

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 1

ЗАТВЕРДЖЕНО

Вченою радою факультету
комп'ютерно-інтегрованих
технологій, мехатроніки і
робототехніки

31 серпня 2022 р., протокол № 7

Голова Вченої ради

Олексій ГРОМОВИЙ



МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

для проведення лабораторних робіт
з навчальної дисципліни

«КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ ТА ПРОГРАМУВАННЯ»

для здобувачів вищої освіти освітнього ступеня «бакалавр»
денної та заочної форми навчання

спеціальності 152 - «Метрологія та інформаційно-вимірювальна техніка»
освітньо-професійна програма: «Комп'ютеризовані інформаційно-вимірювальні системи»
факультет комп'ютерно-інтегрованих технологій, мехатроніки і робототехніки
кафедра автоматизації та комп'ютерно-інтегрованих технологій
імені професора Б.Б.Самотокіна

Схвалено на засіданні кафедри
автоматизації та комп'ютерно-
інтегрованих технологій
ім. проф. Б.Б. Самотокіна
29 серпня 2022 р.,
протокол № 6

Завідувач кафедри

А.Ткачук Андрій ТКАЧУК

Гарант ОПП

Ю.Подчашинський Юрій ПОДЧАШИНСЬКИЙ

Розробник: к.т.н., доцент кафедри автоматизації та комп'ютерно-інтегрованих технологій
ім. проф. Б.Б.Самотокіна ДОБРЖАНСЬКИЙ Олександр

Житомир
2022– 2023 н.р.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 2

З М І С Т

Ч А С Т И Н А І

Лабораторна робота № 1 Ознайомлення з середовищем розробки Microsoft Visual С++. Перевірка базових навичок використання інформаційних і комунікаційних технологій	6
Лабораторна робота № 2 Принципи побудови обчислювальних компонент засобів вимірювальної техніки: Циклічні обчислення	20
Лабораторна робота № 3 Принципи побудови обчислювальних компонент засобів вимірювальної техніки: Математичні операції та організація меню користувача	33
Лабораторна робота № 4 Принципи побудови обчислювальних компонент засобів вимірювальної техніки: Операції з масивами.	49
Лабораторна робота № 5 (продовження Л.р.№4) Принципи побудови обчислювальних компонент засобів вимірювальної техніки: Операції з масивами.	75
Лабораторна робота № 6 (продовження Л.р.№5) Принципи побудови обчислювальних компонент засобів вимірювальної техніки: Операції з масивами.	94
Лабораторна робота № 7 Сучасні інженерні та математичні пакети для створення моделей приладів і систем вимірювань: Моделювання функціонування вимірювальних систем на базі мікроконтролерів.	116

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 3

Лабораторна робота № 8 (продовження Л.р.№7)

Сучасні інженерні та математичні пакети для створення моделей приладів і систем вимірювань: Моделювання функціонування вимірювальних систем на базі мікроконтролерів.

129

Лабораторна робота № 9 (продовження Л.р.№10)

Сучасні інженерні та математичні пакети для створення моделей приладів і систем вимірювань: Моделювання функціонування вимірювальних систем на базі мікроконтролерів.

135

Лабораторна робота № 10 (продовження Л.р.№9)

Сучасні інженерні та математичні пакети для створення моделей приладів і систем вимірювань: Моделювання функціонування вимірювальних систем на базі мікроконтролерів.

145

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 4

Ч А С Т И Н А І І

Лабораторна робота № 1

**Інформаційні процеси у комп'ютеризованих
вимірювальних системах. Операції з формами Windows. 116**

Лабораторна робота № 2

**Інформаційні процеси у комп'ютеризованих
вимірювальних системах. Використання пам'яті
програмами в Windows. 133**

Лабораторна робота № 3

**Інформаційні процеси у комп'ютеризованих
вимірювальних системах. Використання пам'яті
програмами в Windows. 142**

Лабораторна робота № 4

**Інформаційні процеси у комп'ютеризованих
вимірювальних системах. Застосування GDI+ та
Windows Forms. 147**

Лабораторна робота № 5 (продовження л.р. №4)

**Інформаційні процеси у комп'ютеризованих
вимірювальних системах. Застосування GDI+ та
Windows Forms. 164**

Лабораторна робота № 6 (продовження л.р. №5)

**Інформаційні процеси у комп'ютеризованих
вимірювальних системах. Застосування GDI+ та
Windows Forms. 188**

Д О Д А Т К И

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	<i>Екземпляр № 1</i>	<i>Арк __ / 5</i>

Ч А С Т И Н А І

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 6

Лабораторна робота № 1

Ознайомлення з середовищем розробки Microsoft Visual C++

Мета роботи

Здобути навички використання середовища розробки Microsoft Visual C++, створити проект, що містить консольну програму, ознайомитися зі структурою проекту Visual C++, структурою консольної програми, здобути навички написання, компіляції, виправлення помилок та відлагоджування програм.

Порядок виконання роботи.

1. Створити проект консольної програми Win32, яка виконує математичні операції.

1. Виконати оголошення та ініціалізацію декількох змінних основних базових типів, що слугуватимуть у якості вихідних даних.
2. Написати набір команд дій над оголошеними змінними, в яких використовуються операції, вказані у табл.1.
3. Відкомпілювати програму та виправити помилки у разі їх наявності. Сповіднення про помилки та пояснення причин їх виникнення за протоколювати у звіті.

Таблиця 1.

Типи математичних операцій над даними

+	*	++	<<	&	?:	*= /=	+= -=	<=	&=	sizeof
-	/	--	>>	^		%=		>=	^= =	
	%								<<=	
									>>=	

- Виконати програму у покроковому режимі, перевіrivши значення всіх змінних у процесі, пересвідчившись у правильності роботи програми (в коментарі до кожного рядку записати очікуване значення змінної). Записати у звіт спосіб запуску та роботи у покроковому режимі.
- Виконати виведення імен та значень вихідних даних у консоль. Назви, знаки-оператори операцій над змінними та результати виконання операцій над змінними також вивести у консоль. Наприклад:

```

*****
Вихідні дані операції «+»:
a = 11
b = 12
Результат виконання операції a + b = 23
*****

```

Короткі теоретичні відомості до роботи.

