МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ТЕХНІЧНИЙ КОЛЕДЖ ТНТУ ІМЕНІ ІВАНА ПУЛЮЯ

ВІДДІЛЕННЯ ЕЛЕКТРОННИХ АПАРАТІВ

Циклова комісія програмних систем і комплексів

**ЗВІТ**

про виконання лабораторних робітз дисципліни:

**«ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»**

Студента  3  курсу групи КН-321 спеціальності 122 «Комп’ютерні науки»

Слюсарчина О.В

(прізвище та ініціали)

Перевірив: Р.О. Слободян

(підпис)

Тернопіль – 2020

## Лабораторна робота №1

**Тема:** Класи та структури.

**Мета:** Навчитись створювати класи засобами мови С++, виконати завдання згідно варіанту.

**ПОРЯДОК ВИКОНАННЯ РОБОТИ**

**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.

### **КОД ПРОГРАМИ**

#include <iostream>

using namespace std;

class Int

{

private: int x; public:

Int(int x)

{

this->x=x;

};

Int()

{};

void SetInt(int x)

{

this->x=x;

}

int GetInt()

{

return x;

};

void Sum(Int n1, Int n2)

{

x= n1.GetInt()+n2.GetInt() ;

};

void Null()

{ x=0;

};

};

int main()

{

Int n1(2),n2(3),n3;

int k; // Надання змінні х обєкта 3 свого резутьтату cout<<"Enter x to objekt 3"<<endl; cin>>k; n3.SetInt(k);

k=n3.GetInt();

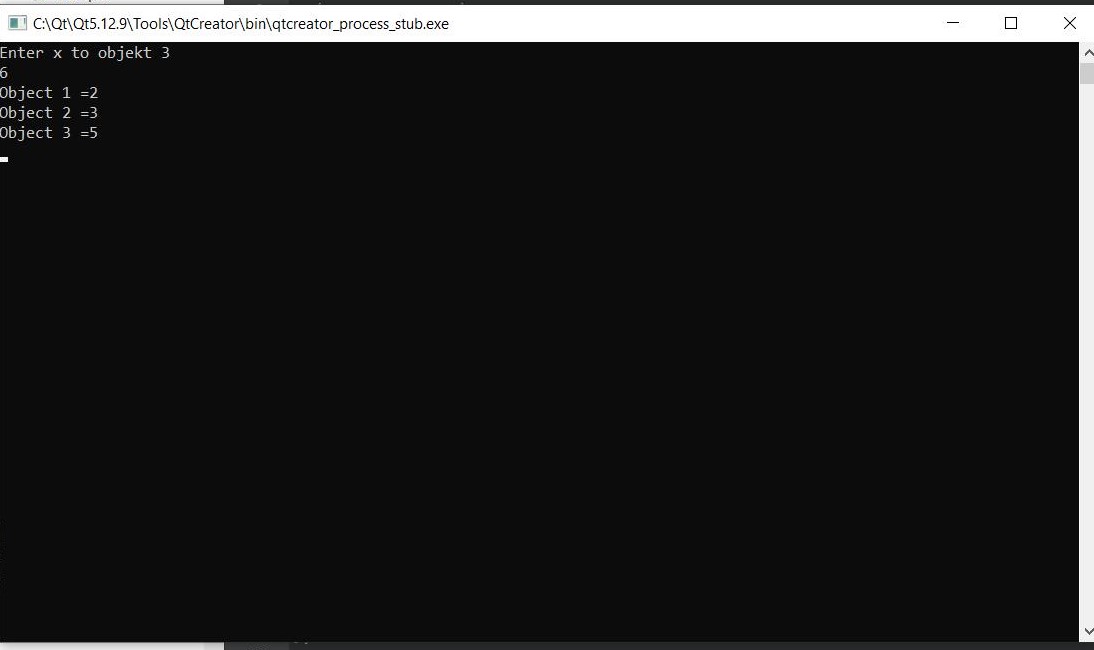
n3.Null(); // Обнулення змінної х обєкта 3

n3.Sum(n1,n2); //Додаєтся два ініціалізованних об'єкта, надєтся результат третьому

cout<<"Object 1 ="<<n1.GetInt()<<endl; //Вивід Всіх обєктів cout<<"Object 2 ="<<n2.GetInt()<<endl; cout<<"Object 3 ="<<n3.GetInt()<<endl;

return 0; }

**РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ**



**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.

### **КОД ПРОГРАМИ**

#include <iostream> #include <conio.h>

using namespace std;

class Casa

{

unsigned int n; double pay; public:

Casa(unsigned int n,double pay)

{

this->n=n;

this->pay=pay;

};

void payingCar()

{ n++;

pay+=0.5;

};

void nonepayCar()

{

n++;

};

void display()

{

cout<<"N="<<n<<endl<<"Pay="<<pay<<"$"<<endl;

};

};

int main()

{ int a;

