1.Write a C++ program to demonstrate the use of Friend function in class.

2.Create two classes' dist1 (meters, centimeters) and dist2 (feet, inches). Accept two

distances from the user, one in meters and centimeters and the other in feet and inches. Find

the sum and difference of the two distances. Display the result in both (meters and centimeters)as well as feet and inches (use

friend function).

Code:

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

class DM
{
public:
double meter,centimeter;
};

class DB
{
public:
double feet,inches; friend
void add(DM,DB);
};

void add(DM dm,DB db)
```

```
{
double d1,d2; cout<<"\nEnter the distance in meter
and entimeter:";
cin>>dm.meter>>dm.centimeter;
cout<<"\nEnter the distance in feet and inches:";
cin>>db.feet>>db.inches;
d1=dm.meter+(db.feet)/3.281;
d2=dm.centimeter+(db.inches)*2.54; cout<<"\nMeter +
Feet = "<<d1<<" meter"; cout<<"\nCentimeter + inches
= "<<d2<<" cemtimeter";
}
void main()
{
clrscr();
DM  dm;
DB db;
add(dm,db);
getch();
}
```

3.Write a C++ program to create a class Student which contains data members as

Roll_Number, Stud_Name, Marks in five subjects. Write member functions to

accept Student information. Display all details of student along with a percentage

and class obtained depending on percentage. (Use array of objects)

```
#include <iostream> using
        namespace std;
        class Student
        {
                private:
                        intRoll_Number, Marks[5];
                        charStud_Name[80], Class;
        public:
                        void accept();
                        void display();
        };
        void Student :: accept()
        {
                cout<< " Accept Student Roll Number "<<endl;
        cin>>Roll_Number;            cout<< " Accept Student
        Name"<<endl;            cin>>Stud_Name;
        cout<< " Accept Student Marks "<<endl;            float
        total = 0;
                for(inti = 0; i< 5; i++)
                {
                        cout<< " \t Subject "<<i<<endl;
                        cin>> Marks[i];
        total = total + Marks[i];
                }
                total = total / 5; // calculate Percentage
                if(total < 60)
        Class='B';
                if((total >= 60)&&(total < 70))
                        Class='A';
        if(total >= 70)
                        Class='O';
        }
```

```
void Student :: display()
{
          cout<< " " <<Roll_Number<< "\t" <<Stud_Name<< "\t ";
for(inti = 0; i< 5; i++)                    cout<< Marks[i] << "\t";
          cout<< Class << "\n";
}

int main()
{
          int n, i;
          Student p[20];
          cout<< " * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * " <<endl;          cout<< " \t How many Student data to
  enter " <<endl;          cout<< " * * * * * * * * * * * * * * * * * * *
  * * * * * * * * *
* * * * " <<endl;          cin>> n;          cout<< " * * * * * * * * * *
  * * * * * * * * * * * * * * * * * *
* * * * " <<endl;
          cout<< " \t Enter Student Details " <<endl;
          for ( i = 0 ; i< n ; i ++)
          {
                    cout<<" * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " <<endl;
                    cout<<" \t Enter Student Details of Student := "
<<i<<endl;
                    cout<<" * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " <<endl;
                    p[i].accept();
          }
          cout<< " * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * " <<endl;
```

```
            cout<< " Display Student Details " <<endl;            cout<<
        " * * * * * * * * * * * * * * * * * * * * * * * * * * * *
        * * * * " <<endl;
            cout<< " RNO \tSName \tSub1 \tSub 2 \tSub 3 \tSub4 \tSub
        5";   cout<< "\t Class"<<endl;
            cout<< " * * * * * * * * * * * * * * * * * * * * * * * * * * * *
        * * * * " <<endl;
            for ( i = 0 ; i< n ; i ++)
        p[i].display();            cout<< " * * * * * *
        * * * * * * * * * * * * * * * * * * * * *
        * * * * " <<endl;
            return 0;
        }
```

4.Write a C++ program to print area of circle, square and rectangle using inline function.

Code:

```
#include<iostream.h>
#include<conio.h>
using namespace std;
int area(int); int
area(int,int); float
area(float); float
area(float,float); int
main()
{
ints,l,b; floatr,bs,ht; cout<<"Enter side of a
square:"; cin>>s; cout<<"Enter length and
breadth of rectangle:"; cin>>l>>b;
cout<<"Enter radius of circle:"; cin>>r;
```

```
cout<<"Enter base and height of triangle:";
cin>>bs>>ht; cout<<"Area of square
is"<<area(s); cout<<"\nArea of rectangle is
"<<area(l,b);    cout<<"\nArea of circle is
"<<area(r);    cout<<"\nArea of triangle is
"<<area(bs,ht);
}
int area(int s)
{
   return(s*s);
}
int area(intl,int b)
{
   return(l*b);
}
float area(float r)
{
   return(3.14*r*r);
}
float area(float bs,floatht)
{
  return((bs*ht)/2);
}
```

5. Write a C++ program to create a class Person that contains data members as

Person_Name, City, Mob_No. Write a C++ program to perform following functions:

a. To accept and display Person information

b. To search the mobile number of a given person

c. To search the Person details of a given mobile number

(Use Function Overloading) [20]

Code

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

