

Oop lab 7
2005839
T. DAMAN

Q1.

```
//wap to overload following operators for class complex which stores complex number(s) as object(s)
//overload binary / to find the division of two objects(using friend fnction)
//2005839 T.DAMAN
#include <iostream>
using namespace std;
class complex
{
    float real;
    float imag;
public:
    friend complex operator*(const complex &c1, const complex &c2);
    void display_numbers()
    {
        cout << "[" << real << "," << imag << "]" << endl;
    }
    void set_data()
    {
        cout << "Enter real part : " << endl;
        cin >> real;
        cout << "Enter imaginary part : " << endl;
        cin >> imag;
    }
};
complex operator*(const complex &c1, const complex &c2)
{
    complex c3;
    c3.real = ((c1.real * c2.real) + (c1.imag * c2.imag)) / ((c2.real * c2.real) - (c2.imag * c2.imag));
    c3.imag = (c1.imag * c2.real) - (c1.real * c2.imag) / ((c2.real * c2.real) - (c2.imag * c2.imag));
    return c3;
}
int main()
{
    complex c1, c2;
    c1.set_data();
    cout << "Enter second complex no : " << endl;
    c2.set_data();
    complex c3(c1 * c2);
    c3.display_numbers();
    return 0;
}
```

Output:-

Enter real part :

1

Enter imaginary part :

2

Enter second complex no :

Enter real part :

3

Enter imaginary part :

4

[-1.57143,6.57143]

Q2.

```
//overload binary * to find multiplication of two complex no.
//2005839 T.DAMAN
#include <iostream>
using namespace std;
class complex
{
    double real;
    double imag;
public:
    complex operator*(complex c)
    {
        double real1, real2;
        real1 = real;
        real2 = c.real;
        real = (real * c.real) - (imag * c.imag);
        imag = (real1 * c.imag) + (imag * real2);
        complex temp;
        temp.real = real;
        temp.imag = imag;
        return temp;
    }
    void display() { cout << "the resultant comlex no. is " << real << " + " << imag << "i" << endl; }
    void set()
    {
        cout << "Enter real no.: " << endl;
        cin >> real;
        cout << "Enter imag no.: " << endl;
        cin >> imag;
    }
};
int main()
{
    complex c1, c2;
    c1.set();
    cout << "Enter 2nd complex no: " << endl;
    c2.set();
    c1 * c2;
    c1.display();
    return 0;
}
```

Output:

Enter real no.:

1

Enter imag no.:

2

Enter 2nd complex no:

Enter real no.:

3

Enter imag no.:

4

the resultant complex no. is $-5 + 10i$

Q3.

```
//create a class to store an array overload insertion and extraction operator to input and display the
elements
//2005839 T.DAMAN
#include <bits/stdc++.h>
using namespace std;

class Array
{
    int n;
    int *arr;
public:
    Array()
    {
        n = 0;
    }
    Array(int size_150)
    {
        n = size_150;
        arr = new int[n];
    }
    friend istream &operator>>(istream &in, Array a);
    friend ostream &operator<<(ostream &out, Array a);
};

istream &operator>>(istream &in, Array a)
{
    cout << "Enter elements of array :\n";
    for (int i = 0; i < a.n; i++)
    {
        in >> a.arr[i];
    }
    cout << endl;
    return in;
}

ostream &operator<<(ostream &out, Array a)
{
    cout << "Elements of array are :\n";
    for (int i = 0; i < a.n; i++)
    {
        out << a.arr[i] << " ";
    }
    cout << endl;
    return out;
}

int main()
{
    int n;
    cout << "Enter size of array" << endl;
    cin >> n;
    Array Ar(n);
    cin >> Ar;
    cout << Ar;
    return 0;
}
```

Output:

Enter size of array

6

Enter elements of array :

1
3
5
7
9
0

Elements of array are :

1 3 5 7 9 0

Q4.

```
//create a class that contain two object of float values compare two objects
2005839 T.DAMAN
#include <iostream>
using namespace std;

class number {
private:
    int num1;
public:
    number( ){
        int k;
        cout<<"enter a number"<<endl;
        cin>>k;
        num1 = k;
    }

    bool operator <=(const number& d) {

        if(num1 == d.num1 || num1 < d.num1 ) {
            return true;
        }
        return false;
    }
    bool operator >=(const number& d) {

        if(num1 == d.num1 || num1 >d.num1 ) {
            return true;
        }
        return false;
    }
    bool operator ==(const number& d) {

        if(num1 == d.num1 ) {
            return true;
        }
        return false;
    }
    bool operator >(const number& d) {

        if( num1 >d.num1 ) {
            return true;
        }
        return false;
    }
    bool operator <(const number& d) {
```

```

        if( num1 < d.num1 ) {
            return true;
        }
        return false;
    }
    bool operator !=(const number& d) {

        if(num1 != d.num1 ) {
            return true;
        }
        return false;
    }
};

int main() {
    number D1, D2;

    if( D1 <= D2 ) {
        cout << "D1 is less than or equal to D2 " << endl;
    } else {
        cout << "D2 is less than D1 " << endl;
    }
    if( D1 >= D2 ) {
        cout << "D1 is greater than or equal to D2 " << endl;
    } else {
        cout << "D2 is greater than D1 " << endl;
    }
    if( D1 == D2 ) {
        cout << "D1 is equal to D2 " << endl;
    }
    else if( D1 != D2 ) {
        cout << "D1 is not equal to D2 " << endl;
    }
    else if( D1 > D2 ) {
        cout << "D1 is greater ttha D2 " << endl;
    }
    else if( D1 < D2 ) {
        cout << "D1 is less than D2 " << endl;
    }
    return 0;
}

```

Output:

enter a number

14

enter a number

16

D1 is less than or equal to D2

D2 is greater than D1

D1 is not equal to D2