Інтегроване середовище розробки (IDE) Visual Studio надає набір інструментів, які допомагають у створенні та модифікації коду, виявленні та

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 8

виправлення помилок. При написанні програми на Visual C++ за допомогою Visual Studio першим етапом роботи є вибір типу проекту. Для кожного типу проекту Visual Studio встановлює параметри компілятора і генерує стартовий код.

Створення нового проекту:

1. У меню Файл виберіть команду Створити.
2. В діалоговому вікні створення потрібно обрати закладку створення проекту.
3. В області Типи проектів виберіть пункт Консольна програма Win32 (Win32 Console Application).
4. Введіть назву проекту (наприклад lab1) та місце розміщення проекту (можна перейти до каталогу, в який потрібно зберегти проект або прийняти місце розташування за замовчуванням).
5. Для запуску майстра додатків Win32 натисніть кнопку ОК.
6. У полі вибору типу програми виберіть параметр простий проект (simple application) та натисніть кнопку Готово.

В результаті виконання всіх вище згаданих операцій буде створено проект з файлами (наприклад lab1.cpp).

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 9

Використання оглядача робочого простору (Workspace Explorer).

Оглядач робочого простору спрощує роботу з файлами, класами та іншими ресурсами, включеними до рішення. За допомогою оглядача простору спрощує можна додавати, видаляти або доступатися до файлів які входять до проекту, створювати, переглядати класи та функції. За замовчуванням Workspace містить дві закладки на яких відображається дерево проектів з усіма проектами та файлами, які входять до проектів (FileView) та дерево класів та глобальних функцій (ClassView). Якщо розкрити всі гілки дерева файлів тоді можна отримати уявлення про всі складові проекту. У випадку якщо при створенні проекту було введено назву lab1 тоді у файлі lab1.cpp міститься згенерований текст-заготовка програми:

```
#include "stdafx.h"
int main(int argc, char* argv[]) {
    return 0;
}
```

Написання програмного коду

Для написання програмного коду використовується вбудований в інтегроване середовище редактор коду.

Структура програми

Програма на мові C++ складається з препроцесора директив, вказівок компілятору, оголошень змінних та/або констант, оголошень і визначень функцій.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 10

```
#include "stdafx.h"
```

```
// препроцесор директив
```

```
int main(int argc, char* argv[])
```

```
// оголошення головної програми як
```

```
// функції
```

```
{ // відкриваюча дужка основної програми
```

```
// замінити цей коментар
```

```
// оголошенням змінних та констант
```

```
// замінити цей коментар
```

```
// розміщенням програмних
```

```
// інструкцій користувача
```

```
return 0;
```

```
// обов'язковий кінцевий оператор виходу з програми
```

```
// та надсилання до
```

```
// операційної системи статусу результату
```

```
// виконання програми (0 - виконано успішно)
```

```
} // закриваюча дужка основної програми
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 11

Оголошення змінної

Оголошення змінної задає ім'я та атрибути змінної. Атрибутами змінної можуть бути тип, кількість елементів (для масивів), специфікація класу пам'яті, а також ініціалізатори. **Ініціалізатор** - це константа відповідного типу, що задають значення, яке надається змінній при створенні.

Типово оголошення змінної має наступний синтаксис:

<тип> <ім'я>;

Приклади оголошення змінних

```
int x; // Оголошення змінної цілого типу
int a = 0, b = 0; // Змінні цілого типу, що
                ініціалізуються початковими
                // нульовими значеннями.
const float b = 1.0, // Оголошення змінної типу десятковий
f = 0.0; // дріб
double y = exp(1); // Змінна дробового типу, що
                ініціалізується числом e.
                // exp (x) - функція експоненти у
                ступені x.
```

У мові C++ немає обмежень на кількість символів в імені.

Приклади типових операцій над змінними

Оператор операції	Приклад використання у програмі
+ –	(a + b); // звичайна сума (a – b); // звичайна різниця
* / %	(a * b); // звичайний добуток (a / b); // звичайна частка (a % b); // незвична операція, результат якої // це цілий залишок від цілочислового ділення (a / b) // - цю операцію називають ділення по модулю
++ --	a++; // те саме, що і a = a + 1 ++a; // те саме, що і a = a + 1 b--; // те саме, що і b = b - 1 --b; // те саме, що і b = b - 1
<<= =>> >> <<	// наприклад v = 10, у двійковій системі це 0000 1010 v >>= 1; // операція зміщення на 1 розряд вправо // нове v буде 5, у двійковій системі 0000 0101 v <<= 1; // тепер v знову рівне 10, у двійковій системі 0000 1010 v <<= 2; // тепер v рівне 40, у двійковій системі 0010 1000 (v >> 2); // результат операції 10, але v залишається рівним 40
& ^ 	a & b; // порозрядна операція логічного І a ^ b; // порозрядна операція логічного ВИКЛЮЧАЮЧОГО АБО

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
Екземпляр № 1		Арк. __ / 13
	a b; // порозрядна операція логічного АБО	
?:	(a) ? (b = 3) : (c = 1); // виконуватиметься операція b = 3 // якщо a > 0, або a рівна true // інакше виконуватиметься операція c = 1 // можна передбачити набір операцій	
=	a = ((a + b) + (1 + b) * 5); // звичайне присвоювання результату // обчислення виразу, який справа, змінній з // іменем «a», яка зліва b = a++; // те саме, що і b = a, і потім a = a + 1 b = ++a; // те саме, що і a = a + 1, і потім b = a b = a--; // те саме, що і b = a, і потім a = a - 1 b = --a; // те саме, що і a = a - 1, і потім b = a	
*= /=	a *= b; // те саме, що і a = a * b (результат дії (a * b) записується у // змінну a	
%= += -=-	a /= b; // те саме, що і a = a / b a %= b; // те саме, що і a = a % b a += b; // те саме, що і a = a + b a -= b; // те саме, що і a = a - b	
<= >=	(a <= b); // результат true , якщо a < або = b (a >= b); // результат true, якщо a > або = b	

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 14
&= ^= = <<= >>=	(a &= b); // те саме, що і a = a & b (a ^= b); // те саме, що і a = a ^ b (a = b); // те саме, що і a = a b (a <<= 5); // те саме, що і a = (a << 5) (a >>= 5); // те саме, що і a = (a >> 5)	
sizeof	sizeof(a); // операція, результат якої рівний числу байт, які займає операнд a	

Приклад програми, що виконує елементарні дії над змінними

