CHAPTER-4

Classes and Objects

SHORT ANSWER QUESTIONS

1. **What are the differences between a data type struct and data type class in C++?**

Ans.

struct	class
In C++ struct all members are public by default.	Whereas in class all members are private by default.
structures are declared using the keyword *struct*	classes are declared using the keyword *class*

Example:
`struct S1`
`{`
`  int num;  //default access`
`  //specifier is public`
`  void setNum(int n)`
`  {`
`  //code`
`  }`
`};`
	Example:
	`class C1{`
	`  int num;  //default access`
	`  //specifier is private`
	`  public:`
	`  void setNum(int n)`
	`  {`
	`  //code`
	`  }`
	`};`

2. **Can we use the same function name for a member function of a class and an outside i.e., a non-member function in the same program file? If yes, how are they distinguished? If no, give reasons. Support your answer with examples.**

Ans. Yes. Object of the class is used to distinguish between the member function of a class and a non-member function with same name. Ex-

```
class X{
     public:
       void f()
       {.....}
};
void f()
{.....}
void main(){
  X x;
  x.f();    // member function of the class x
  f();  // non-member function
}
```

3. **When will you make a function inline and why?**

Ans. We will make a function inline when the functions are small that called often. Inline functions run a little faster than the normal functions as the compiler replaces the function call statement with the function code itself and then compiles the entire code. Thus, with inline functions, the compiler does not have to jump to another location to execute the function, and then jump back as the code of the called function is already available to the calling program.

4. **Rewrite the following C++ code after removing the syntax error(s) (if any). Underline each correction.**

```
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();
}
```

Ans.

```
#include<iostream.h>
#include<stdio.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;
       F.AddInfo(); F.ShowInfo();
}
```

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

```
#include[iostream.h]
#include[stdio.h]
class Employee{
      int EmpId=901;
      char EName[20];
   public:
      Employee() {}
      void Joining()
      {
            cin>>EmpId; gets(EName);
      }
      void List()
      {
            cout<<EmpId<<":"
                <<EName<<endl;
      }
}
void main(){
      Employee E;
      Joining.E();
      E.List();
}
```

Ans.

```
#include<iostream.h>
#include<stdio.h>
class Employee{
      int EmpId;
      char EName[20];
   public:
      Employee() {}
```

```
      void Joining()
      {
            cin>>EmpId; gets(EName);
      }
      void List()
      {
            cout<<EmpId<<":"
                <<EName<<endl;
      }
};
void main(){
      Employee E;
      E.Joining();
      E.List();
}
```

6. **Identify the error(s) in the following code fragment:**

```
class X{
      int a b;
      void count(void)
      {
            a++;
      }
   public:
      int x;
      void init(int,int,int);
      void print(void);
};
void X::init(int i,int j,int k){
      a=i;
      b=j;
      x=k;
}
void X::print(void){
     count();
      cout<<"a="<<a;<<"b="
            <<b<<"x="<<x<<"\";
}
void func(void);
      X Ob1;
int main(){
      X Ob2;
      Ob1.init(0,1,2);
      Ob2.init(2,3,4);
      Ob1.print();
      Ob2.print();
      Ob1.count();
      Ob2.count();
}
void func(void)
{
      X Ob3;
      Ob1.init(4,5,6);
      Ob2.init(7,8,9);
      Ob3.init(9,10,11);
      Ob3.a=Ob3.b=Ob3.x;
```

```
      Ob1.count();
      Ob2.count();
      Ob3.count();
      Ob1.print();
      Ob2.print();
      Ob3.print();
}
```

Ans.

```
#include<iostream.h>
#include<stdio.h>
class X   {
      public:
            int a,b;
            void count(void)
            {
                  a++;
            }
            int x;
            void init(int,int,int);
            void print(void);
         };
void X::init(int i,int j,int k)
{
      a=i;
      b=j;
      x=k;
}
void X::print(void)
{     count();
      cout<<"a="<<a<<"b="
          <<b<<"x="<<x<<" ";
}
void func(void);
       X Ob1;
       X Ob2;
int main(){
      Ob1.init(0,1,2);
      Ob2.init(2,3,4);
      Ob1.print();
      Ob2.print();
      Ob1.count();
      Ob2.count();
}
void func(void)
{
      X Ob3;
      Ob1.init(4,5,6);
      Ob2.init(7,8,9);
      Ob3.init(9,10,11);
      Ob3.a=Ob3.b=Ob3.x;
      Ob1.count();
      Ob2.count();
      Ob3.count();
      Ob1.print();
      Ob2.print();
      Ob3.print();
}
```

