***ЛАБОРАТОРНА РОБОТА №1***

**Тема:** Вивчення базових понять класу.

**Мета:** Закріпити базові знання про клас. Навчитись створювати класи засобами мови С++

**Завдання 1.** Створіть клас Int, що імітує стандартний тип int. Єдине поле цього класу повинно мати тип int. Створіть методи, які будуть встановлювати значення поля рівним нулю, ініціалізувати його цілим значенням, виводити значення поля на екран і складати два значення типу Int. Напишіть програму, в якій будуть створені три об'єкти класу Int, два з яких будуть ініціалізованими. Додайте два ініціалізованних об'єкта, надайте результат третьому, а потім відобразіть результат на екрані.

#pragma once

#include <iostream>

using namespace std;

class Int

{

private :

int value;

public :

Int();

Int(int arg);

int getValue();

void setValue(int value);

void display();

int sum(Int other);

};

Int::Int() {

this->value = 0;

}

Int::Int(int arg) {

this->value = arg;

}

int Int::getValue() {

return this->value;

}

void Int::setValue(int value) {

if (value < 0) {

cout << "WRONG ARGUMENT" << endl;

}

else {

this->value = value;

}

}

void Int::display() {

cout << "INT VALUE IS : " << this->value << endl;

}

int Int::sum(Int other) {

return this->value + other.value;

}

int main() {

Int first(4);

Int second(3);

Int third;

cout << "FIRST ";

first.display();

cout << "SECOND ";

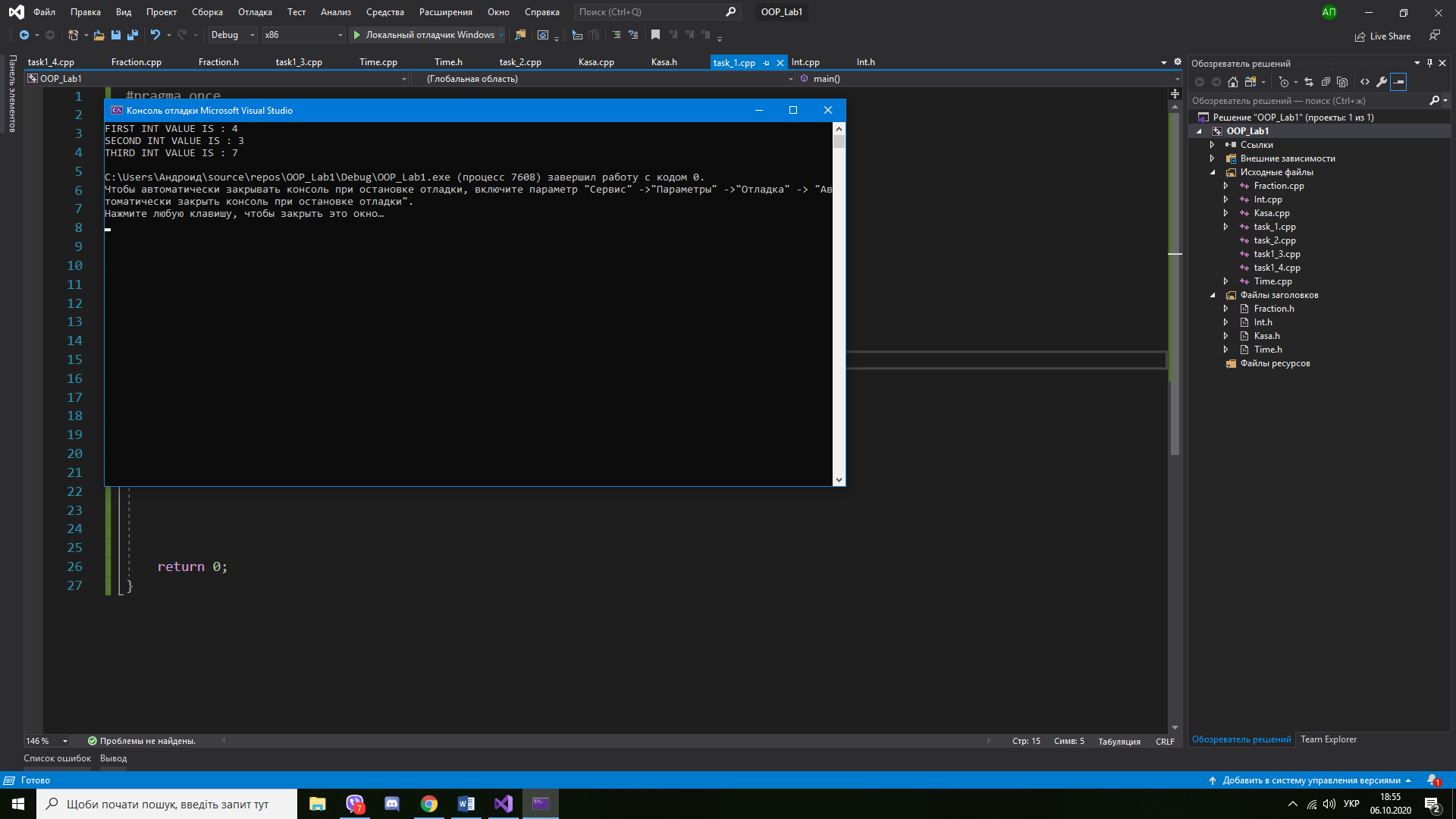
second.display();

