Experiment 5

AIM: Write a program to accept 5 different numbers by creating a class called friendfunc1 and friendfunc2 taking 2 and 3 arguments respectively and calculate the average of these numbers by passing an object of the class to the friend function.

Theory:

Friend Function Like friend class, a friend function can be given special grant to access private and protected members. A friend function can be

a) A method of another class
b) A global function

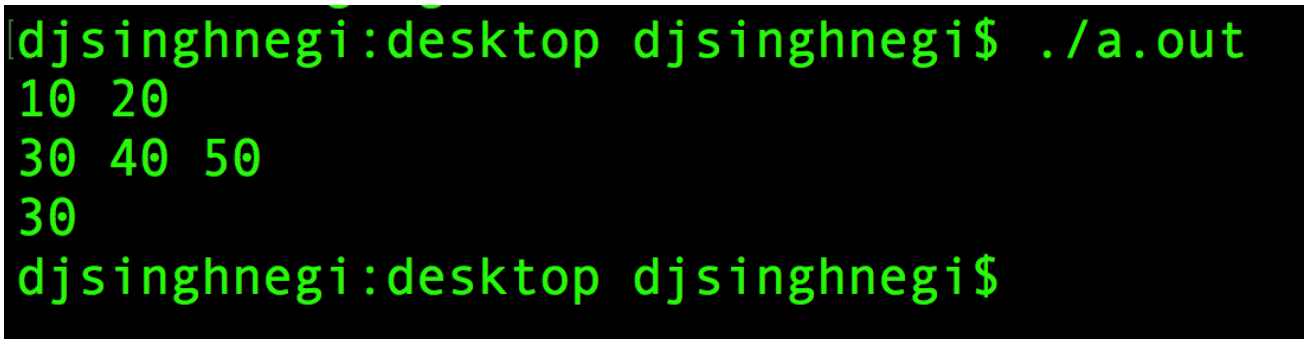
Code:

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

class friendfunc1;
class friendfunc2;

class friendfunc1 {
    int a, b;
    public: friendfunc1() {
        cin >> a >> b;
    }
    friend float avg(friendfunc1, friendfunc2);
};
class friendfunc2 {
    int a, b, c;
    public:
    friendfunc2() {
        cin >> a >> b >> c;
    }
    friend float avg(friendfunc1, friendfunc2);
};
float avg(friendfunc1 f1, friendfunc2 f2) {
    return (f1.a + f1.b + f2.a + f2.b + f2.c) / 5.0;
}
int main() {
    friendfunc1 a;
    friendfunc2 b;
    cout << avg(a, b) << endl;
    return 0;
}
```

Output:



Discussion:

The class friendfunc1 and friendfunc2 have have 2 and 3 data members each, respectively. The function avg is a friend of both these class and can hence access private members of the objects of these classes. Just like primitive data types, user defined data types like objects can be passed in as arguments in functions.