7. **Identify the error(s) in the following code fragment:**

```
int x=5;
int y=3;
class Outer {
        public:
            int x;
            int a;
            static int s;
            class Inner  {
                  public:
                    void f(int i)
                    {
                       x=i;
                       s=i;
                       y=i;
                       a=i;
                    }
            };
            // Inner defination over
            Inner I1;    //Inner object
            void g(it i)
            {     x=i;
                  y=i;
                  a=i;
                  s=i;
            }
      };      //outer definition over
Outer Ob1;      // outer object
int main()
{     Ob1.I1.f(3);     //statement1
      Ob1.g(8);       //statement2
      return 0;
}
```

What will be the values of ::x, ::y, Outer::x, Outer::a, Outer::s, Inner::a after statement 1 and statement 2 of above code?

Ans.

```
#include<iostream.h>
#include<conio.h>
int x=5;
int y=3;
class Outer {
             public:
                int x;
                int a;
                static int s;
                class Inner
                {
                     public:
                         int a; int x;
                         void f(int i)
                         {
                              x=i;
                              s=i;
                              y=i;
                              a=i;
                          }
                 };
```

```
                Inner I1;
                void g(int i)
                {  x=i;
                   y=i;
                   a=i;
                   s=i;
                }
             };
       int Outer::s;
       Outer Ob1;
       int main()
     {     Ob1.I1.f(3);    //statement1
           Ob1.g(8);       //statement2
           return 0;
     }
```

After statement 1 and statement 2 the values are as following:
::x = 5, ::y = 8, Outer::x = 8, Outer::a = 8, Outer::s =8 , Inner::a = 3

8. **Define a class to represent a book in a library. Include the following members:**
Data Members
Book Number, Book Name, Author, Publisher, Price, No. of copies issued, No. of copies
Member Functions
(i) To assign initial values
(ii) To issue a book after checking for its availability
(iii) To return a book
(iv) To display book information.

Ans.

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
class Library
{
   int BookNo;
   char BName[25];
   char Author[25];
   char Publisher[25];
   float Price;
   int No_of_Copies;
   int No_of_Copies_Issued;
public:
   void initial()
   {
      cout<<endl<<"Enter Book Number: ";
      cin>>BookNo;
      cout<<endl<<"Enter Book Name: ";
      gets(BName);
      cout<<endl<<"Enter Author Name: ";
      gets(Author);
      cout<<endl<<"Enter Publisher Name: ";
      gets(Publisher);
      cout<<endl<<"Enter Price: ";
      cin>>Price;
      cout<<endl<<"Enter Number of copies: ";
      cin>>No_of_Copies;
   }
   void issue_book()
   {
```

```
      cout<<"Enter book details......."<<endl;
      initial();
      if(No_of_Copies>0)
      {
         cout<<"enter How many book you want to issue:";
         cin>>No_of_Copies_Issued;
         if(No_of_Copies>=No_of_Copies_Issued)
         {
            No_of_Copies=No_of_Copies-No_of_Copies_Issued;
            cout<<endl<<"    "<<No_of_Copies_Issued<<" book is issued..";
            display();
         }
         else
         {
            cout<<”Copies_Issued<<" books is not available in stock..";
         }
      }
      else
      {
            cout<<"Book is not available";
      }
}
   void return_book()
   {
      cout<<"enter book detail you want to return...";
      cout<<endl<<"Enter Book Number: ";
      cin>>BookNo;
      cout<<endl<<"Enter Book Name: ";
      gets(BName);
      No_of_Copies=No_of_Copies+No_of_Copies_Issued;
      cout<<endl<<BookNo<<":"<<BName<<"Book is returned......";
   }
   void display()
   {
      cout<<"Book Number: "<<BookNo<<endl;
      cout<<"Book Name: "<<BName<<endl;
      cout<<"Author Name: "<<Author<<endl;
      cout<<"publisher Name: "<<Publisher<<endl;
      cout<<"Price: "<<Price<<endl;
   }
};
void main()
{
      clrscr();
      Library l1;
      int ch;
      cout<<"1->Issue book..."<<endl;
      cout<<"2->Return Book....."<<endl;
      cout<<"Enter your choice...";
      cin>>ch;
      switch(ch)
      {
            case 1:
                  l1.issue_book();
                  break;
            case 2:
```