Casa K(0,0); cout<<"Pay"<<endl; point1:

cout<<"y-Yes n-No End"<<endl;

while(1)

{

a=getch(); if(a==121)

{

K.payingCar(); cout<<"Yes"<<endl; goto point1;

}

else if(a==110)

{

K.nonepayCar();

cout<<"No"<<endl;

goto point1;

}

else if(a==27)

{

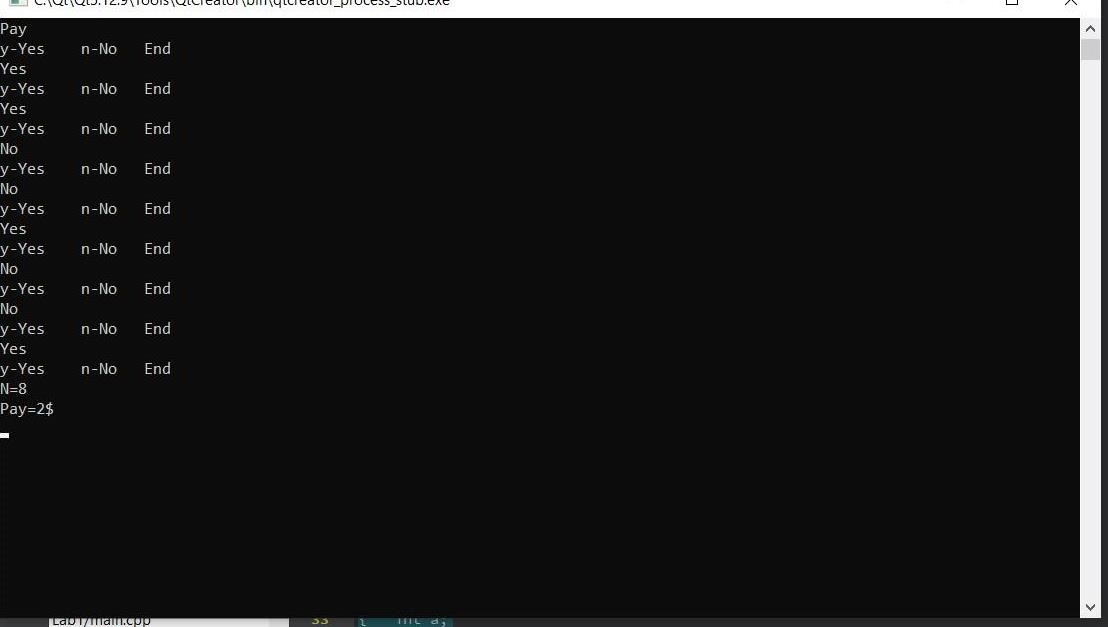
K.display();

}

}

return 0; }

### **РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ**



**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

### **КОД ПРОГРАМИ**

#include <iostream>

#include <iomanip>

using namespace std;

class time { int hour,min,sec; public: time(int hour,int min,int sec)

{

this->hour=hour; this->min=min;

this->sec=sec;

}

time(){};

void GetTime()

{ while(sec>60){min++;sec=sec-60;}; while(min>60){hour++;min=min-60;}; while(hour>=12){hour=(hour-12);};

cout<<"Time="<<setw(2)<<hour<<":"<<setw(2)<<min<<":"<<setw(2)<<sec<<endl;

};

void SetTime(int hour,int min,int sec)

{

this->hour=hour; this->sec=sec;

this->min=min;

}

void SumTime(time obj1,time obj2)

{

hour=obj1.hour+obj2.hour; min=obj1.min+obj2.min; sec=obj1.sec+obj2.sec; while(sec>60){min++;sec=sec-60;}; while(min>60){hour++;min=min-60;}; while(hour>=12){hour=(hour-12);};

cout<<"Time="<<setw(2)<<hour<<":"<<setw(2)<<min<<":"<<setw(2)<<sec<<endl;

}

};

int main()

{

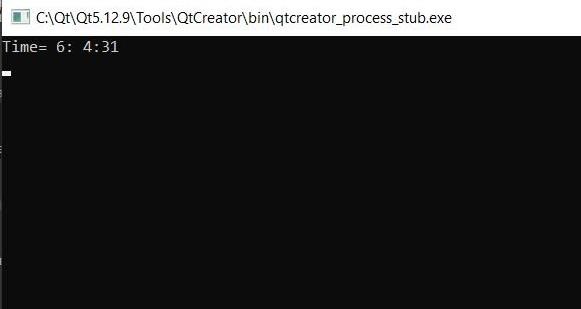
class time N1(12,3,1), N2(6,1,30), N3;

N3.SumTime(N1,N2); //Формат дати 12:59:59

return 0;

}

### **РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ**



**Завдання4:** У даній лабораторній роботі в 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 №4:** Атрибут first – ціле додатнє число , номінал купюри; номінал може приймати значення 1,2,5,10,50,100,500,1000,5000. Атрибут second – ціле додатнє число , кількість купюр данного достоїнства. Реалізувати метод summa() – обчислення грошової суми.

