```
/* The following code example is taken from the book
 * "The C++ Standard Library - A Tutorial and Reference, 2nd Edition"
 * by Nicolai M. Josuttis, Addison-Wesley, 2012
 *
 * (C) Copyright Nicolai M. Josuttis 2012.
 * Permission to copy, use, modify, sell and distribute this software
 * is granted provided this copyright notice appears in all copies.
 * This software is provided "as is" without express or implied
 * warranty, and with no claim as to its suitability for any purpose.
 */
#include <iostream>
#include <complex>
using namespace std;

int main()
{
    // complex number with real and imaginary parts
    // - real part: 4.0
    // - imaginary part: 3.0
    complex<double> c1(4.0,3.0);

    // create complex number from polar coordinates
    // - magnitude: 5.0
    // - phase angle: 0.75
    complex<float> c2(polar(5.0,0.75));

    // print complex numbers with real and imaginary parts
    cout << "c1: " << c1 << endl;
    cout << "c2: " << c2 << endl;

    // print complex numbers as polar coordinates
    cout << "c1: magnitude: " << abs(c1)
         << " (squared magnitude: " << norm(c1) << ") "
         <<    " phase angle: " << arg(c1) << endl;
    cout << "c2: magnitude: " << abs(c2)
         << " (squared magnitude: " << norm(c2) << ") "
         <<    " phase angle: " << arg(c2) << endl;

    // print complex conjugates
    cout << "c1 conjugated:  " << conj(c1) << endl;
    cout << "c2 conjugated:  " << conj(c2) << endl;

    // print result of a computation
    cout << "4.4 + c1 * 1.8: " << 4.4 + c1 * 1.8 << endl;

    // print sum of c1 and c2:
    // - note: different types
    cout << "c1 + c2:        "
         << c1 + complex<double>(c2.real(),c2.imag()) << endl;

    // add square root of c1 to c1 and print the result
    cout << "c1 += sqrt(c1): " << (c1 += sqrt(c1)) << endl;
}
```