Лабораторна робота 1  
 з дисципліни «Об’єктно-орієнтоване програмування»

Тема роботи: «Класи і об’єкти. Конструктори. Основи перевантаження операцій»  
 Варіант 4

Виконав студент

групи КН-22

Пашковський П. В.

Викладач:

Красовська Г. В.

Клас Complex має поля:

* Private int **first** – цілочисельна дійсна частина,
* Private int **second** – цілочисельна уявна частина,

та методи:

* Public double sqr() - для зведення числа у квадрат.
* Public void scan() – для зчитування значення з клавіатури
* Public void print() – для виведення значень полів на екран
* Complex(int f, int s) – конструктор для полів класу
* Complex(int f) – ініціалізує поле first та ставить значення 0 для другого поля



*Рисунок 1 UML Діаграма класу Complex*

Тестування та аналіз роботи програми:



*Рисунок 2 UML діаграми створених екземплярів класу Complex*

Результат зведення у квадрат чисел a і b:

7:24

0:2

a+b 5:4

a+5 9:3

(a=5) 5:3

5+a 9:3

++a 5:3

a++ 6:3

a 5:3

enter first and second:5 2

enter first and second:3 7

enter first and second:3 11

enter first and second:9 9

enter first and second:6 6

enter first and second:4 6

21:20

-40:42

-112:66

0:162

0:72

-20:48

Код програми(С++):

#include <iostream>

#include <vector>

using namespace std;

class Complex {

int first=0, second=0;

public:

Complex(int f,int s): first(f), second(s) {}

Complex(int f): first(f) {}

Complex() {}

void scan() {

cout <<"enter first and second:";

cin >> first >> second;

}

Complex operator+(Complex& b){

return Complex(first+b.first,second+b.second);

}

Complex operator+(int b){

return Complex(first+b,second);

}

Complex operator=(int b){

return Complex(b,second);

}

Complex operator++(int){

return Complex(first+1,second);

}

Complex operator++(){

++first;

return \*this;

}

void print() {

cout << first <<":"<< second<<endl;

}

Complex sqr() {

return Complex(first\*first - second\*second,2\*first\*second);

}

friend Complex operator+(int a,Complex b);

};

Complex operator+(int a,Complex b){

return Complex(a+b.first,b.second);

}

ostream& operator <<(ostream&out,Complex c){

c.print();

return out;

}

int main() {

Complex a(4,3);

Complex b(1,1);

Complex c = a.sqr();

c.print();

c = b.sqr();

c.print();

cout << "a+b\t" << (a+b);

cout << "a+5\t" << (a + 5);

cout << "(a=5)\t" << (a = 5);

cout << "5+a\t" << (5 + a);

cout << "++a\t" << (++a);

cout << "a++\t" << (a++);

cout << "a\t" << (a);

vector<Complex> v;

for(int i = 0;i<6;++i)

{

c.scan();

v.push\_back(c);

}

for(vector<Complex>::iterator i = v.begin();i!=v.end();++i)

i->sqr().print();

}

Код програми(java):

package com.fit.labs.lab01.task01.entity;  
  
public class Complex {  
 private int first;  
 private int second;  
  
 public Complex() {  
 }  
  
 public Complex(int first, int second) {  
 this.first = first;  
 this.second = second;  
 }  
  
 public int getFirst() {  
 return first;  
 }  
  
 public void setFirst(int first) {  
 this.first = first;  
 }  
  
 public int getSecond() {  
 return second;  
 }  
  
 public void setSecond(int second) {  
 this.second = second;  
 }  
  
 public int getTotal() {  
 return first \* second;  
 }  
  
 @Override  
 public String toString() {  
 return "Product = {" +  
 "weight=" + first +  
 ", price=" + second +  
 "}";  
 }  
}