**мІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**нАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «лЬВІВСЬКА ПОЛІТЕХНІКА»**

**ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**Кафедра ІСМ**

|  |
| --- |
| **2022** |



**ЗВІТ**

**Про виконання лабораторної роботи № 3.3C**

**«Успадкування замість композиції»**

**З дисципліни**

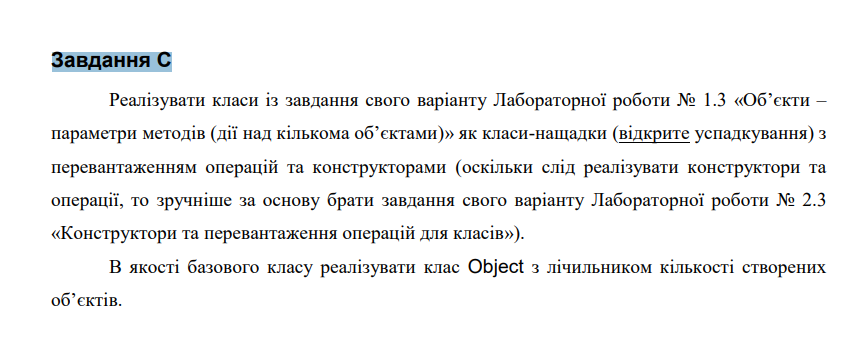
**«Об’єктно-орієнтоване програмування»**

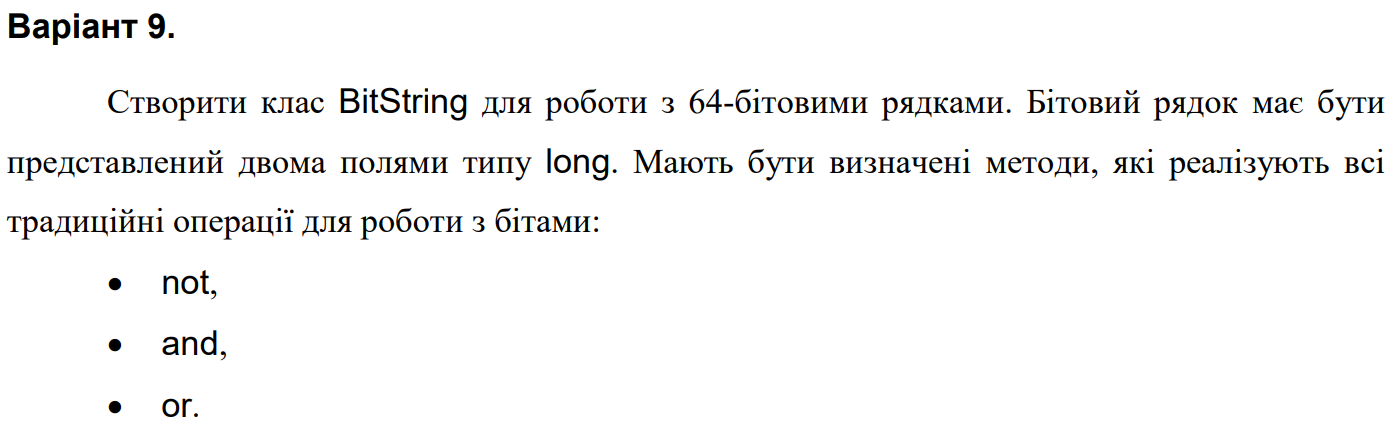
**Студента групи ІТ-11**

**Лисецького Володимира Любомировича**

**Мета роботи:** Освоїти використання успадкування.

**Умова завдання:**

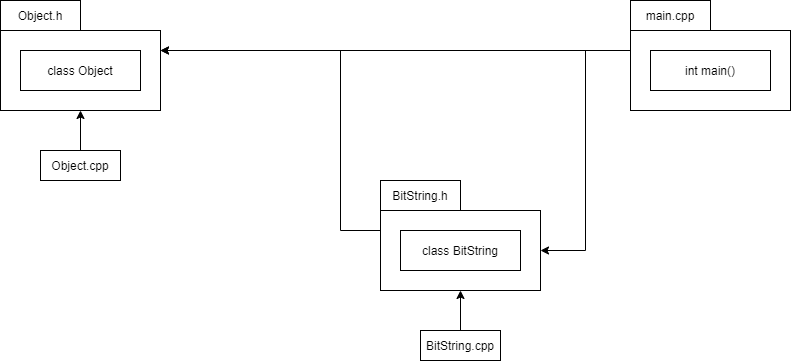
****

****

**UML-діаграма класів:**



**Структурна схема програми:**

****

**Текст програми:**

////////////////////////////////////////////////

// Lab3\_3C.cpp : This file contains the 'main' function. Program execution begins and ends there.