```
#include "stdafx.h"
```

```
int main()
```

```
{ // відкриваюча дужка основної програми
```

```
// далі інструкції користувача
```

```
// оголошення змінних
```

```
int a, b;
```

```
float operand_a, operand_b, result_c;
```

```
// елементарні дії над змінними
```

```
a = b + a;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 15

```
a = ++b;
```

```
a = b++;
```

```
result_c = operand_a / operand_b;
```

```
return 0;
```

```
// обов'язковий кінцевий оператор виходу з програми
```

```
// та надсилання до
```

```
// операційної системи статусу результату
```

```
// виконання програми (0 - виконано успішно)
```

```
} // закриваюча дужка основної програми
```

Компіляція, повідомлення про помилки, виправлення помилок та запуск програми.

Для того щоб відкомпілювати проект потрібно вибрати пункт меню Build\Build <назва проекту> або F7 після чого на закладці Build у вікні Output (яке за замовчуванням знаходиться в нижній частині середовища розробки) буде відображатися інформація про хід процесу компіляції. Якщо в програмному коді відсутні помилки, тоді в результаті компіляції отримуємо повідомлення:

```
Compiling...
```

```
lab1.cpp
```

```
Linking...
```

```
lab1.exe - 0 error(s), 0 warning(s)
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 16

Якщо, все ж таки, в програмному кодї є помилки тоді на закладці Build у вікні Output буде вміщуватися опис помилок (назва файлу та рядок в якому присутня помилка, код помилки та короткий опис помилки) та повідомлення про те, що створити файл lab1.exe не вдалося, як наприклад:

```
Compiling...
lab1.cpp
D:\lab1\lab1.cpp(10) : error C2143: syntax error :
missing ';' before 'return'
Error executing cl.exe.

lab1.exe - 1 error(s), 0 warning(s)
```

Подвійне натискання на повідомленні про помилку на закладці Build у вікні Output призводить до відкриття файлу з помилкою та встановленні курсору у рядок з помилкою.

Після виправлення помилок та вдалої компіляції можна запустити створений файл на виконання, для чого потрібно вибрати пункт меню Build\Execute <назва проекту> або Ctrl+F5.

Налагодження програмного коду.

Налагодження – етап, який присутній при написанні будь-якої досить складної програми, та який є необхідним у випадку, якщо в результаті виконання програми було отримано неочікуваний результат або поведінка програми цілком є незрозумілою. Призначення даного етапу полягає у тому, що програміст локалізує місце знаходження помилок у програмі шляхом аналізу

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. ___ / 17

змінних та порядку виконання команд. Для того, щоб виконувати налагодження потрібно запустити програму в режимі налагодження, це можна зробити за допомогою пункту меню Build\Start Debug\Go або F5.

В даному режимі можна виконувати кожен рядок програми крок за кроком, встановлювати точки зупинки програми, заходити в середину функцій, переглядати значення змінних, реєстрів, стеку тобто всього того, що потрібно для перевірки коректності написаного програмного коду:

1. При досягненні програмою рядка з встановленою точкою зупинки виконання програми тимчасово припиняється (програма знаходиться в режимі призупинення виконання). Рядок коду, який виконується в даний момент відмічено жовтою стрілкою ліворуч від рядка.
2. Щоб дізнатися поточне значення змінної наведіть на неї курсор, ім'я змінної та її значення, відобразиться у вікні спливаючі підказки. Для того, щоб переглянути значення за змінної у вікні Watch можна виділити змінну і перетягнути її в дане вікно.

Як приклад налагодження програми можна навести наступну послідовність дій:

1. Відкрити файл, що містить можливі логічні помилки.
2. Встановити точки зупинки в рядках де потрібно виконувати налагодження (встановити або видалити в поточному рядку точку зупинки можливо за допомогою клавіші F9). Ліворуч від рядка з встановленою точкою зупинки з'являється зафарбоване червоним кольором коло.
3. Запустити програму в режимі налагодження.
4. Після зупинки виконання в точці зупинки переглянути значення потрібних змінних на закладках Auto або Watch.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 18

5. Продовжити виконання програми в покроковому режимі (F10 – Step Over або F11 – Step Into), якщо є необхідність перевірки послідовності виконання команд або до наступної точки зупинки (F5).
6. Якщо виникає потреба в зупинці виконання програми потрібно натиснути сполучення клавіш SHIFT + F5.

Організація обчислення та виведення даних у консоль

Наведемо приклад програми, в якій відбувається виведення текстового рядку у консоль.