third.setValue(first.sum(second));

cout << "THIRD ";

third.display();

return 0;



**Завдання 2.** Уявіть пункт для прийому платежів за проїзд по автостраді. Кожна проїжджаюча машина повинна заплатити за проїзд 50 центів, однак частина машин платить за проїзд, а частина проїжджає безкоштовно. У касі ведеться облік числа проїхавших машин і сумарна виручка від плати за проїзд. Створіть модель такої каси за допомогою класу Kasa. Клас повинен містити два поля. Одне з них, типу unsigned int, призначене для обліку кількості проїхали автомобілів, а друге, що матиме тип double, міститиме сумарну виручку від оплати проїзду. Конструктор повинен ініціалізувати обидва поля нульовими значеннями. Метод payingCar () інкрементує число машин і збільшує на 0,50 сумарну виручку. Інший метод, nорауСаг (), збільшує на одиницю число автомобілів, але залишає без зміни виручку. Метод display () виводить обидва значення на екран. Там, де це можливо, зробіть методи константними. Створіть програму, яка продемонструє роботу класу. Програма повинна запропонувати користувачеві натиснути одну клавішу для того, щоб зімітувати оплату водієм, і іншу клавішу, щоб зімітувати несумлінного водія. Натискання клавіші Esc повинно привести до видачі поточних значень кількості машин і виручки, і до завершення програми.

#pragma once

#include <iostream>

using namespace std;

class Kasa

{

private :

unsigned int carsAmount;

double totalMoney;

public :

Kasa();

void payingCar();

void nonPayCar();

void display();

};

Kasa::Kasa() {

this->carsAmount = 0;

this->totalMoney = 0;

}

void Kasa::payingCar() {

this->carsAmount++;

this->totalMoney += 0.5;

}

void Kasa::nonPayCar() {

this->carsAmount++;

}

void Kasa::display() {

cout << "CARS AMOUNT : " << carsAmount << endl;

cout << "TOTAL MONEY : " << totalMoney << endl;

}

int main() {

Kasa kasa;

int ch;

do {

cout << "PAY (1 - YES / 0 - NO) :";

cin >> ch;

if (ch == 1) {

kasa.payingCar();

}

else if(ch == 0){

kasa.nonPayCar();

}

else {

kasa.display();

