**[เฉลย] Lab5-ลายเซ็นของเมธอด, การโอเวอร์โหลดของเมธอด**

1. จำนวนเชิงซ้อนเป็นจำนวนที่ประกอบด้วย 2 ส่วน คือ ส่วนจำนวนจริง (real part) และส่วนจำนวนจินตภาพ (imaginary part) เขียนแทนด้วย a+bi เมื่อ a เป็นจำนวนจริง และ bi เป็นจำนวนจินตภาพ

เช่น 5 + 2i มีจำนวนจริงคือ 5 และจำนวนจินตภาพ 2i

หรือ 4-3i มีจำนวนจริงคือ 4 และจำนวนจินตภาพ -3i

คุณสมบัติของจำนวนจินตภาพ เมื่อกำหนดให้ X = (a+bi) , Y = ( c + di) มีตัวดำเนินการดังนี้

1.1. การบวก X + Y = (a+c) + (b+d)i

1.2. การลบ X - Y = (a-c) + (b-d)i

1.3. การคูณ X\*Y = (( a\*c)-(b\*d)) + ((a\*d) + (b\*c))i

1.4. Conjugate ของ X คือ



1.5. การหาร (มีข้อควรระวัง ตัวหารต้องไม่เป็น 0)



1.6. คุณส่วนกลับจำนวนเชิงซ้อน เรียกว่า (Reciprocal) คำนวณได้ดังนี้



1.7. การปรับสเกล แทนด้วย k(a+bi) = ka + kbi เมื่อ k เป็นจำนวนจริงใด ๆ

**จงเขียนคลาส ComplexNumber ที่สามารถทำได้ดังนี้**

ComplexNumber c1 = new ComplexNumber(5.0,2.0); // c1 = 5.0 + 2.0i

ComplexNumber c2 = new ComplexNumber(4.0,-3.0); // c2 = 4.0 - 3.0i

String c1str = c1.getComplex2String(); // “5.0 + 2.0i”

String c2str = c2.getComplex2String(); // “4.0 - 3.0i”

ComplexNumber c3,c4,c5,c6; //

c3.assign(ComplexNumber.add( c1,c2)); // จะเท่ากับ c3=c1+c2 จะมีค่า 9- 1i

c4.assign( ComplexNumber.subtract( c1,c2)); // จะเท่ากับ c4=c1-c2 จะมีค่า 1+ 5i

c5.assign( ComplexNumber.multiply( c1,c2)); // จะเท่ากับ c5=c1\*c2

c6.assign( ComplexNumber.divide( c1,c2)); // จะเท่ากับ c6=c1/c2

//------------------------------------------------------------------------------------------------

public class ComplexNumber {

//a+bi

private Double realpart; //real

private Double imaginarypart; //imaginary i

public ComplexNumber() {

this.realpart = 0.0;

this.imaginarypart = 0.0;

}

public ComplexNumber(Double realpart, Double imaginarypart) {

this.realpart = realpart;

this.imaginarypart = imaginarypart;

}

public String getComplex2String() {

//"a+bi"

String a = this.realpart.toString();

String b = this.imaginarypart.toString();

if (this.imaginarypart < 0.0)

return a.concat(b).concat("i");

else

return a.concat("+".concat(b).concat("i"));

}

public void assign(ComplexNumber c) {

this.realpart = c.getRealpart();

this.imaginarypart = c.getImaginarypart();

}

public static ComplexNumber addComplexNumber(ComplexNumber c1, ComplexNumber c2) {

ComplexNumber tmp = new ComplexNumber();

tmp.realpart = c1.realpart + c2.realpart;

tmp.imaginarypart = c1.imaginarypart + c2.imaginarypart;

return tmp;

}

public static ComplexNumber subtractComplexNumber(ComplexNumber c1, ComplexNumber c2) {

ComplexNumber tmp = new ComplexNumber();

tmp.realpart = c1.realpart - c2.realpart;

tmp.imaginarypart = c1.imaginarypart - c2.imaginarypart;

return tmp;

}

public static ComplexNumber multiplyComplexNumber(ComplexNumber c1, ComplexNumber c2) {

ComplexNumber tmp = new ComplexNumber();

tmp.realpart = (c1.realpart \* c2.realpart) - (c1.imaginarypart \* c2.imaginarypart);

tmp.imaginarypart = (c1.realpart \* c2.imaginarypart) + (c1.imaginarypart \* c2.realpart);

return tmp;

}

public static ComplexNumber divideComplexNumber(ComplexNumber c1, ComplexNumber c2) {

if (c2.realpart == 0.0 && c2.imaginarypart == 0.0) {

System.out.println("Error! divide by zero");

return null;

}

else {

ComplexNumber recp = reciprocalComplexNumber(c2);

ComplexNumber tmp = multiplyComplexNumber(c1, recp);

return tmp;

}

}

public static ComplexNumber conjugateComplexNumber(ComplexNumber c) {

c.imaginarypart = -c.imaginarypart;

return c;

}

public static ComplexNumber reciprocalComplexNumber(ComplexNumber c) {

ComplexNumber tmp = new ComplexNumber();

ComplexNumber conj = conjugateComplexNumber(c);

ComplexNumber mulp = multiplyComplexNumber(c, conj);

tmp.realpart = conj.realpart / mulp.realpart;

tmp.imaginarypart = conj.imaginarypart / mulp.realpart;

return tmp;

}

public double getRealpart() {

return this.realpart;

}

public void setRealpart(double realpart) {

this.realpart = realpart;

}

public double getImaginarypart() {

return this.imaginarypart;

}

public void setImaginarypart(double imaginarypart) {

this.imaginarypart = imaginarypart;

}

}

//------------------------------------------------------------------------------------------------

public class Main {

public static void main(String[] args) {

ComplexNumber c1 = new ComplexNumber(5.0, 2.0); // c1 = 5.0 + 2.0i

ComplexNumber c2 = new ComplexNumber(4.0, -3.0); // c2 = 4.0 - 3.0i

String c1str = c1.getComplex2String(); // “5.0 + 2.0i”

String c2str = c2.getComplex2String(); // “4.0 - 3.0i”

System.out.println(c1str);

System.out.println(c2str);

ComplexNumber c3 = new ComplexNumber();

ComplexNumber c4 = new ComplexNumber();

ComplexNumber c5 = new ComplexNumber();

ComplexNumber c6 = new ComplexNumber();

c3.assign(ComplexNumber.addComplexNumber(c1,c2)); // จะเท่ากับ c3=c1+c2 จะมีค่า 9- 1i

c4.assign(ComplexNumber.subtractComplexNumber(c1,c2)); // จะเท่ากับ c4=c1-c2 จะมีค่า 1+5i

c5.assign(ComplexNumber.multiplyComplexNumber(c1,c2)); // จะเท่ากับ c5=c1\*c2

c6.assign(ComplexNumber.divideComplexNumber(c1,c2)); // จะเท่ากับ c6=c1/c2

}

}