```
#include "stdafx.h"

#include <iostream>
    // доручення до програми файлу бібліотеки
    // функцій (введення)виведення даних
    // (з)на різні носії та інтерфейси

using namespace std;
    // дозвіл використовувати короткі імена
    // стандартних функцій (про всяк випадок)

int main(int argc, char* argv[])
{
    cout << " Лабораторна робота № 1 ";
    // cout - означає консоль
    // cout << - означає направлення
    // виводу рядка, що у "" на
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 19

```
// КОНСОЛЬ
```

```
cout << "\n"
// невидимий символ керуючої послідовності,
// що переводить курсор консолі у новий
// рядок

cout << " студента гр. АТ - 25 Харченка Богдана ";

int result_c,
int operand_a = 30, operand_b = 44;

result_c = operand_a / operand_b;

// виведення у консоль вихідних даних

cout << "Змінна a = " << operand_a;
cout << "\n";
cout << "Змінна b = " << operand_b;
cout << "\n";

// виведення у консоль результату

cout << "Результат операції (a + b)" << result_c;
cout << "\n";

return 0;
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 20

Лабораторна робота № 2

Циклічні обчислення

Мета роботи.

Здобути навички використання командних інструкцій виведення даних **cout**, введення даних **cin**, циклу **do-while**, **while**, **for** та перевірки умов **if**, **?:** при створенні програм ітераційних обчислень у середовищі розробки Microsoft Visual C++ 6.0.

Порядок виконання роботи.

1. Створити проект консольної програми Win32, яка виконує операцію розрахунку таблиці значень заданої функції.

1. Скласти програму, яка реалізує алгоритм обчислення суми послідовності значень математичного виразу Y (табл. 2) в заданому діапазоні значень аргументу X :
 - a. Результати обчислень виводяться у таблицю у консоль.
 - b. Таблиця повинна включати лише ті значення функції Y , які відповідають певній умові згідно варіанту.
 - c. Підібрати таке значення зміни аргументу X , щоб у таблицю виводилося не більше 50 результатів.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 21

d. При виконанні завдання обов'язкове використання умовних операторів та операторів циклу.

- Відкомпілювати програму та виправити помилки у разі їх наявності. Сповіднення про помилки та пояснення причин їх виникнення запротоколювати у звіті.

Таблиця 2.

Варіанти типових алгебраїчних виразів, що зустрічаються на практиці

№	Вираз Y	Діапазон значень X	Умова включення значення Y до таблиці
1.	$y = 5.1 * x - 3 / (x^2 + 1)$	-50..50	$0 < y < 10$
2.	$y = 1.1 * x - 3*x*\sin(x) + 1.2$	-7..7	$-1 < y < 1$
3.	$y = \ln((x^2 + 1)/(x^2 - 1)) + 1.2*x$	-15..15	$ y < 5$
4.	$y = x^4 + 5*x^3 - x^2 + 120$	-25..25	$y \% 7 == 6$
5.	$y = x ((x << 3) + (x << 1) + \sim x)$	-127..127	$(y \& 37) <= 5$
6.	$y = 0.01*\exp(-0.5*x) + 0.05*x^3$	-10..20	$y < 0 y > 90$
7.	$y = 4.3*\ln(x + 1) / (0.05*x^2 - 1)$	-0.99..19.9	$-1 < y < 0.5$
8.	$y = x^5 - 5*x^3 + x + 37$	-20..20	$(y \% 13) == 5$

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015		Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1		Арк ___ / 22
9.	$y = (x - 2.1) / (x^2 + x - 2.1)$	-30..10	$y < -0.1 \ \ y > 0.1$
10.	$((x << 3) + (x << 1) x) ^ x$	-64..64	$(y \& 0x110) != 0$
11.	$y = \ln(x^2) + 3 * x / \exp(0.5*x)$	-15..20	$0 < y < 5$
12.	$y = 2*x*\cos(x) + 3*x*\sin(x) - x$	-25..25	$y < -30 \ \ y > 10$
13.	$y = (x^2 - 5) \% 13 - (x^3 + x) \% 7$	-10..20	$y == 5$
14.	$y = x^4 - 10.2*x^3 - x^2 + 2.1*x + 37.7$	-20..10	$y < 2*x$
15.	$y = 1/x - 1/x^2 + 1/x^3 - 1/x^4$	1..10	$y < 0.1$
16.	$y = ((x << 1) (x << 2) \& (x >> 3)) x$	-64 .. 64	$(y \& 0x30) < 48$
17.	$y = x^4 - 3*x^3 + 3*x^2 + 7$	0..30	$(y \% 11) < 5$
18.	$y = 128*x^4 - 13*x^2 + 3*x*\sin(x)$	- 6.28..6.28	$ y < 100*x$
19.	$y = (x - \sin(x)) / (x^2 - 15)$	-10..10	$y < -0.5 \ \ y > 0.5$
20.	$((x << 3) + (x >> 1) \& x) x$	-64..64	$(y \& 0xf0) == 0x30$
21.	$y = 22*\ln(x^2 + x + 2.8) - 5.2 * x^2$	-15..10	$0 < y < 50$
22.	$y = 2*\exp(-0.1*x) + 3*x*\sin(x)*\exp(-0.5*x)$	0..50	$y < 0.05$
23.	$y = (x^2 - 5) \% 13 + x^2 \& x$	0..100	$(y \% 5) > 2$
24.	$y = x^2 - 1.2*x + 11 - 2.1/x + 3.7/x^2$	5..50	$y < 30*x$
25.	$y = \ln(x) - 1/(x^2 + 1.2*x^3 + 1)$	1..40	$0 < y < 3$

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015		Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1		Арк __/23
26.	$y = x \& ((x \ll 3) + (x \ll 1))$	-127..127	$(y \mid 17) \geq 10$
27.	$y = 3.1 * \text{tg}(-0.5 * x) + 5 * x^2 - 17$	-20..20	$y < 0 \mid \mid y > 300$
28.	$y = (x \ll 1) \mid (x \ll 2) \& (x \gg 3) \mid x$	-64 .. 64	$-20 < y < 20$
29.	$y = x^2 + (x + \sin(x)) / (x - \sin(x))$	-10..10	$-10 < y < 10$
30.	$y = (7 * x^2 - 12 * x) \% 41 - x$	-100..100	$ y - x < 5$
31.	$y = (15 * x^3 - x^2 + 11 * x - 12) \% 127$	-100..100	$-17 < y < 17$

3. Виконати виведення вихідних даних у консоль. Оператори математичних дій над змінними та результати виконання операцій над змінними також вивести у консоль.

Наприклад:

```

*****

Лабораторна робота студента групи АТК-10 ФІКТ Добржанського О.О.