```
#include<string.h>
class person
 {      char
name[10],city[10],mno[12];
public:
    void accept()
    {
            cout<<"\n Enter name : ";
cin>>name;                cout<<"\n
Enter city : ";
            cin>>city;
cout<<"\n Enter mob no : ";
cin>>mno;
    }

    void display()
    {
            cout<<"\n Name of person = "<<name;
            cout<<"\n city = "<<city<<"\n mobile no = "<<mno;
            cout<<"\n==================================
==";
    }

    int display(char a[])
    {
            if(strcmp(name,a)==0)
{ cout<<"mno="<<mno;
              return 1;
             }
              return 0;
    }


    int display(char mbno[],int no)
```

```
    {

            if(strcmp(mno,mbno)==0)
             {
                        display();
return 1;
             }
               return 0;
}

};

void main()
 {
   intn,i,cnt=0,ans;
charsname[20],mbno[12];
   personob[20];     clrscr();
cout<<"\n Enter no of persons : ";
cin>>n;
   for(i=0;i<n;i++)
    {
            ob[i].accept();
    }

   for(i=0;i<n;i++)
    {
            ob[i].display();
    }

   cout<<"\n Enter name of person to be search : ";
cin>>sname;
   for(i=0;i<n;i++)
    {
            ans=ob[i].display(sname);
```

```
            if(ans==1)
cnt++;
```

```
    }
     if(cnt==0)
     cout<<"\n person not found\n ";

  cout<<"\n Enter mobile no to be search : ";
cin>>mbno;
              cnt=0,ans=0;
   for(i=0;i<n;i++)
    {
               ans=ob[i].display(mbno,1);
               if(ans==1)
cnt++;
    }
      if(cnt==0)
      cout<<"\n person not found\n";


  getch();
}
```

Q6. Write a program in C++ that copies one file to another file.

```
#include<iostream>
#include<fstream>
usingnamespacestd;

intmain()
{ ifstream fs;
          ofstreamft;
          stringstr;
          charsourcefile[50], destinationfile[50];
          cout<<"Enter Source File with Extension: ";
```

```
        gets(sourcefile);

        fs.open(sourcefile);

        if (!fs)
        { cout<<"Error in Opening Source File...!!!";
                exit(1);
        } cout<<"Enter Destination File with Extension:

        "; gets(destinationfile); ft.open(destinationfile);

        if (!ft)
        { cout<<"Error in Opening Destination File...!!!";
                fs.close();
                exit(2);
        }

        if (fs &&ft)
        {
                while (getline(fs, str))
                { ft<<str<<"\n";
                }

                cout<<"\n\n Source File Date Successfully Copied to
Destination File...!!!";

        }
        else
```

```
        { cout<<"File Cannot Open...!!!";
        } cout<<"\n\n Open Destination File and

        Check!!!\n";

        fs.close();
        ft.close();
}
```

7. To calculate the area of circle, rectangle and triangle using function overloading.

Code:

```
/* C++ program to find Area using Function Overloading  */


#include<iostream>
using namespace std;
int area(int);
```

```
ints,l,b; floatr,bs,ht; cout<<"Enter side of a
square:"; cin>>s; cout<<"Enter length and
breadth of rectangle:"; cin>>l>>b;
cout<<"Enter radius of circle:"; cin>>r;
cout<<"Enter base and height of triangle:";
```

```
return(s*s);
}
int area(intl,int b)
{
return(l*b);
```

```
}

float area(float r)

{
```

8.Create a class student containing data members: a. Roll_no b. name c. marks1, marks2,marks3 Write necessary member functions:

to accept details of all students
to display details of one student

to display details of all students

(Use Function overloading).

Code:

```
#include<iostream.h>
#include<conio.h>
class stud
{
```

```
public: int rno;
char name[20];
int  m1,m2,m3;
public:
void details()
{ cout<<“\nROLL_NO NAME MARKS THREE SUBJECT
\n”; cin>>rno>>name>>m1>>m2>>m3;
}
void details(int nm)
{ cout<<“\nDetails of ONE student
:\n”;
cout<<“\nROLL_NO\tNAME\tMARK1\tMARK2\tMARK3 \n”;
cout<<“=======================================\n” ;
cout<<nm<<“\t”<<name<<“\t”<<m1<<“\t”<<m2<<“\t”<<m3<<endl<<
“\n\n”;
}
void details(char*)
{
cout<<rno<<“\t”<<name<<“\t”<<m1<<“\t”<<m2<<“\t”<<m3<<endl;
}
}; void
main() {
int rn,i,n; stud s[20]; clrscr();
cout<<“How many student inform: \n”;
cin>>n; for(i=0;i<n;i++)
{
s[i].details();
} cout<<“Enter the roll number:\n
“; cin>>rn; for(i=0;i<n;i++)
{
if(rn==s[i].rno)
```

```
{
s[i].details(rn);
}
}
cout<<“Details of all student: \n”;
cout<<“ROLL_NO\tNAME\tMARK1\tMARK2\tMARK3 \n\n”;
cout<<“=====================================\n” ;
for(i=0;i<n;i++)
{
s[i].details(s[i].name);
}
getch();
}
```

10. Write a C++ program to find volume of cube, cylinder and rectangle using function overloading.

Code:

```
 #include<iostream>

using namespace std;

float vol(int,int);

float vol(float);     int

vol(int);



   int main()     {
```

```
        cout<<"Volume of cylinder is "<<vol(r,h);
cout<<"\nVolume of cube is "<<vol(a);
cout<<"\nVolume of sphere is "<<vol(r1);
        return 0;
    }     float vol(int
r,int h)
    {
return(3.14*r*r*h);
    }
```

11. Write a C++ program to create a class District. having district_code, district_name,

area_sqft, population, literacy_rate. For displaying details use appropriate manipulators.

The program should contain following menu :

a. Accept details of n district
b. Display details of district.

c. Display details of district having highest literacy rate.

Code:

```
#include<iostream.h>
#include<conio.h>
class district
{
public: int code;
char name[20];
```

```
int area,pop,lrate;
public:
void details()
{ cout<<“\District code\t name\tarea\tpopulation\tliteracy
rate”<<endl; cin>>code >>name>>area>>pop>>lrate;
}
void details(int nm)
{ cout<<“\nDetails of ONE district :\n”;
cout<<“\ncode\tname\tarea\tpop\tlrate \n”;
cout<<“=======================================\n” ;
cout<<“\ncode\tname\tarea\tpop\tlrate \n”;
}
void details(char*)
{ cout<<“\ncode\tname\tarea\tpop\tlrate
\n”;
}
}; void
main()
{
int i,n,hrate;
district s[20]; clrscr(); cout<<“How many
district information:”<<endl; cin>>n;
for(i=0;i<n;i++)
{
s[i].details();
}
hrate=s[0].lrate
for(i=0;i<n;i++)
{
if(hrate<=s[i].lrate)
{
```

```
hrate=s[i].lrate;
}
}
```

Cout<<"Highest literacy rate"<<endl;

Cout<<hrate; cout<<"Details of all district: \n"; cout<<"\District code\t name\tarea\tpopulation\tliteracy rate"<<endl ;

```
cout<<"=====================================\n" ;
for(i=0;i<n;i++)
{
s[i].details(s[i].name);
}
getch();
}
```

11. Create base class called shape. Use this class to store two double type values that could beused to compute the area of figures. Derive two specific classes called cylinder and

rectangle from the base shape. Add to the base class, a member function get_data(),

print_data() to initialize base class data members and display_area(), display_perimeter() to compute and display area and perimeter of shape.

Code:

```
#include<iostream>
using namespace std;

class Shape
{
```

```
   public: double a,b;
      void get_data ()
      {
         cin>>a>>b;
      }
      virtual void display_area () = 0;
};

class Triangle:public Shape
{
   public: void display_area ()
   {
      cout<<"Area of triangle "<<0.5*a*b<<endl;
   }
};

class Rectangle:public Shape
{
   public: void display_area ()
```

```
    {
        cout<<"Area of rectangle "<<a*b<<endl;
    }
};

int main()
{
    Triangle t;
Shape *st = &t;
    cout<<"Enter base and altitude: ";      st-
>get_data();
    st->display_area();

    Rectangle r;
Shape *sr = &r;
    cout<<"Enter length and breadth: ";
    sr->get_data();      sr-
>display_area();
    return 0;
}
```

A book (ISBN) and CD (data capacity) are both types of media (id, title) objects. A person buys 10 media items, each of which can be either book or CD.

Display the list of all
books andCD’s bought. Define the classes and appropriate member functions to accept and

display data.Use pointers and concepts of polymorphism (virtual functions).

#include<iostream>

```
#include<string.h>
#include<stdlib.h>
using namespace std;

class media
{
   protected:       int  id;
char   title[50];   public:
media(int n,char *s)
       {            id = n;
strcpy(title,s);
       }
       virtual void display()=0;
};

class book : public media
{
       long isbn;     public:        book(int
n,char *s,long isbn):media(n,s)
       {
          this->isbn=isbn;
       }

       void display()
       {
          cout<<"\nMedia Id:"<<id;
cout<<"\nTitle :"<<title;
          cout<<"\nISBN :"<<isbn;
       }

};
```

```
class cd: public media
{
        int capacity;      public:
cd(int n,char *s,int c):media(n,s)
        {
capacity=c;
        }

        void display()
        {
           cout<<"\nMedia Id:"<<id;
           cout<<"\nTitle :"<<title;
           cout<<"\nCAPACITY :"<<capacity;
        }
};

int main()
{
   media **m; int inputs; cout <<
"Enter total media inputs : "; cin >>
inputs;

   m = new media*[inputs];
   for(int i=0;i<inputs;i++)
   {        int
id,capacity;
long isbn;         char
title[30],
mediaType;

        cout<<"Enter the media type(C for CD/B for Book):";
cin>>mediaType;
```

```
        cout<<"\nENTER ID:";
cin>>id;
        cout<<"\nENTER TITLE:";
        cin>>title;

        if(mediaType == 'B' || mediaType == 'b')
        {
           cout<<"ENTER ISBN: ";
cin>>isbn;            m[i] = new
book(id,title,isbn);
        }
        else if(mediaType == 'C' || mediaType == 'c')
        {
           cout<<"ENTER CAPACITY: ";
cin>>capacity;            m[i] = new
cd(id,title,capacity);
        }

