```
/*Implement a class Complex which represents the Complex Number data type.
Implement
the following operations:
1. Constructor (including a default constructor which creates the complex number
0+0i).
2. Overloaded operator+ to add two complex numbers.
3. Overloaded operator* to multiply two complex numbers.
4. Overloaded << and >> to print and read Complex Numbers.*/

#include<iostream>
using namespace std;
class complex
{
      float x;
      float y;
public:
        complex()
      {
            x=0;
            y=0;
      }

      complex operator+(complex);
      complex operator*(complex);
      friend istream &operator >>(istream &input,complex &t)
      {
            cout<<"Enter the real part";
            input>>t.x;
            cout<<"Enter the imaginary part";
            input>>t.y;
      }
      friend ostream &operator <<(ostream &output,complex &t)
      {
                  output<<t.x<<"+"<<t.y<<"i\n";
      }

};

complex complex::operator+(complex c)
{
      complex temp;
      temp.x=x+c.x;
      temp.y=y+c.y;
      return(temp);
}


complex complex::operator*(complex c)
{
      complex temp2;
      temp2.x=(x*c.x)-(y*c.y);
      temp2.y=(y*c.x)+(x*c.y);
      return (temp2);
}


int main()
{
      complex c1,c2,c3,c4;
        cout<<"Default constructor value=\n";
        cout<<c1;
      cout<<"\nEnter the 1st number\n";
      cin>>c1;
      cout<<"\nEnter the 2nd number\n";
```

```
      cin>>c2;
      c3=c1+c2;
      c4=c1*c2;
      cout<<"\nThe first number is ";
      cout<<c1;
      cout<<"\nThe second number is ";
      cout<<c2;
      cout<<"\nThe addition is ";
      cout<<c3;
      cout<<"\nThe multiplication is ";
      cout<<c4;
      return 0;
}
/*
student@student-OptiPlex-3010:~$ ./a.out
Default constructor value=
0+0i

Enter the 1st number
Enter the real part2
Enter the imaginary part4

Enter the 2nd number
Enter the real part4
Enter the imaginary part8

The first number is 2+4i

The second number is 4+8i

The addition is 6+12i

The multiplication is -24+32i
student@student-OptiPlex-3010:~$
*/
```

```
/*============================================================================
Assignment no : A7
Title       : Implement C++ program to write a class template to represent a
generic
            vector. Include following member functions:
            To create the vector.
            To modify the value of a given element
            To multiply by a scalar value
            To display the vector in the form (10,20,30,...)
CLASS       :
Batch       :
ROLL NO           :
Date        :
============================================================================*/
#include<iostream> //header file
#include<vector>
using namespace std; //declared scope of program
void display (vector<int>&v) //function declaration
{
      for(int i=0;i<v.size();i++)
      {
            cout<<"   "<<v[i];
      }
      cout<<" \n";
}
int main()
{
      vector<int> v;
      cout<<"\n Initial Size: "<<v.size();
      int x;
      cout<<"\n Enter The 5 Element: ";
      for(int i=0;i<5;i++)
      {
            cin>>x;
            v.push_back(x);
      }
      cout<<"\n Size After Insertion: "<<v.size();
      cout<<"\n Vector Element: ";
      display(v);
      cout<<"\nVector Element After insertion of 3 at End of Vector: ";
      v.push_back(3);
      display(v);
      vector<int>::iterator itr=v.begin();
      itr=itr+3;
      v.insert(itr,9);
      cout<<"\n Content After Insertion 9 at position 3rd: ";
      display(v);
      v.erase(v.begin()+3,v.begin()+5);
      cout<<"\n After Erasing(3rd-4th position): ";
      display(v);
}

/*-------------------------OUTPUT-------------------------------------
gescoe@gescoe-OptiPlex-3010:~/Desktop/se$ g++ fifth.cpp
gescoe@gescoe-OptiPlex-3010:~/Desktop/se$ ./a.out

 Initial Size: 0
 Enter The 5 Element: 6 9 4 2 1

 Size After Insertion: 5
 Vector Element:    6   9   4   2   1

 Vector Element After insertion of 3 at End of Vector:    6   9   4   2
                                                          1   3
```

```
 Content After Insertion 9 at position 3rd:    6   9   4   9   2   1   3

 After Erasing(3rd-4th position):    6   9   4   1   3

--------------------------------------------------------------------------*/
```

```
/*Develop an object oriented program in C++ to create a database of student
information system containing the following information: Name, Roll number,
Class, division, Date of Birth, Blood group, Contact address, telephone number,
driving license no. etc Construct the database with suitable member functions
for initializing and destroying the data viz constructor, default constructor,
Copy constructor, destructor, static member functions, friend class, this