*****

Результат ітераційного обчислення значення функції:

y(x) = x*x*x + log10( |x| ) + ln( |x| ) + sin(x) + |x|

на інтервалі значень x = -6.28 .. + 6.28

за умови, що y(x) буде у межах -70 .. +70

*****

```

Введіть величину кроку зміни x : 0.1

Введіть кількість значущих цифр для відображення $y(x)$: 4

Введіть ширину поля таблиці у знаках: 10

№з.п.	x	$y(x)$
-------	-----	--------

1	-6.28	$y(x)$ за межами
---	-------	------------------

2	-6.18	$y(x)$ за межами
---	-------	------------------

3	-6.08	$y(x)$ за межами
---	-------	------------------

...

43	3.52	48.57
----	------	-------

44	3.62	52.44
----	------	-------

45	3.72	56.54
----	------	-------

46	3.82	60.86
----	------	-------

47	3.92	65.41
----	------	-------

48	4.02	$y(x)$ за межами
----	------	------------------

49	4.12	$y(x)$ за межами
----	------	------------------

50	4.22	$y(x)$ за межами
----	------	------------------

...

Виведення таблиці закінчено

Для продовження натисніть будь-яку клавішу . . .

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 25

4. Виконати програму у покроковому режимі, перевіряючи значення всіх змінних у процесі, пересвідчившись у правильності роботи програми (в коментарі до кожного рядку записати очікуване значення змінної). Записати у звіт спосіб запуску та роботи у покроковому режимі.

Короткі теоретичні відомості до роботи.

Підготовка програми до введення виведення

! Не забуваємо підключити відповідний файл заголовку:

```
#include <iostream>
```

та визначити інструкцію

```
using namespace std;
```

Оператор виведення даних від користувача у консоль

```
cout << y ; //виведення значення змінної y
cout << "\n"; //перехід на новий рядок
cout << "Введіть друге ціле число\n"; //виведення
        рядку-запрошення
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 26

Оператор введення даних від користувача з консолі

```
int a;
cin >> a; //читання цілого
float b;
cin >> b; //читання дробу
char b;
cin >> b; //читання 1-го символу
```

`cin` використовує операцію “узяти з потоку” `>>`, щоб отримати від користувача значення і присвоює це значення змінній `a`.

Коли комп'ютер виконує попередній оператор, він чекає від користувача введення значення змінної `a`. У відповідь користувач набирає на клавіатурі ціле число і потім натискає *клавішу повернення* — **return** (що називається іноді *клавішею введення* — **enter**), щоб послати це число в комп'ютер. Комп'ютер потім привласнює це число, або *значення*, змінній `a`.

Організація аналізу даних

Перевірку виконання певних умов можливо здійснити за допомогою оператора умови.

Перебір всіх даних на певному діапазоні можливо здійснити за допомогою оператора циклу.

Подамо пояснення та приклади таких операторів.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 27

Покроковий оператор циклу

```
for ([<початковий вираз>; [<вираз умова>]; [<вираз збільшення >]) <оператор> ;
```

Тіло оператора **for** (<оператор>) виконується до тих пір, поки значення <вираз умова> не стане хибним (рівним 0). <Початковий вираз> використовуються для ініціалізації значень, які використовуються під час організації циклу. <вираз збільшення> зазвичай використовуються для модифікації параметрів циклу та інших значень. <початковий вираз> обчислюється тільки один раз до першої перевірки <вираз умова>, а <вираз збільшення> обчислюється після кожного виконання <оператора>. Будь-який з трьох виразів заголовку циклу, і навіть всі три можуть бути пропущені. Якщо пропущено <вираз умова>, то його значення вважається істинним, і цикл стає нескінченним.

Приклад обчислення суми цілих чисел в діапазоні [10, 35):

```
int sum = 0;
for( int a = 10; a < 35; a++ )
{
    // інструкції, що мають виконуватись у циклі
    sum += a;
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 28

Оператор циклу з передумовою

while (<вираз>) <оператор> ;

Оператор **while** використовується для організації циклів з перевіркою умови (<вираз>) перед виконанням дій (<оператор>) в циклі. Як правило <вираз> називають умовою виконання циклу, а <оператор> тілом циклу. Наведемо приклад використання **while** для обчислення суми чисел в діапазоні [a, b).

```
int sum = 0;
int a = 10, b = 35;

while( a < b )
{
    // інструкції, що мають виконуватись у циклі
    sum += a; a++;
}
```

Оператор циклу з після-умовою

```
int sum = 0;
int a = 10;

do
{
    // інструкції, що мають виконуватись у циклі
    sum += a; a++;
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 29

}

```
while( a < b )
```

Деякі функції для виконання математичних обчислень

! Не забуваємо підключити відповідний файл заголовку:

```
#include "math.h"
```

! Не забуваємо, що змінні для математичних функцій мають тип переважно

```
double a;  
long double d; // точніший за double
```

Деякі математичні функції дозволяють роботу зі змінними типу

```
float aa; // менш точний за double
```

Основні математичні функції

```
double res;
```

```
res = sin(a); // синус кута, що заданий в радіанах
```

```
res = cos(a); // косинус кута, що заданий в радіанах
```

```
res = tan(a); // тангенс кута, що заданий в радіанах
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 30

```
res = log(a); // натуральний логарифм ln
```

```
res = exp(a); // функція експоненти, число e у степені a
```

Оператор умови

```
if (<умова>) <оператор 1> [else <оператор 2>];
```

Даний оператор використовується для перевірки виконання умови (операція порівняння), яка задається у <умова>. Оператор **if** виконує деякі дії визначених у <оператор 1>, якщо <умова> виконується. Оператор виконує <оператор 2>, якщо <умова> не виконується.

Для прикладу використаємо умовний оператор визначення максимального значення двох чисел.