    }

    cout<<"\n\nMEDIA DETAILS ARE:";
    for(int i=0;i<inputs;i++)
    {
       cout<<"\n_____-___\n";        m[i]-
>display();        cout<<"\n----------\n";
    }

    return 0;
}
```

12. Write a C++ program to sort an Array in Ascending order.

Code:

```
#include  <iostream>

using namespace std;

int main(){
int i, j, size, temp;
int arr[25];
```

```
cout << "Enter the total no. of elements: ";
cin >> size;

cout << "Enter the elements of the array: " << endl;
for (i = 0; i < size; i++){
cin >> arr[i];
}
// Sorting elements in ascending order
for (i = 0; i < size; i++){
for (j = i; j < size; j++){
if (arr[i] > arr[j+1]){
temp = arr[i];
arr[i] = arr[j+1];
arr[j+1] = temp;
}
}
}
// Displaying output
cout << "Elements sorted in the ascending order are: " << endl;
for (i = 1; i <= size; i++){
cout << arr[i] << endl;
}
return 0;
}
```

13. Create a class time that contains hours, minute and seconds as data members. Write the member function to overload operator '+' to add two object of type time, (UseParameterized constructor to accept values for time).

Code:

```
/ Program to add two times

#include
using namespace std;
class Time{ private: int
hour; int minute; public:
                //constructor for initializing objects
                //this constructor uses default arguments
                Time(int h = 0, int m = 0){
                        hour = h; minute =
                        m;
                }
                Time add(Time t){
                        Time temp;
                        temp.minute = minute + t.minute;
                        temp.hour = hour + t.hour;
                        if(temp.minute >= 60){
                        temp.hour++; temp.minute -= 60;
                        }
                        return temp;
                }
                // function to display time void display(){
                cout<<hour<<" hr "<<minute<<" min"<<endl; }
};

int main(){
        Time t1(4,30), t2(5,30),t3;
        T3=t1.add(t2);
        T1.display();
        T2.display();
        T3.display();
        Return 0;
```

```
    }
```

14. Write a C++ Program to read an integer n and print its factorial.

Code:

```
#include <iostream>
using namespace std;
int main()
{
int i,fact=1,number;
cout<<"Enter any Number: ";
cin>>number;
for(i=1;i<=number;i++){
fact=fact*i;
}
cout<<"Factorial of " <<number<<" is: "<<fact<<endl;
return 0;
}
```

15. Write a C++ program using class which contains two data members of type integer.

Create and initialize the object using default constructor, parameterized constructor and

parameterized constructor with default value. Write a member function to display maximumfrom given two numbers for all objects.

Code:

```
#include <iostream>
using namespace std;
```

```
class ConstDemo
{          int n1,
n2;      public:
        ConstDemo()
        {
           n1 = n2 = 0;
        }
        ConstDemo(int a1 , int a2 = 0 )
        {
           if(a2 ==0)
a2 = a1;
n1 = a1;
n2 = a2;
        }
        void maxNumber()
        {
           if ( n1 > n2)
           {
              cout << " n1 ( value = " << n1 << " ) is greater ";
cout << "than n2 ( value = " << n2 << " )";
           }
else
              if ( n2 > n1 )
              {
                 cout << " n2 ( value = " << n2 << " ) is greater ";
cout << "than n1 ( value = " << n1 << " )";
              }
else
              {
                 cout << " Both values n1 ( value = " << n1 << " ) ";
cout << "and n2 ( value = " << n2 << " ) are same. ";
              }
```

```
        }
}; int
main() {
while(true)
  {    int n, n1, n2;    cout << "* * * * * * * * * * * * * * * * * *
* * * * * * * * * * *
*"<<endl;
    cout << " \t Accept two integer data members " << endl;
cout << "    Display maximum from two integer data numbers "
<< endl;
    cout << "* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
    cout << " \t Initialize the object using " << endl;
cout << " 1 : Default Constructor \t 2 : Parameterized
Constructor" << endl;
    cout << "    3 : Parameterized Constructor with default value" <<
endl;
    cout << "\t \t Exit : Other" << endl;    cout << " \t
Choose Appropriate option " << endl;
    cout << "* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
    cout << " \t ";
cin >> n;
    switch(n)
    {
case 1:
        {
        ConstDemo c1;
        cout << "* * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
        cout << " \t Display maximum from two integer data numbers
" << endl;
```

```
        cout << "* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
c1.maxNumber();
break;        }
case 2:
        {
        cout << " \t Accept first Parameter " << endl << " \t ";
cin >> n1;    cout << " \t Accept Second Parameter " <<
endl << " \t ";  cin >> n2;
        ConstDemo c2( n1, n2);
        cout << "* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
        cout << "    Display maximum from two integer data numbers
" << endl;
        cout << "* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
        c2.maxNumber();
break;        }
case 3:
        {
        cout << " \t Accept One Parameter " << endl ;         cout <<
" \t ( This Parameter use as default Parameter )" << endl;
        cout << " \t ";
cin >> n1;
        ConstDemo c3( n1);
        cout << "* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
        cout << " \t Display maximum from two integer data numbers
" << endl;
        cout << "* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
c3.maxNumber();
```

