Tema 2

NICOLETA RADU – OOP C++

# Completarea Aplicatiei cu numere complexe

#include<iostream>

using namespace std;

class complex

{

private:

// real and imaginary parts, a(real) and b(imaginary)

double a, b;

public:

// complex number operations

void read();

void write();

complex sum(complex);

complex subtract(complex);

complex multiply(complex);

complex divide(complex);

private:

complex negativ();

complex conjugate();

double absolute();

};

void complex::read()

{

cout << "Real part of the numer = ";

cin >> a;

cout << "Imaginary part of the number = ";

cin >> b;

}

void complex::write()

{

cout << "Complex number is: " << a << "," << b << endl;

}

complex complex::sum(complex A)

{

complex rezultat;

rezultat.a = a + A.a;

rezultat.b = b + A.b;

return rezultat;

}

complex complex::negativ()

{

complex rezultat{};

rezultat.a = -a;

rezultat.b = -b;

return rezultat;

}

complex complex::subtract(complex A)

{

complex rezultat{};

rezultat = sum(A.negativ());

return rezultat;

}

complex complex::multiply(complex A)

{

complex rezultat{};

rezultat.a = a \* A.a - b \* A.b;

rezultat.b = a \* A.b - b \* A.a;

return rezultat;

}

double complex::absolute()

{

return sqrt(a \* a + b \* b);

}

complex complex::conjugate()

{

complex rezultat;

rezultat.a = a;

rezultat.b = -b;

return rezultat;

}

complex complex::divide(complex A)

{

complex rezultat;

complex n;

n = multiply(A.conjugate());

rezultat.a = n.a / pow(A.absolute(), 2);

rezultat.b = n.b / pow(A.absolute(), 2);

return rezultat;

}

**APEL**

/\*complex a, b;

a.read();

b.read();

a.write();

b.write();

complex rezultat;

rezultat = a.sum(b);

cout << "Suma: " << endl;

rezultat.write();

rezultat = a.divide(b);

cout << "Diferenta: " << endl;

rezultat.write();

rezultat = a.multiply(b);

cout << "Inmultire: " << endl;

rezultat.write();

rezultat = a.divide(b);

cout << "Impartire: " << endl;

rezultat.write();\*/

# Aplicatie cu fractii

#include<iostream>

using namespace std;

class Fractie

{

private:

// partile unei fractii, numitor si numarator

double numitor, numarator;

public:

void citire();

void scriere();

Fractie adunare(Fractie);

Fractie scadere(Fractie);

Fractie inmultire(Fractie);

Fractie impartire(Fractie);

Fractie putere(int);

private:

Fractie inversa();

};

void Fractie::citire()

{

cout << "Numitor: ";

cin >> numitor;

cout << "Numarator: ";

cin >> numarator;

}

void Fractie::scriere()

{

cout << "Fractia este: " << numarator << " / " << numitor << endl;

}

Fractie Fractie::inversa()

{

Fractie rezultat;

rezultat.numitor = numarator;

rezultat.numarator = numitor;

return rezultat;

}

Fractie Fractie::adunare(Fractie A)

{

Fractie rezultat;

double aux;

if (numitor > A.numitor)

{

aux = numitor / A.numitor;

rezultat.numitor = A.numitor \* aux;

}

else if (numitor < A.numitor)

{

aux = A.numitor / numitor;

rezultat.numitor = numitor \* aux;

}

rezultat.numarator = numarator + A.numarator;

return rezultat;

}

Fractie Fractie::scadere(Fractie A)

{

Fractie rezultat;

double aux;

if (numitor > A.numitor)

{

aux = numitor / A.numitor;

rezultat.numitor = A.numitor \* aux;

}

else if (numitor < A.numitor)

{

aux = A.numitor / numitor;

rezultat.numitor = numitor \* aux;

}

rezultat.numarator = numarator - A.numarator;

return rezultat;

}

Fractie Fractie::inmultire(Fractie A)

{

Fractie rezultat;

rezultat.numarator = numarator \* A.numarator;

rezultat.numitor = numitor \* A.numitor;

return rezultat;

}

Fractie Fractie::impartire(Fractie A)

{

Fractie rezultat;

rezultat = inmultire(A.inversa());

return rezultat;

}

Fractie Fractie::putere(int A)

{

Fractie rezultat;

rezultat.numitor = pow(numitor, A);

rezultat.numarator = pow(numarator, A);

return rezultat;

}

**APEL**

Fractie a{}, b{}, rezultat{};

a.citire();

b.citire();

a.scriere();

b.scriere();

cout << "Adunare" << endl;

rezultat = a.adunare(b);

rezultat.scriere();

cout << "Scadere" << endl;

rezultat = a.scadere(b);

rezultat.scriere();

cout << "Inmultire" << endl;

rezultat = a.inmultire(b);

rezultat.scriere();

cout << "Impartire" << endl;

rezultat = a.impartire(b);

rezultat.scriere();

cout << "Ridicare la putere cu 2" << endl;

rezultat = a.putere(2);

rezultat.scriere();