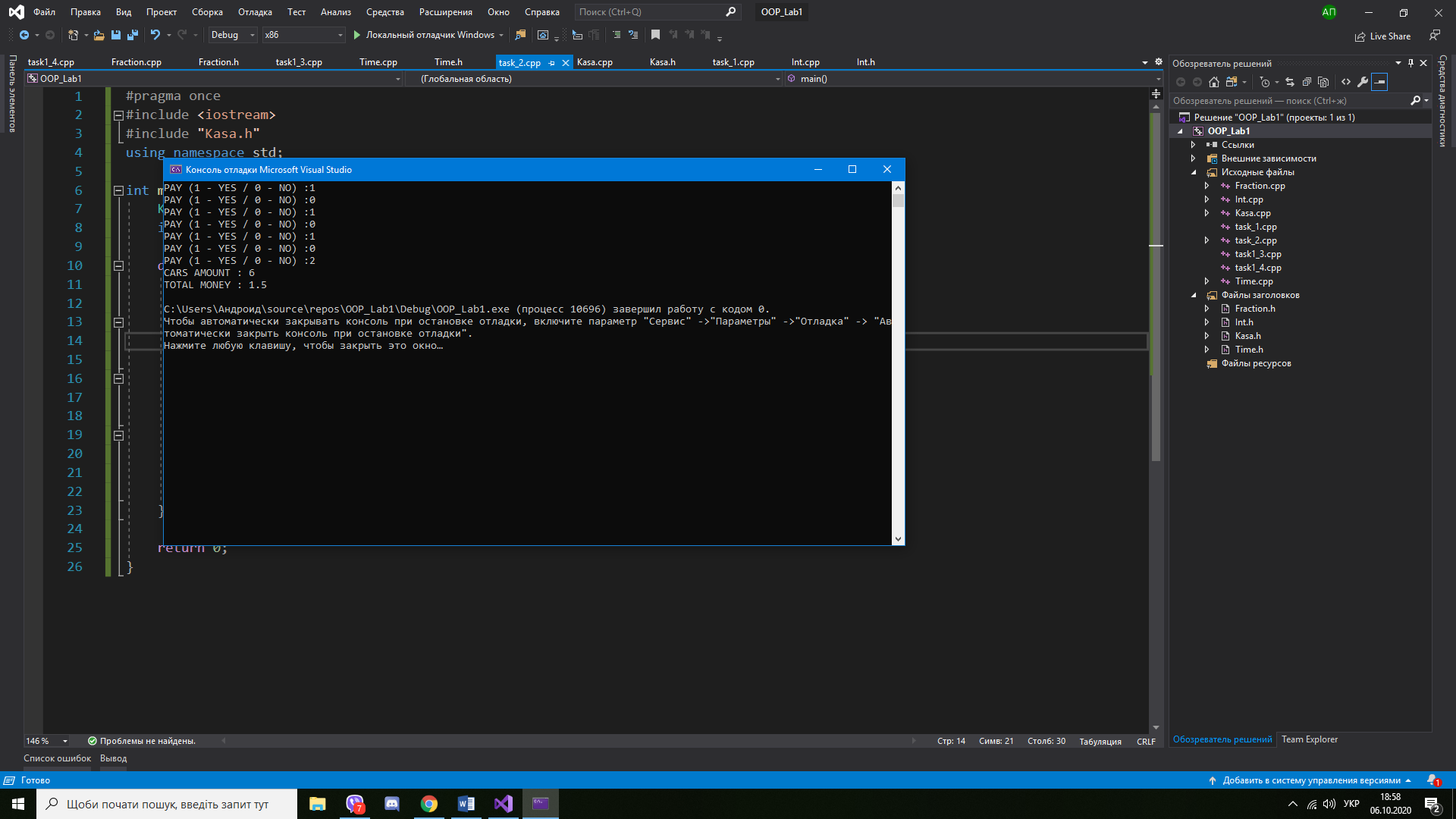
break;

}

} while (true);

return 0;

}



**Завдання 3.** Створіть клас з ім'ям time, що містить три поля типу int, призначених для зберігання годин, хвилин і секунд. Один з конструкторів класу повинен ініціалізувати поля нульовими значеннями, а інший конструктор - заданим набором значень. Створіть метод класу, який буде виводити значення полів на екран у форматі 11:59:59, і метод, складає значення двох об'єктів типу time, переданих в якості аргументів. У функції main () слід створити два ініціалізованих об'єкта (подумайте, чи повинні вони бути константними) і один неініціалізованний об'єкт. Потім складіть два ініціалізованих значення, а результат надайте третьому об'єкту і виведіть його значення на екран. Де можливо, зробіть методи константними.