```
                    l1.return_book();
                    break;
            }
            getch();
}
```

9. **Declare a class to represent fixed-deposit account of 10 customers with the following data members:**
Name of the depositor, Account Number, Time Period (1 or 3 or 5 years), Amount.
The class also contains following member functions:
(a) To initialize data members.
(b) For withdrawal of money (after alf of the time period has passed).
(c) To display the data members.

Ans. Same as Question no. 14 in which Withdraw() function is defined for withdraw money.

10. **Define a class to represent batsmen in a cricket team. Include the following members:**
Data Members:
First name, Last name, Runs made, Number of fours, Number of sixes
Member Functions:
(i) To assign the initial values
(ii) To update runs made (It should simultaneously update fours and sixes, if required).
(iii) To display the batsman's information
Make appropriate assumptions about access labels.

Ans.

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
class Batsman{
      char F_Name[30];
      char L_Name[30];
      int Runs_made,fours,sixes;
      public:
            void initial(){
                  cout<<endl<<"Enter First Name: ";
                  gets(F_Name);
                  cout<<endl<<"Enter Last Name: ";
                  gets(L_Name);
                  cout<<endl<<"Enter The Runs Made: ";
                  cin>>Runs_made;
                  cout<<endl<<"Enter how many fours: ";
                  cin>>fours;
                  cout<<endl<<"Enter how many sixes: ";
                  cin>>sixes;
            }
            void update(){
                  int new_run,new_four,new_sixes,cal_four,cal_six;
                  cout<<endl<<"Enter new runs Made: ";
                  cin>>new_run;
                  cout<<endl<<"Enter new fours Made: ";
                  cin>>new_four;
                  cout<<endl<<"Enter new sixes Made: ";
                  cin>>new_sixes;
                  fours=fours+new_four;
                  sixes=sixes+new_sixes;
                  cal_four=fours*4;
                  cal_six=sixes*6;
                  Runs_made=Runs_made+new_run+cal_four+cal_six;
                  display();
                  cout<<"Total Runs Made: "<<Runs_made<<endl;
```

```
                cout<<"Number of fours: "<<fours<<endl;
                cout<<"Number of sixes: "<<sixes<<endl;
          }
          void display(){
               cout<<".....Batsman's information....."<<endl;
                cout<<"Name: "<<F_Name<<" "<<L_Name<<endl;
          }
};
void main(){
     clrscr();
      Batsman b1;
      b1.initial();
      b1.update();
     getch();
}
```

11. **Define a class to represent bowlers in a cricket team. Include the following members:**
Data Members:
First name, Last name, Overs bowled, Number of Maiden overs, Runs given, Wickets taken.
Member Functions:
(i) To assign the initial values, (ii) To update the information, (iii) To display the bowler's information
Make appropriate assumptions about access specifiers.

Ans.

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
class Bowlers{
      char F_Name[30];
      char L_Name[30];
      int Overs_bowled,Maiden_overs,Runs_given,Wickets;
      public:
            void initial(){
                  cout<<endl<<"Enter First Name: ";
                  gets(F_Name);
                  cout<<endl<<"Enter Last Name: ";
                  gets(L_Name);
                  cout<<endl<<"Enter The Overs bowled: ";
                  cin>>Overs_bowled;
                  cout<<endl<<"Enter how many overs maden: ";
                  cin>>Maiden_overs;
                  cout<<endl<<"Enter how many runs given: ";
                  cin>>Runs_given;
                  cout<<endl<<"Enter how many wickets taken: ";
                  cin>>Wickets;
            }
            void update(){
                  int
new_over_bolwed,new_maiden_overs,new_runs_given,new_wickets;
                  cout<<endl<<"Enter new overs bowled: ";
                  cin>>new_over_bolwed;
                  cout<<endl<<"Enter new madden overs: ";
                  cin>>new_maden_overs;
                  cout<<endl<<"Enter new runs given: ";
                  cin>>new_runs_given;
                  cout<<endl<<"Enter new wickets taken: ";
                  cin>>new_wickets;
                  Overs_bowled=Overs_bowled+new_over_bolwed;
                  Maiden_overs=Maiden_overs+new_maiden_overs;
```

