the content of all the data
members on the screen.

Ans:

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
#include<string.h>
class Travel
{ long TravelCode;
char Place[21];
int No_of_travellers,No_of_buses;
public:
Travel( )
{ TravelCode=201;
strcpy(Place,"Nainital");
No_of_travellers=5;
No_of_buses=1;
}
void NewTravel( )
{ cout<<"\nEnter the Travel Code:
";
cin>>TravelCode;
cout<<"\nEnter the Place to
Travel: ";
gets(Place);
cout<<"\nEnter the Number of
Travellers: ";
cin>>No_of_travellers;
if(No_of_travellers>=40)
No_of_buses=3;
else if(No_of_travellers>=20)
No_of_buses=2;
else
No_of_buses=1;
}
void ShowTravel( )
{ cout<<"\nThe Plan Code:
"<<TravelCode;
cout<<"\nThe Place of Travel:
"<<Place;
cout<<"\nNumber of Travellers: “
<<No_of_travellers;
cout<<"\nNumber of Buses: “
<<No_of_buses;
}
};
void main( )
{ clrscr( );
Travel T;
T.NewTravel( );
T.ShowTravel( );
getch();
}
```

XII Computer Chap 4 to6 4 7

7) Define a class Play in C++ with the following specifications: (2003 D)

Private members of class **Play**

*Play code integer

*Playtime 25 character

*Duration float

*Noofscenes integer

Public member function of class Play

*A constructer function to initialize Duration as 45 and Noofscenes as

*Newplay() function to values for Playcode and Playtitle.

*Moreinfor() to assign the values of assign the values of Duration and Noofscenes with the of corresponding values passed as parameters to this function.

*Shoplay() function to display all the dataq members on the screen.

Ans:

```
#include<iostream.h>
#include<conio.h>
#include<string.h>
#include<stdio.h>
class Play
{ int Playcode;
char Playtitle[25];
float Duration;
int Noofscenes;
public:
Play( )
{ Duration=45;
Noofscenes=5;
}
void Newplay( )
{ cout<<"\nEnter the Play Code: ";
cin>>Playcode;
cout<<"\nEnter the Play Title: ";
gets(Playtitle);
}
void Moreinfor(float D,int N)
{ Duration = D;
Noofscenes = N;
}
void Showplay( )
{ cout<<"\nThe Play Code : “
<<Playcode;
cout<<"\nThe Play Title : “
<<Playtitle;
cout<<"\nThe Duration :”
<<Duration;
cout<<"\nThe No of
Scenes:"<<Noofscenes;
}
};
void main( )
{ clrscr( );
Play P;
P.Newplay( );
float Dur;
int NS;
cout<<"\nEnter the Duration and
Number of Scenes: ";
cin>>Dur>>NS;
P.Moreinfor(Dur,NS);
P.Showplay( );
getch( );
}
```