#pragma once

#include <iostream>

using namespace std;

class Time

{

private :

int h;

int m;

int s;

public :

Time();

Time(int h, int m, int s);

void display();

Time\* sum(Time other);

};#pragma once

#include <iostream>

using namespace std;

Time::Time() {

this->h = 0;

this->m = 0;

this->s = 0;

}

Time::Time(int h, int m, int s) {

this->h = h;

this->m = m;

this->s = s;

}

void Time::display() {

cout << h << ":" << m << ":" << s << endl;

}

Time\* Time::sum(Time other) {

Time\* result = new Time();

int result\_h = this->h + other.h;

int result\_m = this->m + other.m;

int result\_s = this->s + other.s;

if (result\_s >= 60) {

result\_m += result\_s / 60; // m = 5 c = 121 ->> m = 7

result\_s = result\_s % 60;

}

if (result\_m >= 60) {

result\_h += result\_m / 60;

result\_m = result\_m % 60;

}

if (result\_h >= 24) {

result\_h = result\_h % 24;

}

result->h = result\_h;

result->m = result\_m;

result->s = result\_s;

return result;

}

int main() {

Time first(12,44,36);

Time second(14,34,22);

Time\* third;

cout << "FIRST ";

first.display();

cout << "SECOND ";

second.display();

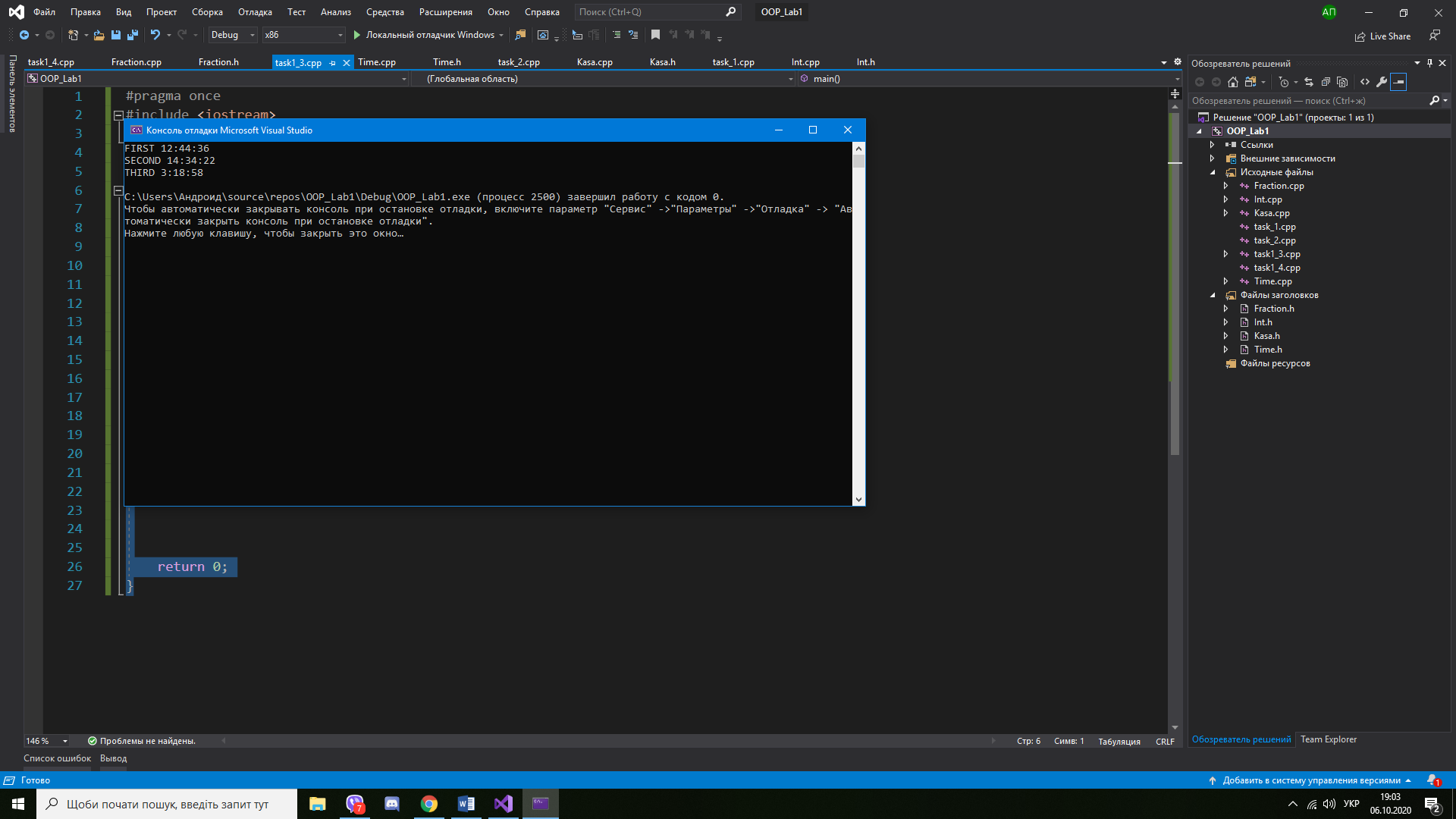
third = first.sum(second);

