

Міністерство освіти і науки України

Відокремлений структурний підрозділ «Тернопільський фаховий
коледж Тернопільського національного технічного університету
імені Івана Пулюя»

Циклова комісія комп'ютерних наук

ЗВІТ

про виконання лабораторної роботи №1

з дисципліни: “Об’єкто-орієнтоване програмування”

на тему: “Вивчення базових понять класу”

Виконала:
студентка групи КН-321
Байдецька. В. П.

Прийняв:
Слободян

Тернопіль 2025

Мета: Закріпити базові знання про клас, набути навичок розробки класів мовою програмування C++.

Хід роботи

Завдання 1. Створіть клас Integer, який буде обгорткою для стандартного типу int:

єдине поле класу Integer повинно мати тип int. Створіть методи, які будуть:

- встановлювати значення поля рівним нулю;
- ініціалізувати об'єкт класу цілим значенням;
- виводити значення поля на екран;
- сумувати два значення типу Integer.

Напишіть програму, в якій будуть створені три об'єкти класу Integer, два з яких будуть

ініціалізованими цілочисельними числами. Додайте два ініціалізованих об'єкта за допомогою

визначеного методу і результат збережіть у третьому об'єкті класу Integer, а потім відобразіть

на екрані значення збереженні у всіх трьох об'єктах класу Integer.

Код:

```
#include <iostream>
using namespace std;
class Integer{
    int a;
    public:
        void setInt(int a);
        int getInt();
        void init();
};

void Integer::setInt(int a){
    this->a = a;
}

int Integer::getInt(){
    return a;
```

```

    }

    void Integer::init(){
        a = 0;
    }

int main(){
    Integer x1;
    Integer x2;
    Integer y;
    int x, y1;
    cout << "enter x1 ";
    cin >> x;
    x1.setInt(x);
    cout << "enter x2 ";
    cin >> x;
    x2.setInt(x);
    y1 = x1.getInt() + x2. getInt();
    y.setInt(y1);
    cout << "x1 = "<<x1.getInt()<<endl;
    cout << "x2 = "<<x2.getInt()<<endl;
    cout << "y = "<<y.getInt()<<endl;
    return 0;
}

```

Виконання:

```

[cheshyrka@archlinux OOP]$ ./a.out
enter x1 2
enter x2 3
x1 = 2
x2 = 3
y = 5

```

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 такі атрибути і методи:

- перший атрибут для обліку кількості автомобілів, які проїхали повз пункт;
- другий міститиме сумарну виручку від оплати проїзду.
- конструктор, який повинен ініціалізувати обидва поля нульовими значеннями;
- метод `payingCar()`, який інкрементує число машин і збільшує на 0,50\$ сумарну виручку;
- метод `порауCar()`, який збільшує на одиницю число автомобілів, але залишає без зміни виручку;
- метод `display()`, який 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:

```
#include <iostream>
#include <termios.h>
#include <unistd.h>
using namespace std;

class Kasa{
    int carCount;
    double money;
public:
    Kasa(); // Constructor
    void payingCar();
    void noPayingCar();
    void display() const;
};

Kasa::Kasa(){
    carCount = 0;
    money = 0;
}

void Kasa::payingCar(){
    carCount++;
    money += 0.5;
}

void Kasa::noPayingCar(){
    carCount++;
}

void Kasa::display() const{
    cout << "Усього автомобілів проїхало: " << carCount << endl;
    cout << "Виручка: " << money << endl;
}

char getch() {
    struct termios oldt, newt;
    char ch;
    tcgetattr(STDIN_FILENO, &oldt);
```

```

    newt = oldt;
    newt.c_lflag &= ~(ICANON | ECHO);
    tcsetattr(STDIN_FILENO, TCSANOW, &newt);
    ch = getchar();
    tcsetattr(STDIN_FILENO, TCSANOW, &oldt);
    return ch;
}

```

```

int main(){
    Kasa kasa;

    cout << "Виберіть подію:" << endl;
    cout << "1. Водій проїхав й оплатив проїзд" << endl;
    cout << "2. Водій ухильнувся" << endl;
    cout << "3. Кінець дня, що там по касі?" << endl;
    cout << "ESC - Вихід з програми" << endl << endl;

    while(true){
        cout << "Натисніть клавішу: " << flush; // flush ensures immediate
display
        char key = getch();
        cout << endl;

        if(key == 27){ // ESC key
            cout << "Вихід з програми..." << endl;
            kasa.display();
            break;
        }

        switch(key){
            case '1':
                kasa.payingCar();
                cout << "Водій оплатив проїзд" << endl << endl;
                break;
            case '2':
                kasa.noPayingCar();
                cout << "Водій ухильнувся від оплати" << endl << endl;
                break;
            case '3':
                cout << "Кінець дня:" << endl;
                kasa.display();