```
                    Runs_given=Runs_given+new_runs_given;
                    Wickets=Wickets+new_wickets;
                    display();
                    cout<<"Total overs bowled: "<<Overs_bowled<<endl;
                    cout<<"Total maidden overs: "<<Maiden_overs<<endl;
                    cout<<"Total runs given: "<<Runs_given<<endl;
                    cout<<"Total wickets taken: "<<Wickets<<endl;
               }
               void display(){
                    cout<<".....Bolwer's information....."<<endl;
                    cout<<"Name: "<<F_Name<<" "<<L_Name<<endl;
               }
};
void main(){
     clrscr();
     Bowlers b1;
     b1.initial();
     b1.update();
     getch();
}
```

12. **Define a class student with the following specifications:**

private members of class student

admno	**integer**
sname	**20 characters**
eng, math, science	**float**
total	**float**
ctotal()	**A function to calculate eng + math + science with float return type**

public member functions of class student

Takedata() function to accept values for admno, sname, eng, math, science and ivoke ctotal() to calculate total.

Showdata() function to display all the data members on the screen.

Ans.

```
class student{
   private:
      int admno;
      char sname[20];
      float eng,math,science;
      float total;
      float ctotal(){
           return eng+math+science;
      }
   public:
      void Takedata(){
           cout<<"Enter admission number: ";
           cin>> admno;
           cout<<endl<<"Enter student name: " ;
           gets(sname);
           cout<< "Enter marks in english:";
           cin>>eng;
           cout<< "Enter marks in math:";
           cin>>math;
           cout<< "Enter marks in science:";
           cin>>science;
           total=ctotal();
```

```
        }
        void Showdata(){
                cout<<endl<<".........Student information...."<<endl;
                cout<<"Admission number "<<admno;
                cout<<"\nStudent name "<<sname;
                cout<<"\nEnglish "<<eng;
                cout<<"\nMath "<<math;
                cout<<"\nScience "<<science;
                cout<<"\nTotal "<<total;
        }
};
int main(){
        clrscr();
        student obj ;
        obj.Takedata();
        obj.Showdata();
        getch();
        return 0;
}
```

13(a). **Considering the following specifications:**

Structure name	Data	Type	Size
Name	first	array of characters	60
	mid	array of characters	40
	last	array of characters	60
Phone	area	array of characters	4
	exch	array of characters	4
	numb	array of characters	6

Class name	Data	Type
P_rec	name	Name
	phone	Phone

with member functions constructors and display_rec.
(i) Declare structures in C++ for Name and Phone.
(ii) Declare a class for P_rec.
(iii) Define the constructor (outside the class P_rec) that gathers information from the user for the above two structures Name and Phone.
(iv) Define the display_rec (outside the class P_rec) that shows the current values.

Ans.

```
#include<iostream.h>
#include<stdio.h>
#include<conio.h>
struct Name
{
        char first[40];
        char mid[40];
        char last[60];
};
struct Phone
{
        char area[4];
        char exch[4];
        char numb[6];
};
class P_rec
{
```

```
      Name name;
      Phone phone;
      p_rec();
      void display_rec();
};
P_rec()
{
      first="abc";
      mid="aaa";
      last="jjj";
      area=1234;
      exch=7546;
      numb=789456;
}
void display_rec()
{
      cout<<first<<mid<<last<<area<<exch<<numb;
}
void main()
{
      clrscr();
      P_rec p;
      p.display_rec();
      getch();
}
```

13(b). **Consider the following class declaration and answer the questions below:**