cout << "THIRD ";

third->display();

return 0;

}



**Завдання 4.** \* У кожному завданні потрібно реалізувати клас. У програмі обов'язково повинні бути продемонстровані різні способи створення об'єктів і масивів об'єктів. Програма повинна демонструвати використання всіх функцій і методів. У всіх завданнях обов’язково повинні бути присутні: • метод ініціалізації Init( ), метод повинен контролювати значення аргументів на коректність; • введення з клавіатури Read( ); • виведення на екран Display( ).

**Варіант 1:**

Атрибут first – ціле число, ціла частина числа; атрибут second – додатнє ціле число, дробова частина числа. Реалізувати метод multiply() – множення на довільне ціле число типу int. Метод повинен правильно працювати при любих допустимих значеннях first і second.

#pragma once

#include <iostream>

using namespace std;

class Fraction

{

private:

int first;

int second;

public :

void Init(int \_first, int \_second);

void Read();

void Display();

void multiply(int value);

};

void Fraction::Init(int \_first, int \_second) {

if (\_second < 0) {

\_second = 0;

}else{

this->second = \_second;

}

this->first = \_first;

}

void Fraction::Read() {

int temp1, temp2;

cout << "Enter first : ";

cin >> temp1;

cout << "Enter second : ";

cin >> temp2;

Init(temp1, temp2);

}

void Fraction::Display() {

cout << first << "." << second << endl;

}

void Fraction::multiply(int value) {

first = first \* value;

int digits = 1;

int old = second;

while (old > 0) {

old = old / 10;

digits\*=10;

}

cout << digits << endl;

double temp\_second = ((double)second / digits) \* value;

cout << temp\_second << endl;

this->first += (int)temp\_second;

this->second = (temp\_second - ((int)temp\_second));

}

int main() {

Fraction first;

cout << "FIRST ";

first.Read();