```
int a = 3, b = -5, max;
```

```
if (a > b)
{
    max = a;
}
else
{
    max = b;
}
```

Приклади операцій порівняння

Оператор операції	Приклад використання у програмі
<	(a < b) // вираз рівний true, якщо a менше b
>	(a > b) // вираз рівний true, якщо a більше b
<=	(a <= b) // вираз рівний true, якщо a менше, або дорівнює b
>=	(a >= b) // вираз рівний true, якщо a більше, або дорівнює b
==	(a == b) // вираз рівний true, якщо a дорівнює b
!=	(a != b) // вираз рівний true, якщо a не дорівнює b
відсутній	(a) // вираз буде сприйнятий як true, якщо a не рівна 0

Забезпечення кириличного шрифту у консолі

1) Прописати серед підключених файлів заголовку

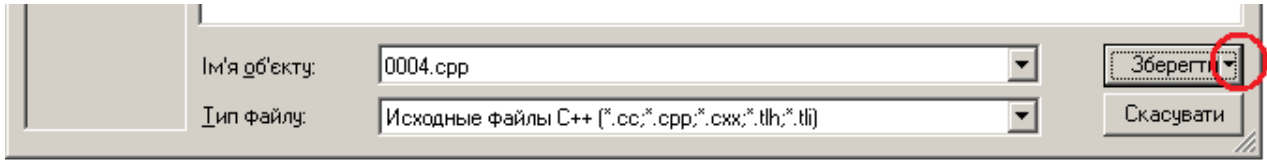
```
#include "windows.h"
```

2) Прописати на початку головної програми інструкцію встановлення кодової сторінки при виводі тексту у консоль:

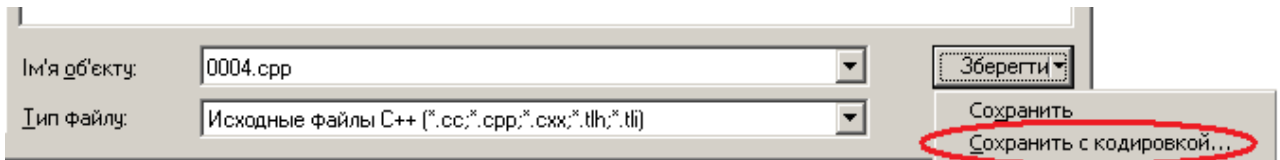
```
SetConsoleOutputCP(1251); // відповідає кодуванню ASCII
```

3) Файл -> **Зберегти** Назва_проекту . сpp як ->

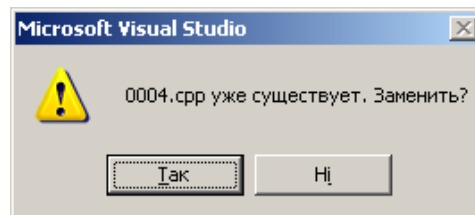
4) Обрати стрілку вибору способу збереження:



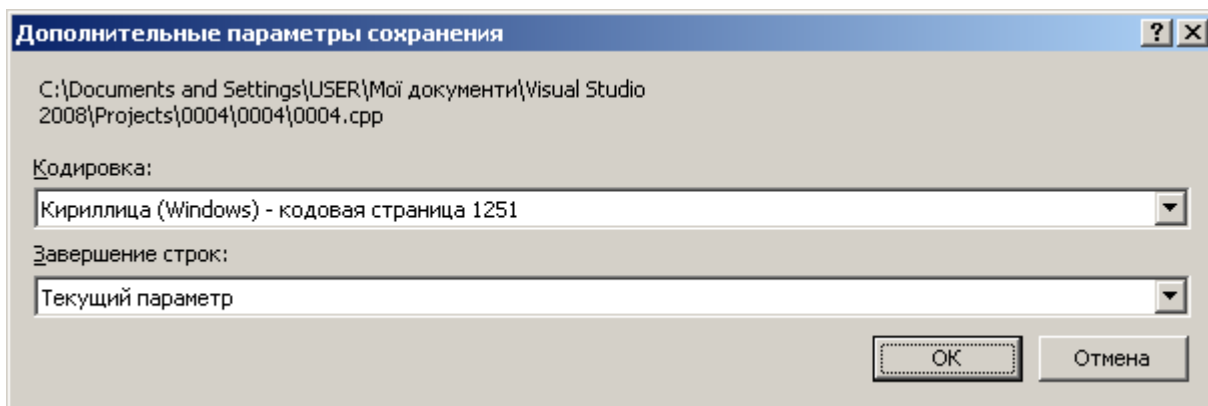
5) Обрати збереження з кодуванням:



6) Підтвердити
заміну існуючого
файлу:



7) Обрати кодування 1251 (що відповідає кодуванню ASCII):



8) Перекомпілювати проект (рішення)

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 33

Лабораторна робота № 3

Математичні операції та організація меню користувача

Мета роботи.

Здобути навички використання командних інструкцій виведення даних **cout**, введення даних **cin**, циклу **do-while** (або **while**, або **for**) та командної інструкції перемикачів варіанту дій **switch-case**, при створенні програм виконання обчислень за вибором користувача і за допомогою меню користувача, у середовищі розробки Microsoft Visual C++ 6.0.

Порядок виконання роботи.

Створити проект консольної програми Win32, яка виконує математичні операції розрахунку значення заданої функції відповідно до вибору користувача і за допомогою меню користувача.

Скласти програму, яка реалізує алгоритм обчислення певних математичних функцій (табл. 3) відповідно введеним аргументам:

- а. У консоль виводяться поточні значення аргументів (змінні *a* та *b*)

- b. В консоль виводиться меню користувача, за допомогою якого користувач обирає яку дію виконувати:
- i. введення початкових даних (змінні a та b)
 - ii. обчислити математичну функцію з переліку відповідно до варіанту (табл. 3).
 - iii. отримати інформацію про розробника програми
 - iv. виконати вихід з програми
- c. Результати обчислень виводяться у консоль після введення вибору користувача та його підтвердження клавішею Enter.
- d. Після виведення результату програма очікує натиснення будь-якої клавіші програма очищує екран та знову виводить поточні початкові дані (змінні a та b), меню та запит на введення вибору користувача.

Відкомпілювати програму та виправити помилки у разі їх наявності. Сповіднення про помилки та пояснення причин їх виникнення запротоколювати у звіті.

Таблиця 3.

Типи математичних операцій над даними

Функції, що доступні, за умови включення «math.h»														
№ з.п.	$a * \cos(b)$	$a * \sin(b)$	$a * \operatorname{atan}(b)$	$\operatorname{pow}(a,b)$	$a * \log(b)$	$\operatorname{atan2}(b/a)$	$b * \operatorname{floor}(a)$	$b * \operatorname{sqrt}(a)$	$b * \operatorname{asin}(a)$	$b * \cos(a)$	$b * \tan(a)$	$b * \log_{10}(a)$	$b * \exp(a)$	$b * \operatorname{ceil}(a)$
1.	acos	sin	atan	pow										
2.			atan	pow	log	atan2								
3.					log	atan2	floor	sqrt						
4.							floor	sqrt	asin	cos				
5.									asin	cos	tan	log10		

Житомирська політехніка		МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015										Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022		
		Екземпляр № 1										Арк ___ / 35		
6.											tan	log10	exp	ceil
7.	acos	sin											exp	ceil
8.		sin	atan									log10	exp	
9.			atan	pow							tan	log10		
10.				pow	log					cos	tan			
11.					log	atan2			asin	cos				
12.			atan	pow									exp	ceil
13.				pow	log							log10	exp	
14.					log	atan2					tan	log10		
15.						atan2	floor			cos	tan			
16.	acos		atan		log		floor							
17.		sin		pow		atan2		sqrt						
18.			atan		log		floor		asin					
19.				pow		atan2		sqrt		cos				
20.					log		floor		asin		tan			
21.						atan2		sqrt		cos		log10		
22.							floor		asin		tan		exp	
23.	acos			pow			floor			cos				
24.		sin			log			sqrt			tan			
25.			atan			atan2			asin			log10		
26.				pow			floor			cos			exp	
27.					log			sqrt			tan			ceil
28.	acos					atan2			asin			log10		
29.	acos				log				asin				exp	
30.		sin				atan2				cos				ceil
31.			atan				floor				tan			ceil

5. Виконати інтерфейс спілкування програми з користувачем дення вихідних даних у консоль. Оператори математичних дій над змінними та результати виконання операцій над змінними також вивести у консоль.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 36

Наприклад, початковий екран:

```

*****
* a= 0  b= 0
*****
* 1: Введення значень змінних:  a та b          *
* 2: (a*b)   3: (a-b)   4: (++b)   5: (b++)      *
* 6: Exit    i: Info                                     *
*****
>

```

Наприклад, екран введення початкових даних:

```

*****
* a= 0  b= 0
*****
* 1: Введення значень змінних:  a та b          *
* 2: (a*b)   3: (a-b)   4: (++b)   5: (b++)      *
* 6: Exit    i: Info                                     *
*****
>1

Введіть a: 2
Введіть b: 4.5

Press any key to continue . . .

```

Наприклад, екран виведення результату обчислення:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 37

```

*****
* a= 2  b= 4.5
*****
* 1: Введення значень змінних:  a та b          *
* 2: (a*b)   3: (a-b)   4: (++b)   5: (b++)      *
* 6: Exit    i: Info                                     *
*****
>2

(a*b)= 9

Press any key to continue . . .

```

Наприклад, екран виведення інформації про розробника:

```

*****
* a= 2  b= 4.5
*****
* 1: Введення значень змінних:  a та b          *
* 2: (a*b)   3: (a-b)   4: (++b)   5: (b++)      *
* 6: Exit    i: Info                                     *
*****
>i

Програму складено:
студентом ЖДТУ факультету ІКТ
групи АТК-11 Добржанським О.О.
Варіант № 31
© Всі права захищено, 2017р.

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 38

Press any key to continue . . .

Наприклад, екран реакції на неправильний вибір користувача:

```

*****
* a= 2  b= 4.5
*****
* 1: Введення значень змінних:  a та b          *
* 2: (a*b)   3: (a-b)   4: (++b)   5: (b++)      *
* 6: Exit    i: Info                                     *
*****
>d

невірне звертання до меню

Press any key to continue . . .

```

6. Виконати програму у покроковому режимі, перевіряючи значення всіх змінних у процесі, пересвідчившись у правильності роботи програми (в коментарі до кожного рядку записати очікуване значення змінної). Записати у звіт спосіб запуску та роботи у покроковому режимі.

Короткі теоретичні відомості до роботи.

Порядок створення застосування:

- оголосити дві змінні в якості початкових даних для обчислень
- проініціалізувати їх початковими нульовими значеннями

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 39

3. оголосити змінну для зберігання інформації про введений користувачем символ
4. проініціалізувати її символом “пробіл”
5. оголосити нескінченний цикл

У циклі:

6. прописати команду очищення вікна консолі - `system("cls");`
7. вивести у консоль рядок із поточними значеннями двох змінних — початкових даних
8. вивести у консоль меню користувача із інформацією про символи (команди будуть складатись з 1-го символу), які користувач повинен ввести для виклику таких команд:
 - 8.1.виклик запиту на введення значень у змінні початкових даних (“Введення початкових даних”)
 - 8.2.виклик I-ї функції за варіантом
 - 8.3.виклик II-ї функції за варіантом
 - 8.4.виклик IV-ї функції за варіантом
 - 8.5.виклик інформації про створювача (“Інфо”)
 - 8.6.виклик команди (“Exit”)
9. вивести запит на введення користувачем команди (“Введіть команду > ”)
- 10.прописати оператор `switch-case`, що керується змінною, яка містить символ введеної команди користувачем

В операторі `switch-case`:

- 11.прописати інструкції дій програми при різних варіантах введеної команди користувача, а саме:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 40

