

```

#include<stdio>

class complejo
{
private:

double x,y;

public:

complejo();

complejo(double , double );

complejo(const complejo &);

const complejo& operator+ () const;

const complejo operator-();

const complejo& operator++();

const complejo operator++ (int);

complejo& operator = (const complejo &);

friend const complejo operator+ (const complejo & , const complejo &);

friend const complejo operator+ (const complejo & , float);

friend const complejo operator+ (float , const complejo &);

void emitir(){cout<<"("<<x<<" "<<y<<"")<<endl;}
};

complejo ::complejo () : x(0.0) , y(0.0) {}

complejo ::complejo (const complejo & c) : x(c.x) , y (c.y) {}

complejo ::complejo (double a, double b): x(a) , y(b) {}

const complejo & complejo :: operator + () const
{ return *this;}

const complejo complejo:: operator -()
{
complejo s;
s.x= -x ; s.y =-y ;
return s;
}

const complejo &complejo :: operator ++ ()
{ x++ ; y++; return *this;}

```

```

const complejo complejo::operator ++ (int)
{complejo anterior (x,y);
x++ ; y++ ;
return anterior;
}

```

```

complejo &complejo ::operator= (const complejo & b)
{ x = b.x;
y= b.y;
return *this;
}

```

```

const complejo operator + (const complejo &l, const complejo &r)
{return complejo (l.x+r.x , l.y+r.y);}

```

```

const complejo operator+ (const complejo &l , float f)
{ return complejo (l.x+f,l.y);}

```

```

const complejo operator+ (float f , const complejo &l)
{return complejo (f+l.x , l.y);}

```

```

void main()
{
complejo a, b(2,3) , c(b);
cout<<"resultado: a vale"<<endl;
a.emitir();
cout<<"resultado: b vale"<<endl;
b.emitir();
cout<<"resultado: c vale"<<endl;
c.emitir();
cout<<"a=b+c"<<endl;
a=b+c;
cout<<"resultado: a vale"<<endl;
a.emitir();
cout<<"b=-a"<<endl;
b=-a;
cout<<"resultado: b vale"<<endl;
b.emitir();
cout<<"resultado: a vale "<<endl;
a.emitir();
cout<<"a++"<<endl;
a++;
cout<<"resultado"<<endl;
a.emitir();

cout<<"suma el real 3 y el complejo b"<<endl;
a=3+b;
a.emitir();

cout<<"suma el complejo b y el real 5"<<endl;
a=b+5;
a.emitir();

}

```