pointer, inline code and dynamic memory allocation operators-new and delete.*/
#include<iostream>
#include<string.h>
using namespace std;

class person_additional_info
{
      char address[20],license[20],insurance[20];
      long int contact;

      public:
            person_additional_info() //Default constructor
            {
                  strcpy(address,"XYZ");
                        strcpy(license,"XY-0000000000");
                  strcpy(insurance,"XY00000000X");
                  contact=000000000;

            }

               ~person_additional_info() //Destructor
                {
                        cout<<"I am in Destructor";
                }

             friend class person;  // Declaration Friend class
};

//Definition of friend class
class person
{
      char name[20], dob[10], blood[10];
      float ht,wt;
      static int count; // Static variable
      person_additional_info *pai;

      public:
            person()  //Default constructor
            {       strcpy(name,"XYZ");
                        strcpy(dob,"dd/mm/yy");
                        strcpy(blood,"A +");
                        ht=0;
                        wt=0;
                  pai=new person_additional_info;
                 }
            person(person*p1) //Copy constructor
            {       strcpy(name,p1->name);
                        strcpy(dob,p1->dob);
                        strcpy(blood,p1->blood);
                        ht=p1->ht;
                        wt=p1->wt;
                  pai=new person_additional_info;
                  strcpy(pai->address,p1->pai->address);
                        strcpy(pai->license,p1->pai->license);
                        strcpy(pai->insurance,p1->pai->insurance);
                  pai->contact=p1->pai->contact;
                }
            static void showcount()  //Static member function
```

```
            {
                  cout<<"\nNo of records count="<<count<<"\n";
                }
                  ~person()    //Destructor
                {
                     cout<<"\nI am in Destructor\n";
                }
            void getdata(char name[20]);
            inline void display();   // Inline function declaration

};
void person::getdata(char name[20])
{
      strcpy(this->name,name);     //this pointer
      cout<<"\n Enter date of birth";
            cin>>dob;
      cout<<"\n Enter blood group";
        cin>>blood;
      cout<<"\n Enter height";
        cin>>ht;
      cout<<"\n Enter weight";
        cin>>wt;
      cout<<"\n Enter address";
      cin>>pai->address;
      cout<<"\n Enter Licence number";
        cin>>pai->license;
      cout<<"\n Enter Insurance policy number";
        cin>>pai->insurance;
      cout<<"\n Enter Contact number";
      cin>>pai->contact;
      count++;
}
//inline function definition
void person::display()
{
      cout<<"\t"<<name;
      cout<<"\t"<<dob;
      cout<<"\t"<<blood;
      cout<<"\t"<<ht;
      cout<<"\t"<<wt;
      cout<<"\t"<<pai->address;
      cout<<"\t"<<pai->license;
      cout<<"\t"<<pai->insurance;
      cout<<"\t"<<pai->contact;
}

int person::count;  //Static variable definition

int main()
{
      person *p1,*p2;
      int ch;
      p1=new person;  //call default constructor & dynamic memory allocation
      p2=new person(p1); //call copy constructor
      cout<<"\tName";
      cout<<"\tDob";
      cout<<"\t   Blood";
      cout<<"\tHt";
      cout<<"\tWt";
      cout<<"\tAddress";
      cout<<"\tLicense";
      cout<<"\tInsurance";
      cout<<"\tContact";
      cout<<endl;
```

```
	cout<<"Default Constructor Value \n";
	p1->display();
	cout<<"\n";
	cout<<"Copy Constructor Value \n";
	p2->display();
	  int n;
	cout<<"\nEnter how many records you want??";
	cin>>n;
	person p3[n];	//array of object
	char name[20];
	int x=0;
	do
	{
		cout<<"\nWelcome to Personal database system";
		cout<<"\n1.Enter the record";
		cout<<"\n2.Display the record";
		cout<<"\n3.Exit";
		cin>>ch;
		switch(ch)
		{
			case 1:
			{
				cout<<"\nEnter the Name ";
				cin>>name;
				p3[x].getdata(name);
				person::showcount();  // calls static function
				x++;
			break;
			}
			case 2:
			{
				cout<<"\tName";
				cout<<"\tDob";
				cout<<"\t   Blood";
				cout<<"\tHt";
				cout<<"\tWt";
				cout<<"\tAddress";
				cout<<"\tLicense";
				cout<<"\tInsurance";
				cout<<"\tContact";
				for(int i=0;i<n;i++)
				{
					cout<<"\n";
					p3[i].display(); //calls inline function
				}
				break;
			}
		}
	}while(ch!=3);
	delete p1; //dynamic memory de-allocation
	delete p2;
	return 0;

}

/*student@student-OptiPlex-3010:~$ g++ groupa6.cpp
student@student-OptiPlex-3010:~$ ./a.out
	Name  Dob	Blood	Ht	Wt	Address	License	Insurance	Contact
Default Constructor Value
	XYZ	dd/mm/yy	A +	0	0	XYZ	XY-0000000000	XY00000000X 0
