Lab 3

Title: Object as an function arguments

Objective:

• To be familiar with how to pass an object as an argument and how to return object

• To solve problems by passing object as an argument

Theory:

Object can be passed in two different ways:
1. **Pass-by-value** where a copy of the entire object is passed to the function. Here, any changes made to the formal object parameters do not affect the actual object parameters.
2. **Pass-by reference** where only the address of the object is transferred to the function. Here, any changes made to the formal object parameters will be reflected in the actual object parameters also.

1. Write a Program to find sum of two complex number by passing object as an argument.

```
#include<iostream>
using namespace std;
class complex{
        private:
                int real,imag;
        public:
                void getcomplex(){
                        cout<<"Enter the real part"<<endl;
                        cin>>real;
                        cout<<"Enter the imaginary part"<<endl;
                        cin>>imag;
                }
                void display(){
                        cout<<real<<"+"<<imag<<"i"<<endl;
                }
                void addcomplex(complex c1,complex c2){
                        real=c1.real+c2.real;
                        imag=c1.imag+c2.imag;
                }
};
int main(){
        complex c1,c2,c3;
        c1.getcomplex();
        c2.getcomplex();
        c3.addcomplex(c1,c2);
        c3.display();
        return 0;
}
```

2. Write a Program to find sum of two complex number by passing and returning object as an argument.

```
#include<iostream>
using namespace std;
class complex{
        private:
                int real,imag;
        public:
                void getcomplex(){
                        cout<<"Enter the real part"<<endl;
                        cin>>real;
                        cout<<"Enter the imaginary part"<<endl;
                        cin>>imag;
                }
                void display(){
                        cout<<real<<"+"<<imag<<"i"<<endl;
                }
                complex addcomplex(complex c1,complex c2){
                        complex temp;
                        temp.real=c1.real+c2.real;
                        temp.imag=c1.imag+c2.imag;
                        return temp;
                }
};
int main(){
        complex c1,c2,c3,result;
        cout<<"for c1"<<endl;
        c1.getcomplex();
        cout<<"for c2"<<endl;
        c2.getcomplex();
        result=c3.addcomplex(c1,c2);
        cout<<"the sum of two complex number is"<<endl;
        result.display();
        return 0;
}
```

3. Write a program to find sum of two complex number, function call by one object passing second object as function argument and return third object adding two objects. Hint:
c3=c1.addComplex(c2);

```
#include<iostream>
using namespace std;
class complex{
        private:
                int real,imag;
        public:
                void getcomplex(){
                        cout<<"Enter the real part"<<endl;
                        cin>>real;
                        cout<<"Enter the imaginary part"<<endl;
                        cin>>imag;
                }
                void display(){
                        cout<<real<<"+"<<imag<<"i"<<endl;
                }
                complex addcomplex(complex c2){
                        complex temp;
                        temp.real=real+c2.real;
                        temp.imag=imag+c2.imag;
                        return temp;
                }
};
int main(){
        complex c1,c2,c3;
        cout<<"for c1"<<endl;
        c1.getcomplex();
        cout<<"for c2"<<endl;
        c2.getcomplex();
        c3=c1.addcomplex(c2);
        cout<<"the sum of two complex number is"<<endl;
        c3.display();
        return 0;
}
```

4. Perform similar operation in above(Que no. 1,2,3) for
   i. Addition of two times with data members hours, minutes and seconds
      a. By passing object as an argument.

```
#include<iostream>
using namespace std;
class time{
    private:
        int hrs,mins,sec;
    public:
        void gettime(){
            cout<<"Enter the hrs"<<endl;
            cin>>hrs;
            cout<<"Enter the mins"<<endl;
            cin>>mins;
            cout<<"Enter the second"<<endl;
            cin>>sec;
        }
        void display(){
            cout<<hrs<<"hrs:"<<mins<<"Mins:"<<sec<<"Sec";
        }
        void addtime(time t1,time t2){
            sec=t1.sec+t2.sec;
            mins=t1.mins+t2.mins;
            hrs=mins/60;
            mins=mins%60+sec/60;
            sec=sec%60;
            hrs=hrs+t1.hrs+t2.hrs;
        }
};
   int main(){
     time t1,t2,t3;
     cout<<"for t1"<<endl;
     t1.gettime();
     cout<<"for t2"<<endl;
     t2.gettime();
     t3.addtime(t1,t2);
     cout<<"sum of two time"<<endl;
     t3.display();
     return 0;
     }
```

