

**ЗВІТ**

**Про виконання лабораторної роботи № 1.2**

**«Оголошення та будова класу»**

**З дисципліни**

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

**Студентки групи ІТ-12**

**Шкринди Володимира Володимировича**

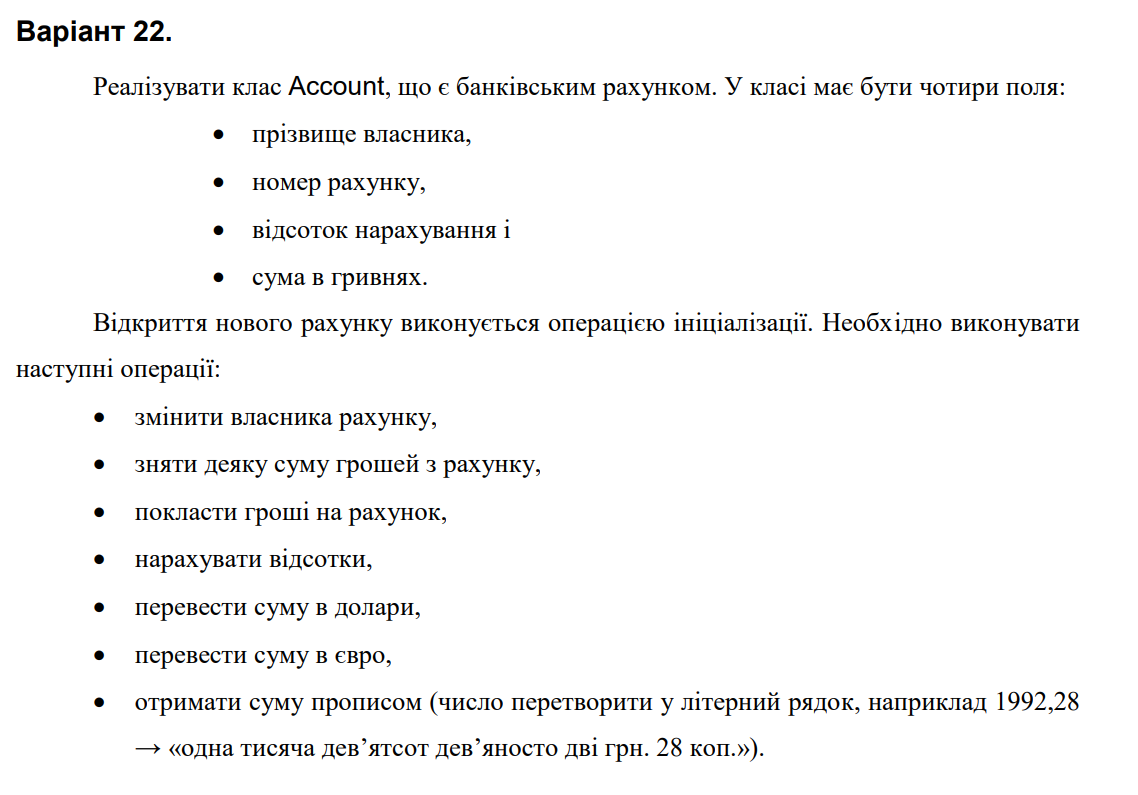
**Прийняв викладач**

**Григорович В. Г.**

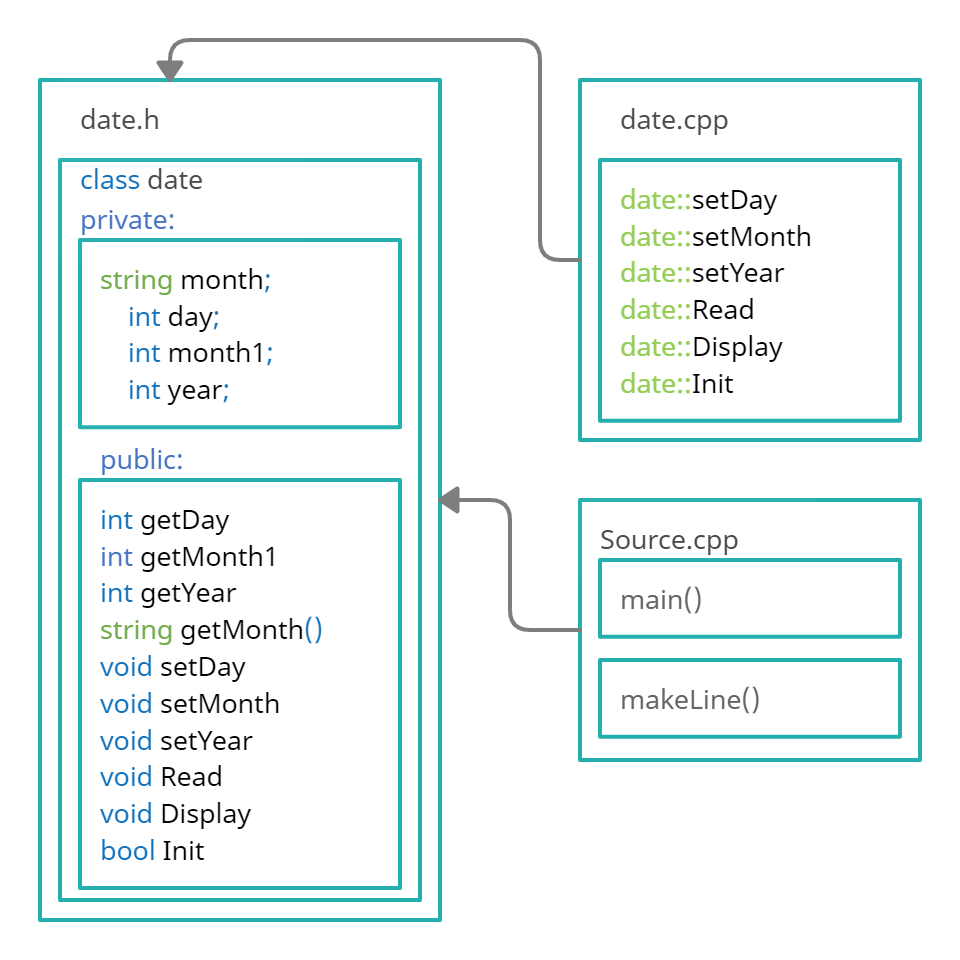
**2021**

**Мета роботи:** Освоїти використання класів та об’єктів.

**Умова завдання:**



**UML-діаграма класів:** 

**Структурна схема програми:** 

**Текст програми:**

**Source.cpp**

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

//Source.cpp

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include "Account.h"

#include <Windows.h>

using namespace std;

Account makeAccount(int no, string name, double summa, double percent)

{

Account a;

if (!a.Init(no, name, summa, percent))

cout << "Неправильні данні!" << endl;

return a;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

Account a1;

a1.Read();

a1.Display();

a1.Withdraw();

a1.Put();

a1.Perc();

a1.toDollar();

a1.Display();

a1.Display();

char s[100];

strcpy(s, a1.SumToNumeral());

cout << s << endl << endl;

a1.SumToNumeral();

Account a2;

string name;

int no;

double summa;

double percent;

cout << " no = ? "; cin >> no;

cout << " name = ? "; cin >> name;

cout << " summa = ? "; cin >> summa;

cout << " percent = ? "; cin >> percent;

a2 = makeAccount(no, name, summa, percent);

a2.Display();

return 0;

}

**Account.h**

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

#pragma once

#include <iostream>

#include <string>

#include <sstream>

using namespace std;

class Account

{

private:

int no;

string name;

double summa;

double percent;

public:

int GetNo() const { return no; }

string GetName() const { return name; }

double GetSumma() const { return summa; }

double GetPercent() const { return percent; }

void SetName(string value) { name = value; }

bool SetNo(int value);

bool SetSumma(double value);

bool SetPercent(double value);

bool Init(int no, string name, double summa,double percent);

void Read();

void Display();

double Withdraw();

double Put();

void Perc() {cout<<"Summa + Percent = "<< Put() + (((summa / 100) \* percent)) <<endl;}

void toDollar() {cout<<"Summa in dollar = "<< summa \* 0.036 << endl;}

void toEuro() {cout<<"Summa in euro = " << summa \* 0.03 << endl; }

const char\* SumToNumeral();