Copy Constructor Value
	XYZ	dd/mm/yy	A +	0	0	XYZ	XY-0000000000	XY00000000X 0
Enter how many records you want??2
```

```
Welcome to Personal database system
1.Enter the record
2.Display the record
3.Exit1

Enter the Name abc

 Enter date of birth15/5/2016

 Enter blood groupo+

 Enter height5

 Enter weight50

 Enter addresspune

 Enter Licence numberjhdf87

 Enter Insurance policy numberhdjsg7786

 Enter Contact number989898989

No of records count=1

Welcome to Personal database system
1.Enter the record
2.Display the record
3.Exit2
      Name  Dob      Blood    Ht    Wt    Address     License     Insurance
      Contact
      abc   15/5/2016   o+    5     50    pune  jhdf87      hdjsg7786   989898989
      XYZ   dd/mm/yy    A +   0     0     XYZ   XY-0000000000     XY00000000X 0
Welcome to Personal database system
1.Enter the record
2.Display the record
3.Exit3

I am in Destructor

I am in Destructor

I am in Destructor

I am in Destructor

*/
```

```
/*Write a C++ program create a calculator for an arithmetic operator (+, -,
*, /). The program should take two operands from user and performs the operation
on those two operands depending upon the operator entered by user. Use a switch
statement to select the operation. Finally, display the result. Some sample
interaction with the program might look like this:
Enter first number, operator, second number: 10 / 3
Answer = 3.333333
Do another (y/n)? y
Enter first number, operator, second number: 12 + 100
Answer = 112
Do another (y/n)? n
*/
#include<iostream>
using namespace std;
class Calculator
{
   private:
         float num1,num2,result;
         char op;
   public:
         void get();
         void calculate();
};
void Calculator::get()
{
   cout<<"\nEnter first number, operator, second number:\n";
   cin>>num1;
   cin>>op;
   cin>>num2;
}
void Calculator::calculate()
{

   switch(op)
   {
      case '+':
                  result=num1+num2;
                  cout<<" Answer = "<<result;

                  break;
      case '-':
                 result=num1-num2;
                  cout<<" Answer = "<<result;

            break;
      case '*':
                   result=num1*num2;
                  cout<<" Answer = "<<result;

             break;
      case '/':
                  if(num2==0)
                  cout<<"\n Error. Not valid.";
                   result=num1/num2;
                  cout<<" Answer = "<<result;

      break;
   }

}
int main()
{
   char ag;
   Calculator obj;
```

```
   x:obj.get();
   obj.calculate();
   cout<<"\n Do another (y/n)? ";
   cin>>ag;
   if(ag=='y'||ag=='Y')
   goto x;
   return 0;
}

/*OUTPUT:
student@student-OptiPlex-3010:~$ g++ groupa5.cpp
student@student-OptiPlex-3010:~$ ./a.out

Enter first number, operator, second number:
10/3
 Answer = 3.33333
 Do another (y/n)? y

Enter first number, operator, second number:
12+100
 Answer = 112
 Do another (y/n)? n*/
```

```
/* Problem Statement:

Create employee bio-data using following classes :
i)    Personal record
ii)   Professional record
iii)  Academic record Assume appropriate data members and member function to
accept required data & print bio-data. Create bio-data using multiple
inheritance using C++

*/


#include<iostream>
using namespace std;
class personal
{
      protected:
            char name[50];
            char address[50];
            char birthdate[50];
            char gender;
      public:
            void get_personal()
            {
                  cout<<"\nEnter name";
                  cin>>name;
                  cout<<"\nEnter Address";
                  cin>>address;
                  cout<<"\nEnter Birthdate(dd/mm/yyyy)";
                  cin>>birthdate;
                  cout<<"\nEnter gender(M/F)";
                  cin>>gender;
            }
};
class professional
{
      protected:
            int expinnoofyear;
            char orgname[50];
            char projname[50];
            char projdetails[50];
      public:
            void get_professional()
            {
                  cout<<"\nEnter number of years of exp";
                  cin>>expinnoofyear;
                  cout<<"\nEnter organization name";
                  cin>>orgname;
                  cout<<"\nEnter project name";
                  cin>>projname;
                  cout<<"\nEnter project Details";
                  cin>>projdetails;
            }
};

class academic
{
      protected:
            int year;
            int marks;
            int percentage;
            char clas[50];
      public:
```

```
            void get_academic()
            {
                  cout<<"\nEnter academic year";
                  cin>>year;
                  cout<<"\nEnter total marks";