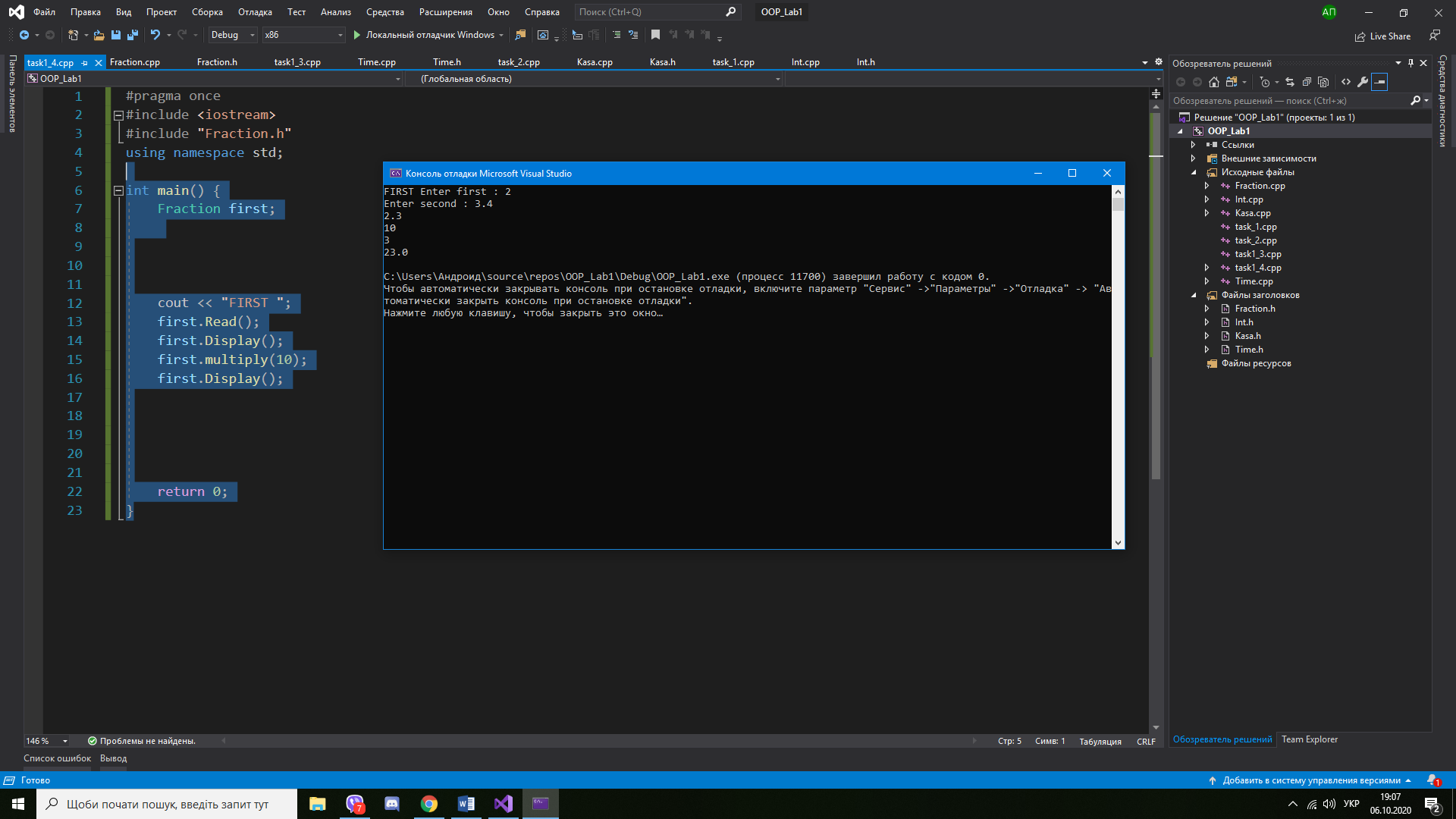
first.Display();

first.multiply(10);

first.Display();

return 0;

}



**Висновок:** на лабораторінй №1 я закріпив базові знання про клас. Навчився створювати класи засобами мови С++