//

#include <iostream>

#include "BitString.h"

#include "Object.h"

using namespace std;

int main()

{

BitString b1, b2;

cout << "first object" << endl;

cin >> b1;

cout << "second object" << endl;

cin >> b2;

cout << endl;

cout << "first object" << endl;

cout << b1;

cout << "second object" << endl;

cout << b2;

cout << endl;

BitString NOT1, NOT2, OR, AND;

NOT1 = (!b1);

NOT2 = (!b2);

OR = (b1 | b2);

AND = (b1 & b2);

cout << "!(first object)" << endl;

cout << NOT1;

cout << "!(second object)" << endl;

cout << NOT2;

cout << "first object | second object" << endl;

cout << OR;

cout << "first object & second object" << endl;

cout << AND;

cout << endl << "object count = " << b1.get\_count() << endl;

}

#pragma once

class Object

{

protected:

static unsigned int count;

public:

Object();

~Object();

unsigned int get\_count() { return count; };

};

///////////////////////////////////////////////////////////////////////////

// BitString.h

#pragma once

#include "Object.h"

#include <iostream>

#include <string>

using namespace std;

class BitString :

public Object

{

private:

long a;

long b;

public:

long getA() const { return a; };

long getB() const { return b; };

void setA(long value) { a = value; }

void setB(long value) { b = value; }

operator string() const;

BitString& operator = (const BitString& obj);

BitString();

BitString(long first, long second);

BitString(BitString& obj);

BitString operator ++();

BitString operator --();

BitString operator ++(int);

BitString operator --(int);

friend BitString operator ! (BitString obj);

friend BitString operator | (BitString obj1, BitString obj2);

friend BitString operator & (BitString obj1, BitString obj2);

friend ostream& operator << (ostream& out, BitString& obj);

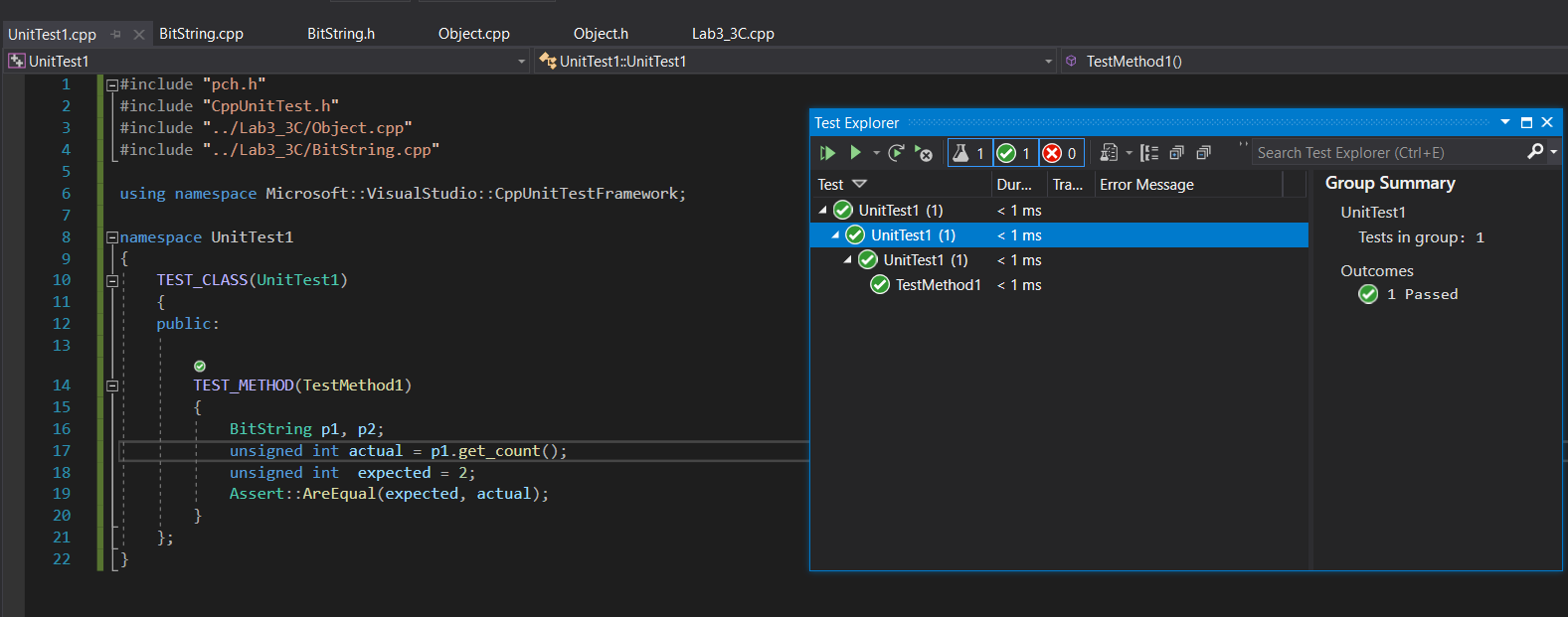
friend istream& operator >> (istream& in, BitString& obj);

};

**Посилання на git-репозиторій з проектом:**

**https://github.com/MarshmallowSoup/oop-lab-3-3C**

**Результати unit-тесту:**

****

**Висновки:6.05 o 5.08 закінчив писати звіт.**