**First.cpp (основной заголовочный файл для выполнения заданий)**

// RK\_Class\_04032022.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

/\*

Вариант 12

Составить описание класса Complex для представления комплексных чисел

с возможностью задания вещественной и мнимой частей числами типа double.

Класс должен содержать:

1. Конструктор без параметров, в котором в переменные-члены

создаваемого объекта записывается 0.

2. Конструктор с двумя параметрами.

3. Метод изменения значений переменных-членов объекта.

4. Метод вывода на экран значений переменных-членов объекта в виде ”-5.4 + 3.9i”.

Написать программу тестирования класса:

1. Создание объекта без параметров, вывод на экран его переменных-членов.

2. Создание объекта с параметрами, вывод на экран его переменных-членов.

3. Создание динамического объекта без параметров, вывод на экран его переменных-членов.

4. Создание динамического объекта с параметрами, вывод на экран его переменных-членов.

5. Создать массив из объектов класса c разными значениями переменных-членов.

6. Найти в массиве объект с максимальной суммой его действительной и мнимой части и вывести его на экран.

\*/

#include <iostream>

#include "Complex.h"

#pragma warning(disable : 4996)

using namespace std;

int main()

{

system("chcp 1251 > nul");

// 1. Создание объекта без параметров, вывод на экран его переменных-членов.

cout << "1. Создание объекта без параметров, вывод на экран его переменных-членов." << endl;

Complex C1;

C1.show();

// 2. Создание объекта с параметрами, вывод на экран его переменных - членов.

cout << "2. Создание объекта с параметрами, вывод на экран его переменных - членов." << endl;

Complex C2(-3.14, 1.352);

C2.show();

// 3. Создание динамического объекта без параметров,

// вывод на экран его переменных - членов.

cout << "3. Создание динамического объекта без параметров, вывод на экран его переменных - членов." << endl;

Complex\* C3 = new Complex;

C3->show();

// 4. Создание динамического объекта с параметрами,

// вывод на экран его переменных - членов.

cout << "4. Создание динамического объекта с параметрами, вывод на экран его переменных - членов." << endl;

Complex\* C4 = new Complex(65.324, -12.454);

C4->show();

// 5. Создать массив из объектов класса

// c разными значениями переменных - членов.

cout << "5. Создать массив из объектов класса c разными значениями переменных - членов." << endl;

int size = 5;

srand(time(NULL));

Complex\* C5 = new Complex[size];

for (int i = 0; i < size; i++) {

// Знакопеременные числа для действительных и мнимых чисел

// 0 - отрицательное число, 1 - положительное число

// Для действительных чисел

if (rand() % 2 == 1) {

C5[i].set\_double\_number(((double)rand() / (double)(RAND\_MAX \* rand() % 100)));

C5[i].get\_double\_number();

}

else {

C5[i].set\_double\_number((-(double)rand() / (double)(RAND\_MAX \* rand() % 100)));

C5[i].get\_double\_number();

}

// Для мнимых чисел

if (rand() % 2 == 1) {

C5[i].set\_imaginary\_numbers(((double)rand() / (double)(RAND\_MAX \* rand() % 100)));

C5[i].get\_imaginary\_numbers();

}

else {

C5[i].set\_imaginary\_numbers((-(double)rand() / (double)(RAND\_MAX \* rand() % 100)));

C5[i].get\_imaginary\_numbers();

}

C5[i].show();

}

// 6. Найти в массиве объект с максимальной

// суммой его действительной и мнимой части и вывести его на экран.

cout << "6. Найти в массиве объект с максимальной суммой его действительной и мнимой части и вывести его на экран." << endl;

Complex\* C6 = new Complex[size];

double\* find\_max\_value = new double[size];

for (int i = 0; i < size; i++) {

// Знакопеременные числа для действительных и мнимых чисел

// 0 - отрицательное число, 1 - положительное число

// Для действительных чисел

if (rand() % 2 == 1) {

C6[i].set\_double\_number(((double)rand() / (double)(RAND\_MAX \* rand() % 100)));

C6[i].get\_double\_number();

}

else {

C6[i].set\_double\_number((-(double)rand() / (double)(RAND\_MAX \* rand() % 100)));

C6[i].get\_double\_number();

}

// Для мнимых чисел

if (rand() % 2 == 1) {

C6[i].set\_imaginary\_numbers(((double)rand() / (double)(RAND\_MAX \* rand() % 100)));

C6[i].get\_imaginary\_numbers();

}

else {

C6[i].set\_imaginary\_numbers((-(double)rand() / (double)(RAND\_MAX \* rand() % 100)));

C6[i].get\_imaginary\_numbers();

}

C6[i].show();

find\_max\_value[i] = C6[i].get\_double\_number() + C6[i].get\_imaginary\_numbers();

cout << "Сумма действительных и мнимых чисел: " << find\_max\_value[i] << endl;

cout << endl << endl;

}

int i\_max = 0;

for (int i = 1; i < size; i++) {

if (find\_max\_value[i] > find\_max\_value[i\_max]) {

i\_max = i;

}

}

cout << "Сумма максимальнных действительных и мнимых чисел: " << find\_max\_value[i\_max] << endl;

}

**Complex.cpp (Вспомогательный заголовочный файл для класса)**

#include "Complex.h"

#include <iostream>

using namespace std;

//3. Метод изменения значений переменных - членов объекта.

void Complex::set\_double\_number(double \_double\_number) { double\_number = \_double\_number; }

void Complex::set\_imaginary\_numbers(double \_imaginary\_numbers) { imaginary\_numbers = \_imaginary\_numbers; }

double Complex::get\_double\_number() { return double\_number; }

double Complex::get\_imaginary\_numbers() { return imaginary\_numbers; }

double get\_double\_number\_through\_friend(Complex \_double\_number)

{

return \_double\_number.double\_number;

}

**Complex.h (Заголовочный файл класса)**

#pragma once

#include <iostream>

using namespace std;

class Complex

{

private:

// Вещественное число

double double\_number;

// Мнимые числа

double imaginary\_numbers;

public:

//1. Конструктор без параметров, в котором в переменные-члены

// создаваемого объекта записывается 0.

Complex()

{

double\_number = 0;

imaginary\_numbers = 0;

}

//2. Конструктор с двумя параметрами.

Complex(double \_double\_number, double \_imaginary\_numbers)

{

double\_number = \_double\_number;

imaginary\_numbers = \_imaginary\_numbers;

}

//3. Метод изменения значений переменных - членов объекта.

void set\_double\_number(double \_double\_number);

void set\_imaginary\_numbers(double \_imaginary\_numbers);

double get\_double\_number();

double get\_imaginary\_numbers();

//4. Метод вывода на экран значений переменных -

// членов объекта в виде ” - 5.4 + 3.9i”.

void show()

{

if (imaginary\_numbers >= 0)

cout << double\_number << " + " << imaginary\_numbers << "i" << endl;

else

cout << double\_number << " - " << abs(imaginary\_numbers) << "i" << endl;

cout << endl;

};

friend double get\_double\_number\_through\_friend(Complex \_double\_number);

};