```
class SmallObj
{
     private:
      int some,more;
      void err_1() {cout<<"error";}
     public:
      void Xdata(int d) {some=d;more=d++;  }
      void Ydata() {cout<<some<<" "<<more; }
};
```

(i) Write the name that specifies the above class.
(ii) Write the data of the class with their access scope.
(iii) Write all member functions of the class along with their access scope.
(iv) Indicate the member function of the SmallObj that sets data.

Ans.

(i) SmallObj
(ii) private int some, more;
(iii) private void err_1(){cout<<"error";}
public void Xdata(int d) {some=d;more=d++; }
public void Ydata() {cout<<some<<" "<<more; }
(iv) public void Xdata(int d) {some=d;more=d++; }

14. **Declare a class to represent bank account of 10 customers with the following data members.**
Name of the depositor, Account number, Type of account (S for Savings and C for Current), Balance amount.
The class also contains member functions to do the following:
(i) To initialize data members
(ii) To deposit money
(iii) To withdraw money after checking the balance (minimum balance in Rs. 1000)
(iv) To display the data members

Ans.

```
#include<iostream.h>
#include<conio.h>
```

```
#include<stdio.h>
class Account
{
      char D_Name[30];
      float Amount;
      char acc_type[2];
      public:
            long Acc_No;
            void initial()
            {
                  cout<<endl<<"Enter Depositers Name: ";
                  gets(D_Name);
                  cout<<endl<<"Enter Account Number: ";
                  cin>>Acc_No;
                  cout<<endl<<"Enter Type of account (S for Saving and C for
Current): ";
                  gets(acc_type);
                  cout<<endl<<"Enter Ammount: ";
                  cin>>Amount;
            }
            void Deposit()
            {
                  float dip;
                  cout<<"Enter Money to deposit:";
                  cin>>dip;
                  display();
                  Amount=Amount+dip;
                  cout<<"After deposit total amount is: "<<Amount;
            }
            void Withdraw()
            {
                  float wid;
                  cout<<endl<<"Entre money to withdraw:";
                  cin>>wid;
                  if(Amount>=1000)
                  {
                        display();
                        Amount=Amount-wid;
                        cout<<"After withdraw the amount is:"<<Amount;
                  }
                  else
                  {
                        cout<<"....you can not withdraw money.....";
                  }
            }
            void display()
            {
                  cout<<"Depositers Name: "<<D_Name<<endl;
                  cout<<"Account Number: "<<Acc_No<<endl;
                  cout<<"Account Type: "<<acc_type<<endl;
                  cout<<"Amount: "<<Amount<<endl;
            }
            long getaccno()
            {
                  return Acc_No;
            }
```

```
};
void main()
{
      clrscr();
      Account A1[10];
      long a;
      int i,flag=0;
      int ch;
      for(i=0;i<10;i++)
      {
            cout<<endl<<"Enter information for Depositer "<<i+1<<":"<<endl;
            A1[i].initial();
      }
      for(i=0;i<10;i++)
      {
            cout<<endl<<"Depositer- "<<i+1<<":"<<endl;
            A1[i].display();
      }
      cout<<"************************************"<<endl;
      cout<<"  1->deposit..."<<endl;
      cout<<"  2->withdraw.."<<endl;
      cout<<"Enter your choice:";
      cin>>ch;
      switch(ch)
      {
            case 1:
                  cout<<endl<<"Enter account number for which diposit
money:";
                  cin>>a;
                  for(i=0;i<10;i++)
                  {
                        if(A1[i].getaccno()==a)
                        {
                              flag=1;
                              break;
                        }
                        else
                        {
                              flag=0;
                        }
                  }
                        if(flag==0)
                        {
                              cout<<"Account number not found.....";
                        }
                  else
                  {
                        A1[i].Deposit();
                  }
                  break;
            case 2:
                  cout<<endl<<"Enter account number for which withdraw
money:";
                  cin>>a;
                  for(i=0;i<10;i++)
                  {
```

```
                        if(A1[i].getaccno()==a)
                        {
                              flag=1;
                              break;
                        }
                        else
                        {
                              flag=0;
                        }
                  }
                        if(flag==0)
                        {
                              cout<<"Account number not found.....";
                        }
                  else
                  {
                        A1[i].Withdraw();
                  }
                  break;
      }
      getch();
}
```

15. **Define a class worker with the following specification:**

Private members of class worker

wname	**25 characters**
hrwrk	**float (hors worked and wagerate per hour)**
totwage	**float(hrwrk*wgrate)**
calcwg	**A fuction to find hrerk* wgrate with float return type**

Public members of class worker

in_data()	**a function to accept values for wno, wname, hrwrk, wgrate and invoke calcwg() to calculate totwage.**
out_data()	**a function to display all the data members on the screen you should give definations of functions.**

Ans.