```

```

        return 0;
    default:
        cout << "Невірний вибір" << endl << endl;
    }
}

return 0;
}

```

Виконання:

```

[cheshyrka@archlinux 00P]$ ./a.out
Виберіть подію:
1. Водій проїхав й оплатив проїзд
2. Водій ухильнувся
3. Кінець дня, що там по касі?
ESC - Вихід з програми

Натисніть клавішу:
Водій оплатив проїзд

Натисніть клавішу:
Водій оплатив проїзд

Натисніть клавішу:
Водій ухильнувся від оплати

Натисніть клавішу:
Вихід з програми...
Усього автомобілів проїхало: 3
Виручка: $1

```

Завдання 3. Створіть клас з ім'ям `Time`, що містить три атрибути типу `int`, призначених для зберігання годин, хвилин і секунд. Один з методів класу повинен ініціалізувати поля нульовими значеннями, а інший метод - заданим набором значень. Створіть метод класу, який буде виводити значення полів на екран у форматі `11:59:59`. Також створіть метод, який сумуватиме значення відповідних атрибутів об'єкта із атрибутами об'єкта `Time`, який переданий в якості аргумента, та повертає суму атрибутів у вигляді нового об'єкта класу `Time`. У функції `main()` слід створити два ініціалізованих об'єкта (подумайте, чи повинні вони бути константними) і один неініціалізований об'єкт. Потім просумуйте два ініціалізованих об'єкти, а результат надайте третьому об'єкту і виведіть його значення на екран. Де можливо, зробіть методи константними.

Код:

```
#include <iostream>
#include <iomanip>
using namespace std;
```

```
class Time{
    int hours;
    int minutes;
    int seconds;
public:
    void initZero();
    void initTime(int h, int m, int s);
    void display() const;
    Time addTime(const Time& other) const;
};
```



```
void Time::initZero(){
    hours = 0;
    minutes = 0;
    seconds = 0;
}
```

```
void Time::initTime(int h, int m, int s){
    hours = h;
    minutes = m;
    seconds = s;
}
```

```
void Time::display() const{
    cout << setfill('0') << setw(2) << hours << ":"
        << setfill('0') << setw(2) << minutes << ":"
        << setfill('0') << setw(2) << seconds;
}
```

```
Time Time::addTime(const Time& other) const{
    Time result;

    int ho = hours + other.hours;
    int min = minutes + other.minutes;
    int sec = seconds + other.seconds;

    if(sec >= 60){
        min += sec/60;
        sec %= 60;
    }
}
```

```
if(min >= 60){  
    ho += min/60;  
    min %= 60;  
}
```

```
result.initTime(ho, min, sec);  
return result;  
}
```

```
int main(){  
    int h1, h2, m1, m2, s1, s2;
```

```
    Time time1;  
    Time time2;  
    Time result;
```

```
    cout << "Введіть перший час:" << endl;  
    cout << "Години: ";  
    cin >> h1;  
    cout << "Хвилини: ";  
    cin >> m1;  
    cout << "Секунди: ";  
    cin >> s1;
```

```
    cout << "\nВведіть другий час:" << endl;  
    cout << "Години: ";  
    cin >> h2;  
    cout << "Хвилини: ";  
    cin >> m2;  
    cout << "Секунди: ";  
    cin >> s2;
```

```
time1.initTime(h1, m1, s1);
time2.initTime(h2, m2, s2);

result = time1.addTime(time2);

cout << "\nРезультати:" << endl;
cout << "Перший час: ";
time1.display();
cout << endl;

cout << "Другий час: ";
time2.display();
cout << endl;

cout << "Сума часів: ";
result.display();
cout << endl;

return 0;
}
```

Виконання:

```
[cheshyrka@archlinux 00P]$ ./a.out
Введіть перший час:
Години: 2
Хвилини: 7
Секунди: 30

Введіть другий час:
Години: 8
Хвилини: 20
Секунди: 10

Результати:
Перший час: 02:07:30
Другий час: 08:20:10
Сума часів: 10:27:40
```

Висновок: В ході виконання цієї лабораторної роботи я закріпила базові знання про клас, набула навичок розробки класів мовою програмування C++.