**КОД ПРОГРАМИ**

#include <iostream>

using namespace std;

class Pay

{

private:

int first=0;

int second=0;

int pay=0;

public:

Pay()

{}

int Nominal()

{

return first;

}

void read()

{

cin>>first;

}

void display()

{

cout<<"Pay="<<pay<<endl;

cout<<"N="<<second<<endl;

}

void suma()

{

pay+=first;

second++;

}

};

int main()

{

Pay pay;

cout<<"Enter price 1,2,5,10,50,100,500,1000,5000 Enter 0 to exit"<<endl;

do{

cout<<"Enter price=";

pay.read();

switch (pay.Nominal())

{

case 1 :{pay.suma();break;};

case 2 : {pay.suma();break;};

case 5 : {pay.suma();break;};

case 10 : {pay.suma();break;};

case 50 : {pay.suma();break;};

case 100 : {pay.suma();break;};

case 500 : {pay.suma();break;};

case 1000 : {pay.suma();break;};

case 5000 : {pay.suma();break;};

default:{cout<<"ERROR"<<endl;};

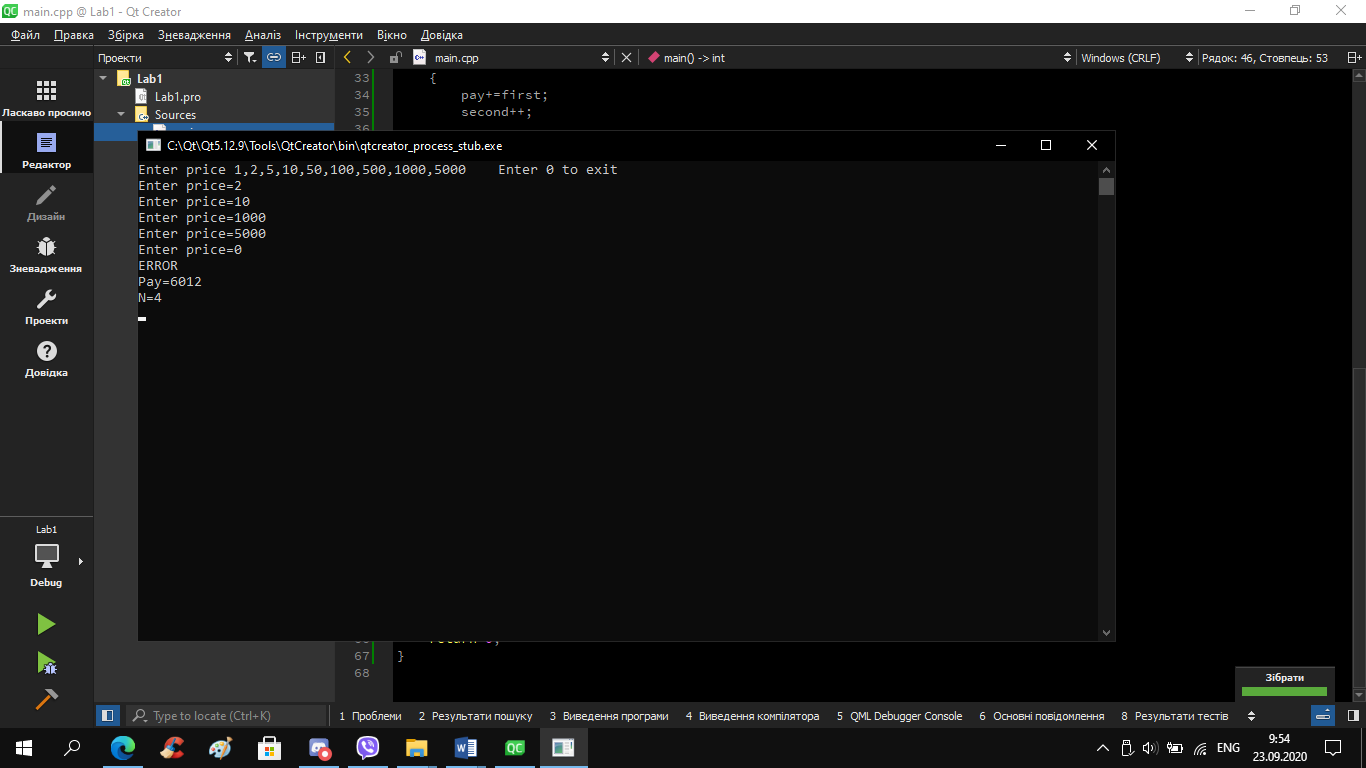
}

}while(pay.Nominal()!=0);

pay.display();

return 0;

### **РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ**



**ВИСНОВОК:** На даній лабораторній роботі я навчився створювати класи засобами мови С++