1. Define a class to represent bowlers in a cricket team with data members name, Overs bowled, Runs given, Wickets taken and use constructor to initialize values and use member function to display bowlers information.

```
#include<iostream>
#include<cstring>
using namespace std;
class bowler{
        private:
                char name[20];
                float over;
                int run;
                int wicket;
        public:
                bowler(){
                strcpy(name,"curry ");
                over=5;
                run=100;
                wicket=5;}
void display(){
        cout<<"Bowler Name:"<<name<<endl;
        cout<<"over:"<<over<<endl;
        cout<<"run:"<<run<<endl;
        cout<<"Wicket:"<<wicket<<endl;
}
};
int main(){
        bowler b1;
        b1.display();
        return 0;
}
```

2. Create a class called employee with data member Code, Name, address, salary. Create a constructor to initialize the member of the class. Also create the another constructor so that we can create an object from another object. Define member function display() to display the information of the class.

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

class Employee {
private:
   int Code;
   char Name[20];
   char address[20];
   float Salary;

public:
   Employee(int c, char n[], char a[], float s) {
      Code = c;
      strcpy(Name,n);
      strcpy(address,a);
      Salary = s;
   }
   Employee(const Employee &e) {
      Code = e.Code;
      strcpy(Name,e.Name);
      strcpy(address,e.address);
      Salary = e.Salary;
   }
    void display() {
      cout << "Employee Details:\n";
      cout << "Code    : " << Code << endl;
      cout << "Name    : " << Name << endl;
      cout << "Address : " << address << endl;
      cout << "Salary  : " << Salary << endl;
   }
};
int main() {
   Employee emp1(101, "LeBron", "USA ", 5455);
   emp1.display();
   Employee emp2(emp1);
   emp2.display();

   return 0;
}
```

3. Create a class time constructor having hour, minute and second as an arguments is use to take two time data from user. The add function that takes two class objects an arguments add them respectively then display the aggregate result?

```
#include<iostream>
using namespace std;
class time{
        private:
                int hours;
                int minute;
                int second;
        public:
                time(){

                };
                time(int h,int m,int s){
                        hours=h;
                        minute=m;
                        second=s;}
                void display(){
                        cout<<hours<<":"<<minute<<":"<<second<<":"<<endl;
                }
                void sum(time,time);
};
   void time::sum(time t1,time t2){
        second=t1.second+t2.second;
        minute=second/60+t1.minute+t2.minute;
        hours=minute/60+t1.hours+t2.hours;
        minute=minute%60;
        second=second%60;}
        int main(){
                int hrs1,hrs2,min1,min2,sec1,sec2;
                cout<<"Enter the hour,minute,second for time 1"<<endl;
                cin>>hrs1>>min1>>sec1;
                cout<<"Enter the hour,minute,second for time 2"<<endl;
                cin>>hrs2>>min2>>sec2;
                time t1(hrs1,min1,sec1);
                time t2(hrs2,min2,sec2);
                time t3;
                t1.display();
                t2.display();
                t3.sum(t1,t2);
                t3.display();
                return 0;
}
```

4. Create a class person with data members name, age, address and citizenship number. Write a constructor to initialize the value of a person. Assign citizenship number if the age of the person is greater than 16 otherwise assign zero to citizenship number. Also create a function to display values.

```
#include<iostream>
#include<cstring>
using namespace std;
class person{
        private:
                char name[20];
                int age;
                char address[20];
                long int ctno;
        public:
                person(char n[],int a,char A[],long int ct){
                        strcpy(name,n);
                        age=a;
                        strcpy(address,A);
                        ctno=ct;
                }
                void display(){
                        cout<<"Name:"<<name<<endl;
                        cout<<"age:"<<age<<endl;
                        cout<<"Address:"<<address<<endl;
                        cout<<"citizen number:"<<ctno<<endl;
                }
};
   int main(){
        char name[20];
        int age;
        char address[20];
        long int ctno=0;
        cout<<"Enter the name"<<endl;
        cin>>name;
        cout<<"Enter the ade"<<endl;
        cin>>age;
        cout<<"Enter the address"<<endl;
        cin>>address;
        if(age>16){
        cout<<"Enter the citizenship number"<<endl;
                cin>>ctno;}
                person p(name,age,address,ctno);
                p.display();
```

```
        return 0;
}
```

5. Write a Program to add two complex number using the concept of Constructor overloading.

```
#include<iostream>
using namespace std;
class complex{
    private:
        int real;
        int img;
    public:
        complex(){

        }
        complex(int r,int i){
            real=r;
            img=i;
        }
        void display(){
            cout<<real<<"+i"<<img<<endl;
        }
        void sum(complex,complex);
        };
        void complex::sum(complex c1,complex c2){
            real=c1.real+c2.real;
            img=c1.img+c2.img;
        }
    int main(){
        int i1,i2,r1,r2;
        cout<<"Enter the 1 st complex number"<<endl;
        cin>>r1>>i1;
        cout<<"Enter the 2nd complex number"<<endl;
        cin>>r2>>i2;
        complex c1(r1,i1);
        complex c2(r2,i2);
        complex c3;
        c1.display();
        c2.display();
        c3.sum(c1,c2);
        c3.display();
        return 0;
    }
```

6. Create a class Mountain with data members name, height, location, a constructor that initializes the members to the values passed it to its parameters, a function called CmpHeight() to compare two objects and DispInf() to display the information of mountain. In main create two objects of the class mountain and print the information of the mountain which is greatest height.

```
#include<iostream>
#include<string.h>
using namespace std;
class mountain{
        private:
                char name[20];
                float height;
                char location[20];
        public:
                mountain(char n[],float h,char l[]){
                        strcpy(name,n);
                        height=h;
                        strcpy(location,l);
                }
                void Dispnf(){
                        cout<<"Name of mountain"<<name<<endl;
                        cout<<"height of mountain"<<height<<endl;
                        cout<<"location of mountain"<<location<<endl;
                }
                friend void cmpheight(mountain m1,mountain m2);
};
   void cmpheight(mountain m1,mountain m2){
        if(m1.height>m2.height){
                m1.Dispnf();
                }
                else{
                        m2.Dispnf();
                }
        }
        int main(){
                char name1[20],name2[20];
                float height1,height2;
                char location1[20],location2[20];
                cout<<"Enter the name,height,location"<<endl;
                cin>>name1>>height1>>location1;
                cout<<"Enter the name,height,location"<<endl;
                cin>>name2>>height2>>location2;
                mountain m1(name1,height1,location1);
                mountain m2(name2,height2,location2);
```

```
        cmpheight(m1,m2);
        return 0;
}
```