                  cin>>marks;
                  cout<<"\nEnter percentage";
                  cin>>percentage;
                  cout<<"\nEnter class";
                  cin>>clas;
            }
};
class biodata: public personal,public academic,public professional
{
      public:
            void display()
            {
                  cout<<"\n--------------Employee Biodata------------"<<endl;

      cout<<"-----------------------------------------------------"<<endl;
                  cout<<"Personal Details"<<endl;
                  cout<<"Name:"<<name<<endl;
                  cout<<"Address:"<<address<<endl;
                  cout<<"Birthdate:"<<birthdate<<endl;
                  cout<<"Gender:"<<gender<<endl;

      cout<<"-----------------------------------------------------"<<endl;
                  cout<<"----------------Academic Details--------------"<<endl;
                  cout<<"Academic
Year\t"<<"marks\t"<<"percentage\t"<<"class\t"<<endl;
                  cout<<year<<"\t"<<marks<<"\t"<<percentage<<"\t"<<clas<<endl;

      cout<<"-----------------------------------------------------"<<endl;
                  cout<<"-----------------Professional
Details--------------------"<<endl;
                  cout<<"Organization Name:"<<orgname<<endl;
                  cout<<"Years of Experince:"<<expinnoofyear<<endl;
                  cout<<"Project Done:"<<projname<<endl;
                  cout<<"Project Details:"<<projdetails<<endl;
            }
};


int main()
{
      biodata b;
      b.get_personal();
      b.get_academic();
      b.get_professional();
      b.display();
        return 0;
}
/*student@student-OptiPlex-3010:~$ g++ groupa13.cpp
student@student-OptiPlex-3010:~$ ./a.out

Enter nameXYZ

Enter AddressPUNE

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

Enter gender(M/F)F

Enter academic year2016
```

```
Enter total marks500

Enter percentage70

Enter classFirst

Enter number of years of exp5

Enter organization namePQR

Enter project namePPP

Enter project DetailsIII

--------------Employee Biodata------------
-----------------------------------------------------
Personal Details
Name:XYZ
Address:PUNE
Birthdate:15/5/2016
Gender:F
-----------------------------------------------------
----------------Academic Details--------------
Academic Year     marks percentage  class
2016  500   70    First
-----------------------------------------------------
-----------------Professional Details--------------------
Organization Name:PQR
Years of Experince:5
Project Done:PPP
Project Details:III
student@student-OptiPlex-3010:~$
*/
```

```
/*Write C++ Program with base class convert declares two variables, val1 and
val2,
which hold the initial and converted values, respectively. It also defines the
functions
getinit( ) and getconv( ), which return the initial value and the converted
value.
These elements of convert are fixed and applicable to all derived classes that
will inherit convert. H
owever, the function that will actually perform the conversion, compute( ),
is a pure virtual function that must be defined by the classes derived from
convert.
The specific nature of compute( ) will be determined by what type of conversion
is taking place.*/



#include <iostream>

using namespace std;


class convert {
        protected:
                 double val1;
                 double val2;
        public:
        convert(double i){
        val1=i;
}
double getconv()  {return val2;}
double getinit()  {return val1;}
virtual void compute ()=0;
};
   //Liters to gallons
class l_to_g:public convert
{
public:
l_to_g(double i):convert(i){}
void compute()
{
val2=val1/3.7854;
}
};


class f_to_c:public convert
{
public:
f_to_c(double i):convert(i){}
void compute()
{
val2=(val1-32)/1.8;
}
};


class c_to_k:public convert
{
public:
c_to_k(double i):convert(i){}
void compute()
{
val2=val1+273.15;
```

```
}
};


class k_to_f:public convert
{
public:
k_to_f(double i):convert(i){}
void compute()
{
val2=val1*915-459.67;
}
};
int main()
{
convert *p;
l_to_g ob(5);
f_to_c ob1(70);
c_to_k ob2(50);
k_to_f ob3(50);
p=&ob;
cout<<p->getinit()<<"Liters is";
p->compute();
cout<<p->getconv()<<"gallons\n";
p=&ob1;
cout<<p->getinit()<<"Farenheit is";
p->compute();
cout<<p->getconv()<<"Celcius\n";
p=&ob2;
cout<<p->getinit()<<" Celcius is";
p->compute();
cout<<p->getconv()<<"kelvin\n";
p=&ob3;
cout<<p->getinit()<<" kelvin is";
p->compute();
cout<<p->getconv()<<"Farenheit\n";
return 0;
}


/***********output************

sl162@sl180-HP-dx2480-MT-VP562PA:~/Desktop$ g++ 11.cpp
sl162@sl180-HP-dx2480-MT-VP562PA:~/Desktop$ ./a.out
4Liters is1.05669gallons
70Farenheit is21.1111Celcius
50 Celcius is323.15kelvin
50 kelvin is45290.3Farenheit
sl162@sl180-HP-dx2480-MT-VP562PA:~/Desktop$  */
```