```
break;        }
default:
       cout << " \t Thank You to Use this Program !";
exit(0);
   }     cout <<
endl;
   cout << "* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
   cout << " \t Are you want to continue ... " << endl;     cout
<< "* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*"<<endl;
   cout << " \t IF Yes : 1 \t No : Any other value " << endl << " \t
";     cin >>
n;     if(n ==
1)
   {
      continue;
   }
else
   {
      cout << " \t Thank You to Use this Program !";
exit(0);
   }
  }
}
```

17. Create a C++ class for a student object with the following attributes—roll no, name, number of subjects, marks of subjects. Write member function for accepting marks and display all information of student along with total and Percentage. Display mark list with using of manipulators.

Program Code for run:

```
#include <iostream>
```

```
using namespace std;
#include<iomanip>
class Student
{
        private:
                int Roll_Number, Marks[5];
char Stud_Name[80], Class;
float Per;
        public:
                void accept();
                void display();
};
void Student :: accept()
{
        cout << " Accept Student Roll Number "<< endl;
cin >> Roll_Number;          cout << " Accept Student
Name"<< endl;          cin >> Stud_Name;          cout
<< " Accept Student Marks "<< endl;          float total
= 0;
        for( int i = 0; i < 5; i++)
        {
                cout << " \t Subject "<< i + 1 << endl;
                cin >> Marks[i];
total = total + Marks[i];
        }
        Per = total / 5; // calculate Percentage
        if(Per < 60)
Class='B';
        if((Per >= 60)&&(Per < 70))
                Class='A';
if(Per >= 70)
                Class='O';
}
```

```
void Student :: display()
{
        cout << setw(7) << Roll_Number << setw(7) <<
Stud_Name << setw(7);
        for( int i = 0; i < 5; i++)                cout<<
Marks[i] << setw(7);          cout << setw(7) <<
setprecision(2) << Per << "    " <<
Class << "\n";
}

int main()
{
        int n, i;
        Student p[20];
        cout << " * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;        cout << " \t How many Student data to
  enter " << endl;        cout << " * * * * * * * * * * * * * * * * * * *
  * * * * * * * * *
* * * * * " << endl;        cin >> n;        cout << " * * * * * * *
  * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;
        cout << " \t Enter Student Details " << endl;
        for ( i = 0 ; i < n ; i ++)
        {
           cout<<"* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;
           cout<<"\t Enter Student Details of Student := " << i + 1
<< endl;
           cout<<"* * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;
           p[i].accept();
        }
```

```
            cout << " * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;
            cout << " Display Student Details " << endl;            cout
<< " * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;
            cout << setw(7) << "RNO" << setw(7) << "SName" <<
setw(7) << "Sub1";            cout << setw(7) << "Sub 2"<<
setw(7) << "Sub 3" << setw(7) << "Sub4";          cout<< setw(7)
<< "Sub 5" << setw(7) << "Per" << setw(7) << "Class" <<endl;
            cout << " * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;
            for ( i = 0 ; i < n ; i ++)                       p[i].display();
cout << " * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;
            return 0;
}
```

19. Write a program in C++ to demonstrate the manipulators.

```
#include<iomanip>

#include<iostream>

using namespace std;

int main()

{

cout<< setw(10) << 1 << endl;

cout<< setw(10) << 10 << endl;

cout<< setw(10) << setfill('*')<< 100 << endl;
cout<< setprecision(2) << 22/7.0 <<
endl; cout<< setbase(8) << 65 << endl;
```

```
cout<< setbase(10) << 0101 << endl;
cout<< setbase(10) << 0x41 << endl;

cout<< setw(5) << setiosflags(ios::left)<<"Hello"<< endl;
}
```

20. Create a base class Conversion. Derive three different classes Weight (Gram,

Kilogram), Volume (Milliliter, Liter), Currency (Rupees, Paise) from Conversion class.

Write a C++ program to perform read, convert and display operations. (Use Pure virtual

function)

code:

```
#include<iostream.h>
#include<conio.h>
class conversion
{
public:
virtual void show()=0;
};
class weight:public conversion
{
int gm,kg;
public:
void get1()
{
cout<<"\n Enter Gram : ";
cin>>gm;
}
void show()
{
kg=gm/1000;
gm=gm % 1000;
cout<<"\n "<<kg<<" :="" kg="" &="" "<<gm<<"="" gram";
 }
};
class volume:public conversion
{
int ltr,mltr;
public:
void get2()
```

```
{
cout<<"\n Enter Milliliter : ";
cin>>mltr;
ltr=mltr % 1000;
mltr=mltr / 1000;
}
void show()
{
cout<<"\n "<<ltr<<" :="" liter="" &="" "<<mltr<<"="" milliliter";
 }
};
class currency:public conversion
{
float pse,rs;
public:
void get3()
{
cout<<"\n Enter Rupees : ";
cin>>rs;
pse=rs*100;
}
void show()
{
cout<<"\n Paise : "<<pse;
 }
};
void main()
{
clrscr();
weight w;
volume v;
currency c;
conversion *p;
cout<<"\n Accept Info..... ";
w.get1();
```

```
v.get2();
c.get3();
cout<<"\n_____";
p=&w;
p->show();
cout<<"\n_____";
p=&v;
p->show();
cout<<"\n_____";
p=&c;
p->show();
getch();
}
```

```
</pse;