```
#include<iostream.h>
#include<stdio.h>
#include<conio.h>

class worker
{
      int wno;
      char wname[25];
      float hewrk,wgrate;
      float totwage;
      float calcwg()
      {

            totwage = hewrk*wgrate;
```

```
            return totwage;
      }

public:
      void in_data();
      void out_data();
};

void worker::in_data()
{
      cout<<"Enter worker number:";
      cin>>wno;
      cout<<"enter worker name:";
      gets(wname);
      cout<<"Enter hours worked: ";
      cin>>hewrk;
      cout<<"Enter wage rate per hour:";
      cin>>wgrate;
      calcwg();
}



void worker::out_data()
{
      cout<<"......Worker Information........"<<endl;
      cout<<"Worker number:"<<wno<<endl;
      cout<<" Worker name:"<<wname<<endl;
      cout<<" Hours worked:"<< hewrk<<endl;
      cout<<" Wage rate per hour:"<< wgrate<<endl;
      cout<<" Total wage:"<<totwage<<endl;
}

int main()
{
      worker obj;
      obj.in_data();
      obj.out_data();

      getch();
      return 0;
}
```

16. **Define a class Teacher with the following specification:**

private members:

name	**20 characters**
subject	**10 characters**
Basic,DA,HRA	**float**
salary	**float**

Calculate() function computes the salary and returns it.
Salary is sum of Basic, DA and HRA

public members:

Readdata() function accepts the data values and invoke the calculate function
Displaydata() function prints the data on the screen.

Ans.

```
#include<iostream.h>
#include<stdio.h>
#include<conio.h>

class Teacher
{
      char name[20];
      char subject[10];
      float Basic,DA,HRA;
      float salary;
      float Calculate()
      {

            salary=Basic+DA+HRA;
            return salary;
      }

public:
      void Readdata();
      void Displaydata();
};

void Teacher::Readdata()
{
      cout<<endl<<"Enter name:";
      gets(name);
      cout<<"Enter subject:";
      gets(subject);
      cout<<"Enter Basic :";
      cin>>Basic;
      cout<<"Enter DA :";
      cin>>DA;
      cout<<"Enter HRA :";
      cin>>HRA;
      Calculate();
}



void Teacher::Displaydata()
{
      cout<<"......Teacher Details........"<<endl;
      cout<<"Name:"<<name<<endl;
      cout<<" Subject:"<<subject<<endl;
      cout<<" Basic:"<<Basic<<endl;
      cout<<" DA:"<<DA<<endl;
      cout<<" HRA:"<<HRA<<endl;
      cout<<" Salary:"<<salary<<endl;
}

int main()
{
      Teacher obj;
      obj.Readdata();
      obj.Displaydata();
```

```
      getch();
      return 0;
}
```

17. **Define a class Student with the following specification:**

private members:

roll_no	**integer**
name	**20 characters**
class	**8 characters**
marks[5]	**integer**
percentage	**float**

Calculate() function that calculates overall percentage of marks and returns the percentage of marks.

public members:

Readmarks() a function that reads marks and invokes the calculate functio
Displaymarks() a function that prints the marks.

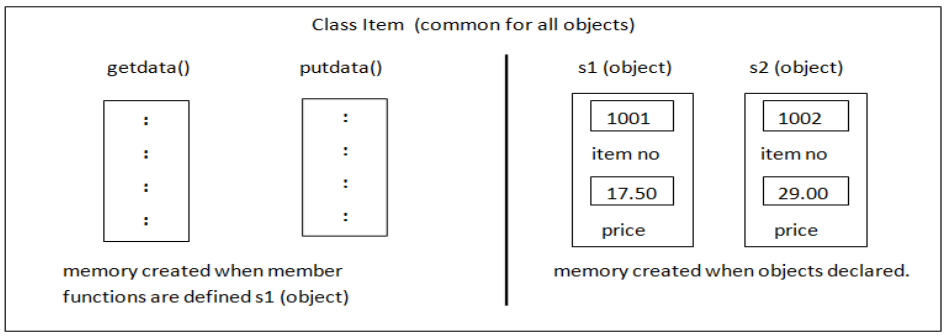
Ans.

