Friend Function

Friend functions of the class are granted permission to access private and protected members of the class in C++. They are defined globally outside the class' scope. Friend functions are not member functions of the class. So, what exactly is the friend function?

A friend function is a function that is declared outside a class, but is capable of accessing the private and protected members of class . There could be situations in programming wherein we want two classes to share their members. These members may be data members, class functions or function templates . In such cases, we make the desired function, a friend to both these classes which will allow to access private and protected data members of the class.

Generally, non-member functions cannot access the private members of a particular class. Once declared as a friend function, the function is able to access the private and the protected members of these classes.

❖ Friend Function Syntax:

```
class className {
    ... .. ...
    friend returnType functionName(arguments);
    ... .. ...
}
```

❖ Declaration of a friend function in C++

```
class class_name
{
    friend data_type function_name(arguments/s);
};
```

❖ Program 1:

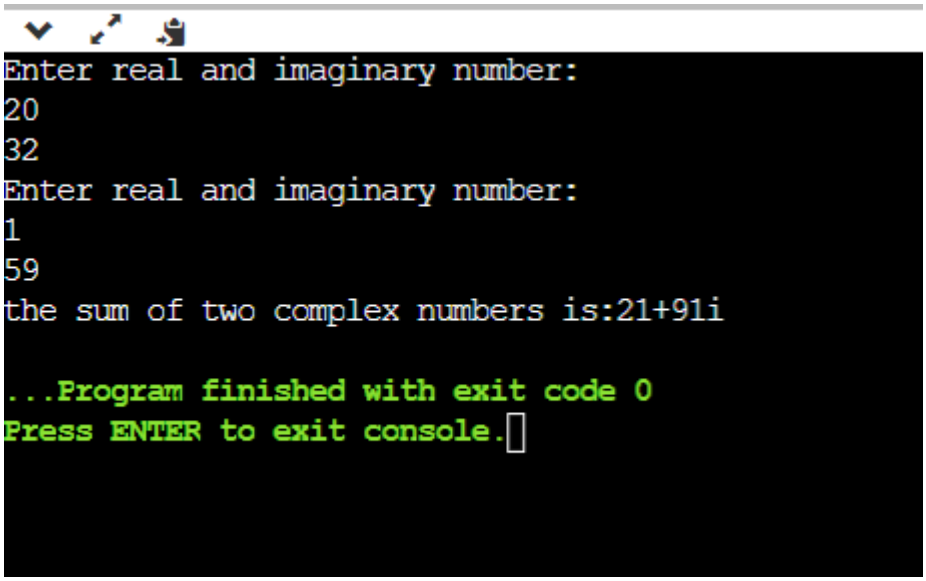
Write a program to add two complex numbers using friend function.

Code:

```
#include <iostream>
using namespace std;
class complex_number
{
   int real, imaginary;
   public:
   void getdata ()
   {
      cout<<"Enter real and imaginary number:"<<"\n";
      cin>>real>>imaginary;
   }
  friend void add (complex_number, complex_number);
  };

void add (complex_number a, complex_number b)
{
   complex_number t;
   t.real = a.real + b.real;
   t.imaginary = a.imaginary + b.imaginary;
   cout<<"the sum of two complex numbers is:"<<t.real<<"+"<<t.imaginary<<"i";
}
int main()
{
   complex_number a, b;
```

```
    a.getdata();
    b.getdata();
    add ( a, b);
 return 0;
}
```

Output:



❖ Program 2:

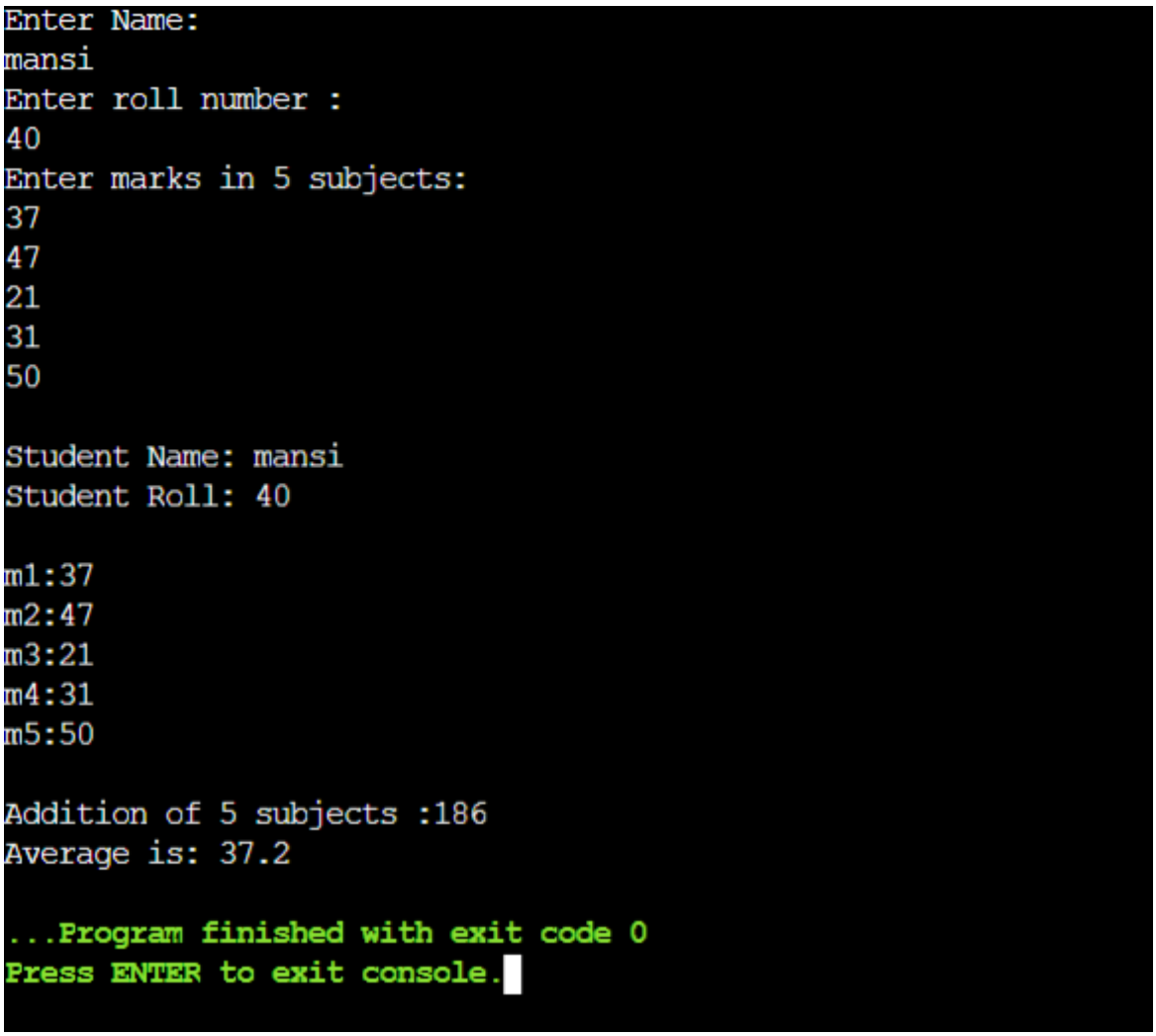
Write a class with data members name, roll no, marks in 5 subjects.
Write a friend Function which is calculating average of 5 subjects.
Display all details of students with average marks.

Code:

```
#include <iostream>
using namespace std;
class student
{
   int rollno,marks[5];
   char name[10];
    public:
   void getdata()
   {
      cout<<"Enter Name:"<<"\n";
      cin>>name;
      cout<<"Enter roll number :"<<"\n";
      cin>>rollno;
      cout<<"Enter marks in 5 subjects: "<<"\n";
      for(int i=0;i<5;i++)
{
         cin>>marks[i];
      }
   }
    friend void avg(student s );
};

void avg(student s){
```

```
   cout<<"\nStudent Name: "<<s.name;
   cout<<"\nStudent Roll: "<<s.rollno;

   cout<<"\n\nm1:"<<s.marks[0] ;
   cout<<"\nm2:"<<s.marks[1] ;
   cout<<"\nm3:"<<s.marks[2] ;
   cout<<"\nm4:"<<s.marks[3] ;
   cout<<"\nm5:"<<s.marks[4]<<"\n\n" ;

   float avg_var= s.marks[0] +s.marks[1] + s.marks[2] + s.marks[3] +s.marks[4];
   cout<<"Addition of 5 subjects :"<<avg_var<<"\n";
   float avg=avg_var/5;
   cout<<"Average is: "<<avg;
}
int main()
{
   student s;
   s.getdata();
   avg(s);
 return 0;
}
```

Output:


Enter Name:
mansi
Enter roll number :
40
Enter marks in 5 subjects:
37
47
21
31
50
Student Name: mansi
Student Roll: 40
m1:37
m2:47
m3:21
m4:31
m5:50
Addition of 5 subjects :186
Average is: 37.2
...Program finished with exit code 0
Press ENTER to exit console.