</ltr<<"></kg<<"
>
```

21. Write a menu driven C++ program using class to perform all arithmetic operation.

(+, -, *, /) (use inline function).

Program Code for run:

```
#include <iostream>
using namespace std;
class ArithmaticOpe
{
   private:
      int n1, n2;
public:
      void accept();
void addition();      void
subtraction();      void
```

```
division();        void
multiplication();
};
inline void ArithmaticOpe :: accept()
{
   cout << " Enter First Integer n1 " << endl << " \t ";
cin >> n1; cout << " Enter Second Integer n2 " <<
endl << " \t ";  cin >> n2;
}
inline void ArithmaticOpe :: addition()
{
   cout << " Addition : " << n1 + n2 << endl;
}
inline void ArithmaticOpe :: subtraction()
{
   cout << " Subtraction : " << n1 - n2 << endl;
}
inline void ArithmaticOpe :: division()
{
   cout << " Division : " << (float) n1 / n2 << endl;
}
inline void ArithmaticOpe :: multiplication()
{
   cout << " Multiplication : " << n1 * n2 << endl;
}
int main()
{
          int n, i;
          ArithmaticOpe obj;
          cout << " * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;
          cout << " \t Enter two integer numbers " << endl;
```

```
cout << " * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * " << endl;
obj.accept();
while(true)
{
cout << " * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * " << endl;
cout << "\tEnter the appropiate integer number for
arithmetic operation";     cout << endl;
cout << " \t Addition : 1 \t\t Subtraction : 2" << endl;
cout << " \t Division : 3 \t\t Multiplication : 4" << endl;
cout << " \t\t QUIT : 5 OR Other" << endl;          cout << " * *
* * * * * * * * * * * * * * * * * * * * * * *
* * * * * * " << endl;
cin >> n;
switch(n)
{
case 1: obj.addition();
break;          case 2:
obj.subtraction();
break;          case 3:
obj.division();
break;
case 4: obj.multiplication();
break;
default : cout << "Thank You to use this program ! ";
exit (0);
}
}        return
0;
}
```

22. Write a C++ program to create a class novel which contains data member as id, name

andauthor. Write member function to accept and display novel information. Also display

the count of novels (use static data member to maintain the count of novel).

Code:

```
#include<iostream.h>
#include<conio.h>
class book
{
        int id;
        char name[20],author[20],pub[20];
static int cnt;         public:
        void getdata()
        {
                cout<<"\nEnter book id : ";
cin>>id;
```

```
                        cout<<"\nEnter book name : ";
cin>>name;
                        cnt++;
cout<<"\nEnter author name : ";
cin>>author;                        cout<<"\nEnter
publication : ";                        cin>>pub;
            }
            void display()
{
                        //cout<<"\n\n**********OUTPUT**********
*******";
                        cout<<"\nbook id = "<<id;
cout<<"\nbook name = "<<name;
cout<<"\nAuthor name = "<<author;
cout<<"\npublication = "<<pub;
            }
            static void no_of_book()
            {
                        cout<<"\nNumber of book = "<<cnt;
            }
}; int

book::cnt;

void main()
{
            clrscr();
            book b[20];int n;
cout<<"\nEnter no f Books : ";
cin>>n;               for(int i=0;i<n;i++)
                        b[i].getdata();
cout<<"\n Book Information are : \n \n";
for( i=0;i<n;i++)
```

```
                    b[i].display();          b[i-
1].no_of_book();
          getch();
}
```

23. Write a C++ program to accept and display employee (e_no,e_name,e_designation)

```
1. #include <iostream>
2. using namespace std;  3. class Employee {  4.
   public:
5.              int id; //data member (also instance variable)
6.              string name; //data member(also instance variable)
7.              float salary;
8.              Employee(int id, string name, float salary)
9.              {
10.             this->id = id;
11.             this->name = name;
12.             this->salary = salary;
13.             }
14.             void display()
15.             {
16.             cout<<id<<"  "<<name<<"  "<<salary<<endl;
17.             }
18.             };
19.             int main(void) {  20.

    Employee e1 =Employee(101, "Sonoo", 890000); //creating an object
of Employee
```

```
21.
    Employee e2=Employee(102, "Nakul", 59000); //creating an object of
Employee
22.         e1.display();
23.         e2.display();
24.         return 0;
25.         }
```

Q27 Create a base class Media. Derive two different classes Book (Book_id, Book_name,

Publication, Author, Book_price) and CD (CD_title, CD_price,CD_capacity) from

Media.Write a C++ program to accept and display information of both Book and CD.

(Use pure virtual function)

```
#include<iostream.h>
#include<conio.h>
class media
{
public: virtual void
display()=0;
}; class book:public
media
{
int bid; char
bnm[20],pub[20],auth[20];
float price; public: void
getbook() { cout<<"\n Enter
```