```
#include<iostream.h>
#include<stdio.h>
#include<conio.h>
class Student{
       int roll_no;
       char name[20];
       char Class[8];
       int marks[5];
       float percentage;
       float Calculate(){

          percentage = (marks[0]+marks[1]+marks[2]+marks[3]+marks[4])/5;
         return percentage;
       }
public:
       void Readmarks();
       void Displaymarks();
};
void Student::Readmarks(){
       cout<<endl<<"Enter roll number: ";
       cin>>roll_no;
       cout<<endl<<"Enter name:";
       gets(name);
       cout<<"Enter marks in ";
       for(int i=0;i<5;i++)
       {
              cout<<endl<<"Subject "<<i+1<<":";

              cin>>marks[i];
       };
       Calculate();
}
void Student::Displaymarks(){
       cout<<"......Student Marksheet........";
       cout<<endl<<"Roll number:"<<roll_no<<endl;
       cout<<" Name:"<<name<<endl;
       cout<<" Marks in subject-1:"<< marks[0]<<endl;
```

```
        cout<<" Marks in subject-2:"<< marks[1]<<endl;
        cout<<" Marks in subject-3:"<<marks[2]<<endl;
        cout<<" Marks in subject-4:"<<marks[3]<<endl;
        cout<<" Marks in subject-5:"<< marks[4]<<endl;
        cout<<" Percentage:"<<percentage<<endl;
}
int main(){
        Student obj;
        obj.Readmarks();
        obj.Displaymarks();

        getch();
        return 0;
}
```

18. **Write a short note on memory allocation of class and its objects.**

Ans. When a class is defined, memory is allocated for its member functions and they are stored in the memory. When an object is created, separate memory space is allocated for its data members. All objects work with the one copy of member function shared by all.



19. **Write a program that invokes a function newdate() to return a object of date type. The function newdate() takes two parameters: an object olddate of date type and number of days (int) to calculate the newdate as olddate + number of days and returns the newdate.**

Ans.

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>

static int days_in_month[] = { 0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31,
30, 31 };
int day, month, year;

unsigned short day_counter;

int is_leap(int y){
    return ((y % 4 == 0 && y % 100 != 0) || y % 400 == 0);
}

class date{
     public:
     //int d,m,y;

           void olddate(int d, int m, int y);
           void next_day();
           void newdate(date set_date,int days);
};
```

```
void date::olddate(int d, int m, int y){
    m < 1 ? m = 1 : 0;
    m > 12 ? m = 12 : 0;
    d < 1 ? d = 1 : 0;
    d > days_in_month[m] ? d = days_in_month[m] : 0;
    if (is_leap(y)){
        days_in_month[2] = 29;
    }
    else {
        days_in_month[2] = 28;
    }
    day = d;
    month = m;
    year = y;
 }
void date::next_day(){
    day += 1; day_counter++;
    if (day > days_in_month[month]) {
        day = 1;
        month += 1;
    if (month > 12) {
        month = 1;
        year += 1;
        if (is_leap(year)) {
            days_in_month[2] = 29;
        } else {
            days_in_month[2] = 28;
        }
    }
}
}

void date::newdate(date olddate,int x){
    int i;
    for (i=0;i<x;i++) next_day();
}
int main(){
     clrscr();
     date d1;
     d1.olddate(22,2,1980);
     d1.newdate(d1,62);
     day_counter = 0;
     cout<<"day:"<<day<<" month:"<<month<<" year:"<<year;
     getch();
}
```

20. **What are static members of a class? When and how are they useful?**

Ans. A class can have static data members as well as static member functions.
The static data members are the class variables that are common for all the objects of the class. Only one copy of static data members is maintained which is shared by all the objects of the class. They are visible only within the class but their lifetime is the entire program. They can be accessed by all the member functions.
A member function that accesses only static data members of a class is static member functions. It cannot access other data members but static members.
The static data members are useful when some data values are to be shared across objects of the same class.

LONG QUESTION ANSWERS