};

**Account.cpp**

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

//date.cpp

#include "Account.h"

using namespace std;

bool Account::SetNo(int value)

{

if (value > 0)

{

no = value;

return true;

}

else

{

no = 0;

return false;

}

}

bool Account::SetSumma(double value)

{

if (value > 0)

{

summa = value;

return true;

}

else

{

summa = -value;

return false;

}

}

bool Account::SetPercent(double value)

{

if (value > 0)

{

percent = value;

return true;

}

else

{

percent = -value;

return false;

}

}

bool Account::Init(int no, string name, double summa,double percent)

{

SetName(name);

return SetNo(no) && SetSumma(summa) && SetPercent(percent);

}

void Account::Read()

{

string name;

int no;

double summa;

double percent;

do

{

cout << " no = ? "; cin >> no;

cout << " name = ? "; cin >> name;

cout << " summa = ? "; cin >> summa;

cout << " percent = ? "; cin >> percent;

} while (!Init(no, name, summa, percent));

}

void Account::Display()

{

cout << " no = " << no << endl;

cout << " name = " << name << endl;

cout << " summa = " << summa<< endl;

cout << " percent = " << percent << endl;

}

double Account::Withdraw()

{

double add;

cout << "Сума стягу: "; cin >> add;

return summa -= add;

}

double Account::Put()

{

double add;

cout << "Сума начислення: "; cin >> add;

return summa += add;

}

const char\* Account::SumToNumeral()

{

const char\* \_centuries[11] = { "",

"сто",

"двісті",

"триста",

"чотириста",

"п'ятсот",

"шістсот",

"сімсот",

"вісімсот",

"дев'ятсот",

"тисяча або більше" };

const char\* \_decades[10] = { "",

"",

"двадцять",

"тридцять",

"сорок",

"п'ятдесят",

"шістдесят",

"сімдесят",

"вісімдесят",

"дев'яносто" };

const char\* \_digits[20] = { "",

"один",

"два",

"три",

"чотири",

"п'ять",

"шість",

"сім",

"вісім",

"дев'ять",

"десять",

"одинадцять",

"дванадцять",

"тринадцять",

"чотирнадцять", "п'ятнадцять",

"шістнаднадцять",

"сімнадцять",

"вісімнадцять", "дев'ятнадцять" };

if (summa >= 1000)

return \_centuries[10];

int sum = floor(summa);

int cen = sum / 100;

sum = sum % 100;

int dec = sum / 10;

int dig;

if (dec == 0 || dec == 1)

dig = sum % 20;

else

dig = sum % 10;

char s[100] = "";

strcat\_s(s, \_centuries[cen]);

strcat\_s(s, " ");

strcat\_s(s, \_decades[dec]);

strcat\_s(s, " ");

strcat\_s(s, \_digits[dig]);

return s;

}**Результати Unit тесту:**

#include "pch.h"

#include "CppUnitTest.h"

#include "../Lab1\_2/Account.h"

#include "../Lab1\_2/Account.cpp"

using namespace Microsoft::VisualStudio::CppUnitTestFramework;

namespace UnitTest1

{

TEST\_CLASS(UnitTest1)

{

public:

TEST\_METHOD(TestMethod1)

{

Account a, b;

bool res1, res2;

res1 = a.SetNo(2);

res2 = b.SetNo(3);