```
Book Id : "; cin>>bid;
cout<<"\n Enter Book name : ";
cin>>bnm;
cout<<"\n Enter Publication : ";
cin>>pub; cout<<"\n Enter
Auther : "; cin>>auth;
cout<<"\n Enter Price : ";
cin>>price; }

void display() { cout<<"\n Book
id : "<<bid; cout<<"\n Book
Name : "<<bnm; cout<<"\n
publication : "<<pub; cout<<"\n
Auther : "<<auth; cout<<"\n
Price : "<<price;
 }
};
class cd:public media
{
char tit[20];
float cd_price;
public: void
getcd() {
cout<<"\n Enter Cd tital : ";
cin>>tit; cout<<"\n Enter Cd price
: "; cin>>cd_price; } void display()
{ cout<<"\n Cd_Tital : "<<tit;
cout<<"\n Cd_price : "<<cd_price;
 } }; void
main()
{
clrscr();
book b;
```

```
b.getbook(); cout<<"\n------------------------
_______"; cd c;
c.getcd(); cout<<"\n_____________________
----------"; media *m; m=&b;
m->display();
m=&c;
cout<<"\n-----------------------------------"; m->display(); getch();}
```

28. Write a C++ program to read the contents of a text file. Count and display number of

characters, words and lines from a file. Find the number of occurrences of a given word present in a file.

Code:

```
#include<iostream>
#include<fstream>
#include<string.h>
#include<cstdlib>
using namespace std;
int main() {
    int noc=0,now=0,nol=0;
FILE *fr;      char
fname[20],ch;

    cout<<"\n Enter Source File Name : ";
gets(fname);      fr=fopen(fname,"r");
if(fr==NULL)
    {
        cout<<"\n Invalid File Name. \n No such File or Directory ";
exit(0);
    }
ch=fgetc(fr);
```

```
while(ch!=EOF)
{          if(ch!=' '
&& ch!='\n')
noc++;
if(ch==' ')
now++;
if(ch=='\n')
        {
nol++;
now++;
        }
        ch=fgetc(fr);
    }      fclose(fr);      cout<<" ----------
----------";      cout<<"\n Total No. of
Characters  : "<<noc;      cout<<"\n Total No. of
Words       : "<<now;      cout<<"\n Total No. of
Lines       : "<<nol;

    return 0;
}
```

31. Q1. Write a C++ Program to prompt the user to input 3 integer values and print these values

inforward and reverse order.

```
#include<stdio.h>
                    #include<conio.h>
                    int main()
                    {
                    int a,b,c;
                    printf("enter a ,b,c");
```

```
scanf("%d%d%d",&
a,&b,&c);
printf("the forward
order:%d,%d,%d",a,
b,c); printf("the
reverse
order:%d%d%d",c,b
,a); getch(); return 0;
}
```

32. Create a class for different departments in a college containing data members as Dept_Id, Dept_Name, Establishment_year, No_of_Faculty, No_of_students. Write a C++ program with following member function:To display department details of aspecific Department.

Code:

```
#include<iostream.h>
#include<conio.h>
#include<fstream.h>
class dept
{
int did char
dname[20]; char
hod[15]; int
nos;
```

```
public:void accept()
{
cout<<"enter the dept id";
cin>>did;
cout<<"enter the dept name";
cin>>dname;
cout<<"enter the hod";
cin>>hod;
cout<<"enter the no of staff";
cin>>nos;
}
void display()
{
cout<<"the dept id is-"<<did;
 cout<<"the dept name is-"<<dname;
 cout<<"the hod is-"<<hod;
 cout<<" the no of staff is-"<<nos;
 }
}
void main()
{
dept d[5];
int n,i;
clrscr();
fstrean file;
file.open("dept.txt",ios::in|ios::out);
cout<<"enter the no of record you want -";
cin>>n;
for(i=0;i<n;i++)
 {
d[i].accept();
file.write((char*)&d[i],sizeof(d[i]));
}
cout<<"\ndetails od department from the file-";
for(i=0;i<n;i++)
```

```
 {
file.read((char*)&d[i],sizeof(d[i]));
d[i].display();
}
file.close();
getch();
}

</n;i++)
</n;i++)
</nos;
</hod;
</dname;
</did;
```

35. Q2. Write a program to create a class Person which contains data members as P_name,

P_City, P_Contact_no. Write the member functions to accept and display the details of 5 persons.

Code:

CPP PROGRAMS

Write a C++ program to create a class Person which contains data members as P_Name, P_City, P_Contact_Number. Write member functions to accept and display five Persons information. Design User defined Manipulator to print P_Contact_Number. (For Contact Number set right justification, maximum width to 10 and fill remaining spaces with '*')

JUNE 28, 2020

```
//Note: If you are not using Borland or Turbo C then add the following line after header files :
```

```
//using namespace std;

#include<iostream.h>
#include<conio.h>
#include<iomanip.h> //manipulator
class person {
   char pname[10],city[10]; //pname=person name , no=ph no
   int no;
public:
   void accept()
   {
    cout<<"Enter person name: ";
cin>>pname;
     cout<<"Enter person city: ";
cin>>city;
     cout<<"Enter phone number: ";
cin>>no;
   }    void
disp()
   {
     cout<<"\nPerson name: "<<pname;
cout<<"\nPerson city: "<<city;
```

```
cout.width(10);       cout.fill('*');
cout.right;
      cout<<"\nperson phone number: "<<setw(10)<<no;
   } }; int
main() {
   person p; //creating instance of class person
   p.accept(); //calling member functions
   p.disp();
getch();
return(0);
}
```

37. Write a C++ Program to compute the sum of specified number of prime numbers.

Code:

```
#include  <iostream>
using namespace std;
bool isPrime(int n){
   for(int i = 2; i < n/2; i++){
```

```
    if(n%i == 0){
return false;
    }   }
return true;
} int findPrimeSum(int n){
int sumVal = 0;   for(float i
= 2; i <= n; i++){
if(isPrime(i))        sumVal
+= i;
  }
  return sumVal;
} int
main(){
```

```
  int n = 15;   cout<<"The sum of prime number between 1
to "<<n<<" is
"<<findPrimeSum(n);
  return 0;
}
```

Output

The sum of prime number between 1 to 15 is 45

38. Implement multiple and hierarchical inheritance. The class All Rounder derives information from both Bowler and Batsman classes which in turn derive information from

theclass Cricketer. Define all 4 classes and write a program to do the following:

- Accept the details of n objects.
- Display the details of n objects.
- Display the object having highest runs.

Code:

```
#include<iostream>
using namespace std;

class Cricketer
{
  protected:
   char name[20];
};

class Bowler:public Cricketer
{
  public:
void accept()
        {
          cout<<"\n\n Enter Bowler name: ";
          cin>>name;
        }

        void display()
        {
           cout<<"\n\n Bowler : "<<name;
        }
};

class Batsman:public Cricketer
```

```
{
   public:
void accept()
        {
          cout<<"\n\n Enter Batsman name: ";
          cin>>name;
        }

        void display()
        {
           cout<<"\n\n Batsman : "<<name;
        }

};


class Allrounder:public Bowler,public Batsman
{    char
name[10];
public:         void
accept()
        {
           Bowler::accept();
Batsman::accept();            cout<<"\n\n Enter
Allrounder name: ";            cin>>name;
        }

        void display()
        {
           Bowler::display();
Batsman::display();
           cout<<"\n\n Allrounder : "<<name;
        }
```

```
};
int main()
{
   Allrounder a;

   a.accept();
   a.display();

  return 0;
}
```

39. Write a class Complex (real, img) along with appropriate constructors. Also write appropriate functions to overload '+' and '-' operator.

Code:

```
#include<iostream>
using namespace std;
class Complex
{
 private:
  float real,img;
public:
  Complex()
  {
real=0;
   img=0;
  }
  void accept()
  {
   cout<<"Enter the complex number:"<<"\n";
cout<<"Real:";    cin>>real;
cout<<"Imaginary:";
```

```
   cin>>img;
  }
  void display()
  {
   cout<<"complex number is:";
cout<<real<<"+"<<img<<"i"<<"\n";
  }
  Complex(float a,float b)
  {
real=a;
   img=b;
  }
  friend Complex operator +(Complex c1,Complex c2)
  {
   c1.real=c1.real+c2.real;
   c1.img=c1.img+c2.img;
return c1;
  }

  friend Complex operator -(Complex c1,Complex c2)
  {    c1.real=c1.real-
c2.real;
c1.img=c1.img-c2.img;
   return c1;
  }

};
int main()
{
 int ch;
 Complex c3;            //default constructor
 Complex c4(4,5);               //parameterized constructor
```

```
 Complex c5;
c5.accept();
```

```
 cout<<"1st";
c4.display();
cout<<"\n";
cout<<"2nd";
c5.display();
cout<<"\n";


 do
{
 cout<<"\n"<<"Enter your choice:";
cout<<"1.Addition"<<"\n"<<"2.Substraction"<<"\n";
cin>>ch;  switch(ch)
 {
  case 1:         cout<<"Addition:";
                              //c3=c4+c5;
        c3=operator+(c4,c5); c3.display();
        cout<<"\n";       break;
  case 2:   cout<<"Substraction:";
                    // c3=c4-c5;
        c3=operator-(c4,c5);   c3.display();
                    cout<<"\n";
                    break;

default:
                                        cout<<"EXIT";
 }
 }while(ch<=2);
return 0;
}
```

Q. Write a C++ program to create a class novel which contains data member as id, name

and author. Write member function to accept and display novel information. Also display

the count of novels (use static data member to maintain the count of novel) code:

```
#include<iostream.h>
#include<conio.h>
class book
{
        int id;
        char name[20],author[20],pub[20];
static int cnt;         public:
        void getdata()
        {
                cout<<"\nEnter book id : ";
cin>>id;                        cout<<"\nEnter book
name : ";                        cin>>name;
                cnt++;
cout<<"\nEnter author name : ";
cin>>author;                            cout<<"\nEnter
publication : ";                        cin>>pub;
        }
        void display()
        {
```

```
                        //cout<<"\n\n**********OUTPUT**********
*******";
                        cout<<"\nbook id = "<<id;
cout<<"\nbook name = "<<name;
cout<<"\nAuthor name = "<<author;
cout<<"\npublication = "<<pub;
                }
                static void no_of_book()
                {
                        cout<<"\nNumber of book = "<<cnt;
                }
}; int

book::cnt;

void main()
{
                clrscr();
                book b[20];int n;
cout<<"\nEnter no f Books : ";
cin>>n;                 for(int i=0;i<n;i++)
                          b[i].getdata();
cout<<"\n Book Information are : \n \n";
for( i=0;i<n;i++)
b[i].display();                 b[i-1].no_of_book();
                getch();
}
```