b. By passing and returning object as an argument.

```
#include<iostream>
using namespace std;
class time{
    private:
        int hrs,mins,sec;
    public:
        void gettime(){
            cout<<"Enter the hrs"<<endl;
            cin>>hrs;
            cout<<"Enter the mins"<<endl;
            cin>>mins;
            cout<<"Enter the second"<<endl;
            cin>>sec;
        }
        void display(){
            cout<<hrs<<"hrs:"<<mins<<"Mins:"<<sec<<"Sec";
        }
        time addtime(time t1,time t2){
           time temp;
            temp.sec=t1.sec+t2.sec;
            temp.mins=t1.mins+t2.mins;
            temp.hrs=temp.mins/60;
            temp.mins=temp.mins%60+temp.sec/60;
            temp.sec=temp.sec%60;
            temp.hrs=temp.hrs+t1.hrs+t2.hrs;
            return temp;
        }
};
   int main(){
     time t1,t2,t3,result;
     cout<<"for t1"<<endl;
     t1.gettime();
     cout<<"for t2"<<endl;
     t2.gettime();
        result=t3.addtime(t1,t2);
        cout<<"sum of two time"<<endl;
        result.display();
        return 0;
     }
```

c. By one object passing second object as function argument and return third object adding two objects.

```
#include<iostream>
using namespace std;
class time{
    private:
        int hrs,mins,sec;
    public:
        void gettime(){
            cout<<"Enter the hrs"<<endl;
            cin>>hrs;
            cout<<"Enter the mins"<<endl;
            cin>>mins;
            cout<<"Enter the second"<<endl;
            cin>>sec;
        }
        void display(){
            cout<<hrs<<"hrs:"<<mins<<"Mins:"<<sec<<"Sec";
        }
        time addtime(time t2){
          time temp;
            temp.sec=sec+t2.sec;
            temp.mins=mins+t2.mins;
            temp.hrs=temp.mins/60;
            temp.mins=temp.mins%60+temp.sec/60;
            temp.sec=temp.sec%60;
            temp.hrs=temp.hrs+hrs+t2.hrs;
            return temp;
        }
};
  int main(){
    time t1,t2,t3,result;
    cout<<"for t1"<<endl;
    t1.gettime();
    cout<<"for t2"<<endl;
    t2.gettime();
        t3=t1.addtime(t2);
        cout<<"sum of two time"<<endl;
        t3.display();
        return 0;
    }
```

ii. Addition of two height with data members feet and inches

a. By passing object as an argument.

```
#include<iostream>
using namespace std;
class height{
        private:
        int feet,inchs;
        public:
        void getheight(){
                cout<<"Enter the feet"<<endl;
                cin>>feet;
                cout<<"Enter the inch"<<endl;
                cin>>inchs;
        }
        void display(){
                cout<<feet<<"feets:"<<inchs<<"inch";
        }
        void addheight(height h1,height h2){
                inchs=h1.inchs+h2.inchs;
                feet=inchs/12+h1.feet+h2.feet;
                inchs=inchs%12;
        }
};
int main(){
        height h1,h2,h3;
        cout<<"for h1"<<endl;
        h1.getheight();
        cout<<"for h2"<<endl;
        h2.getheight();
        h3.addheight(h1,h2);
        h3.display();
        return 0;
}
```

b. By passing and returning object as an argument

```
#include<iostream>
using namespace std;
class height{
        private:
        int feet,inchs;
        public:
        void getheight(){
                cout<<"Enter the feet"<<endl;
                cin>>feet;
                cout<<"Enter the inch"<<endl;
                cin>>inchs;
        }
        void display(){
                cout<<feet<<"feets:"<<inchs<<"inch";
        }
        height addheight(height h1,height h2){
                height temp;
                temp.inchs=h1.inchs+h2.inchs;
                temp.feet=temp.inchs/12+h1.feet+h2.feet;
                temp.inchs=temp.inchs%12;
                return temp;
        }
};
int main(){
        height h1,h2,h3,result;
        cout<<"for h1"<<endl;
        h1.getheight();
        cout<<"for h2"<<endl;
        h2.getheight();
        result=h3.addheight(h1,h2);
        result.display();
        return 0;
}
```

c. By one object passing second object as function argument and return third object adding two objects.

```
#include<iostream>
using namespace std;
class height{
    private:
    int feet,inchs;
    public:
    void getheight(){
            cout<<"Enter the feet"<<endl;
            cin>>feet;
            cout<<"Enter the inch"<<endl;
            cin>>inchs;
    }
    void display(){
            cout<<feet<<"feets:"<<inchs<<"inch";
    }
    height addheight(height h2){
            height temp;
            temp.inchs=inchs+h2.inchs;
            temp.feet=temp.inchs/12+feet+h2.feet;
            temp.inchs=temp.inchs%12;
            return temp;
    }
};
int main(){
    height h1,h2,h3,result;
    cout<<"for h1"<<endl;
    h1.getheight();
    cout<<"for h2"<<endl;
    h2.getheight();
    h3=h1.addheight(h2);
    h3.display();
    return 0;
}
```