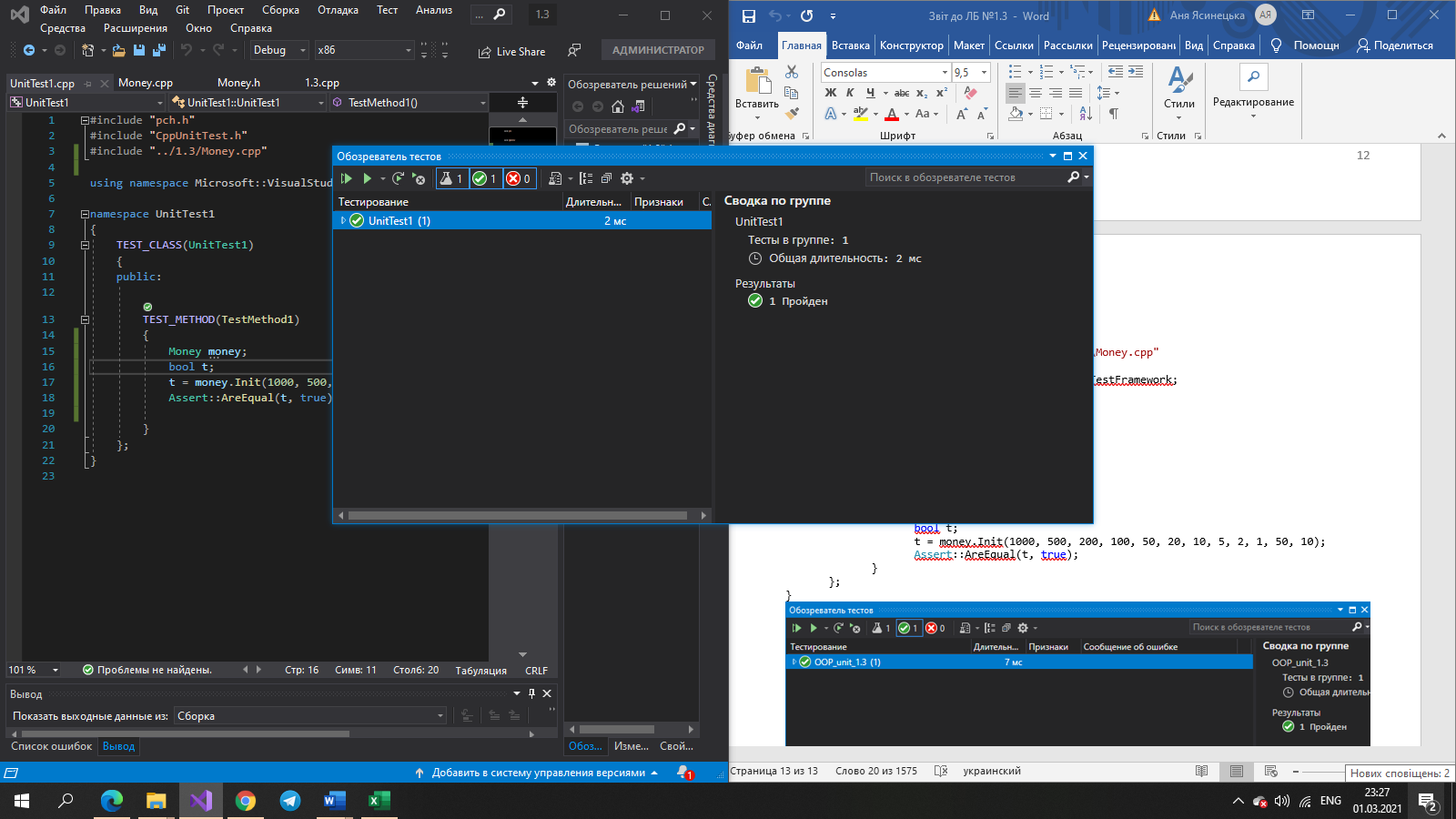
Assert::IsTrue(res1);

Assert::IsTrue(res2);

}

};

}



**Посилання на git-репозиторій з проектом:** [**https://github.com/Ysntska/1.2\_oop.git**](https://github.com/Ysntska/1.2_oop.git)

**Висновок:** У ході лабораторної роботи я навчився використовувати класи та об’єкти.