- 11.1. прописати запити на ведення з консолі значень у дві змінні початкових даних
- 11.2. прописати математичні операції, що реалізують функції відповідно варіанту, та виведення у консоль
- 11.3. прописати виведення у консоль даних про створювача: ВНЗ, факультет, група, ПІБ, рік
- 11.4. прописати команду виходу з програми, у випадку вибору користувачем пункту меню "Exit"
- 11.5. прописати виведення у консоль повідомлення ("Неправильне звертання до меню" або "Такого пункту меню не існує"), у випадку введення користувачем символу, якому не співставлена жодна команда меню

Після оператора switch-case але все ще у циклі:

12. прописати команду паузи до натиснення будь-якої клавіші

Отже програма виконуватиметься за алгоритмом (послідовністю дій):

Початок циклу: очистити вікно консолі -> вивести початкові дані для обчислень -
> вивести меню -> запит на введення обраного пункту меню -> очікувати введення -> виконати обраний пункт меню -> очікувати натиснення будь-якої клавіші -> на Початок циклу

Забезпечення кириличного шрифту у консолі

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 41

1) Прописати серед підключених файлів заголовку

`#include "windows.h"`

2) Прописати на початку головної програми інструкцію встановлення кодової сторінки при виводі тексту у консоль:

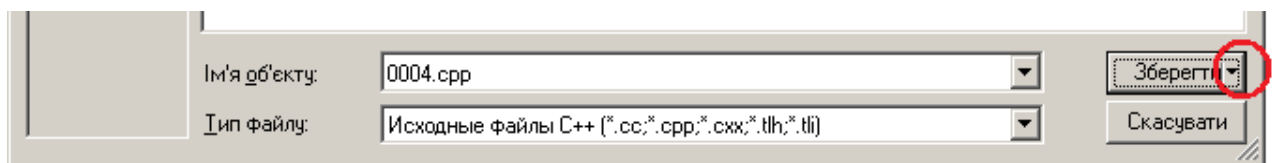
`SetConsoleCP(1251);` // 1251 кодування для символів, що вводяться

`SetConsoleOutputCP(1251);` // 1251 кодування для символів, що виводяться

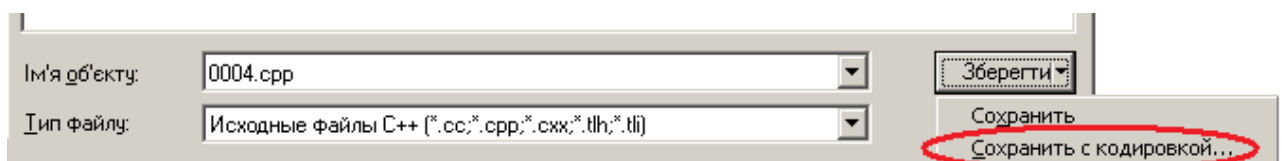
// 1251 відповідає кодуванню ASCII

3) Файл -> **Зберегти** Назва_проекту . cpp як ->

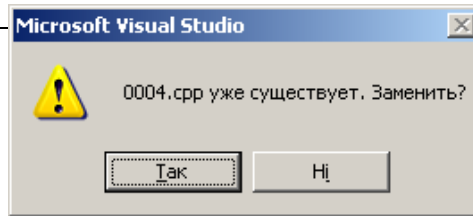
4) Обрати стрілку вибору способу збереження:



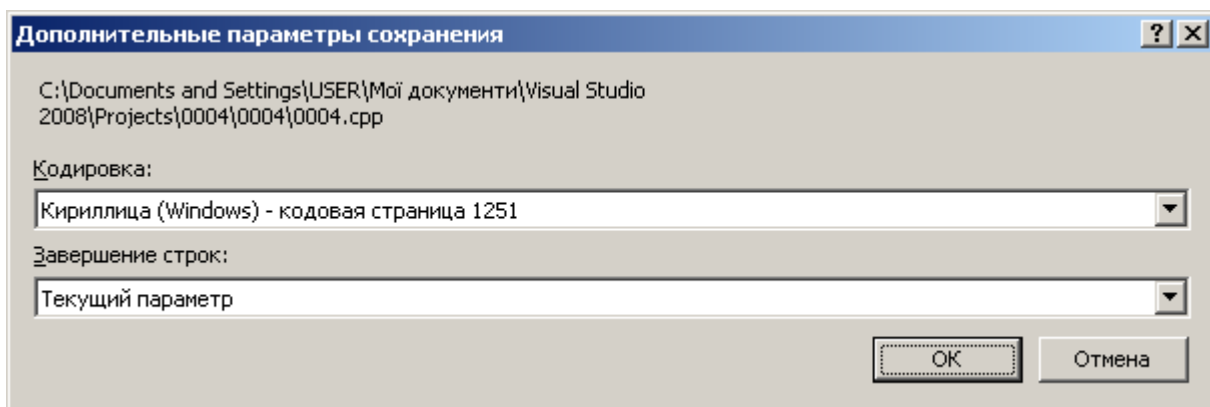
5) Обрати збереження з кодуванням:



6) Підтвердити заміну існуючого файлу:



7) Обрати кодування 1251 (що відповідає кодуванню ASCII):



8) Перекомпілювати проект (рішення)

Підготовка програми до введення виведення

! Не забуваємо підключити відповідний файл заголовку:

```
#include <iostream>
```

```
using namespace std;
```

Оператор виведення даних від користувача у консоль

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 43

```
cout << y ;           //виведення значення змінної y
cout << "\n";         //перехід на новий рядок
cout << "Введіть друге ціле число\n";
//виведення рядку-запрошення
```

Оператор введення даних від користувача з консолі

```
int a;
cin >> a; //читання цілого
float b;
cin >> b; //читання дробу
char b;
cin >> b; //читання 1-го символу
```

Покроковий оператор циклу

```
int sum = 0;

for( int a = 10; a < 35; a++ )
{
    // інструкції, що мають виконуватись у циклі
    sum += a;
}
```

Оператор циклу з передумовою

```
int sum = 0;
int a = 10;

while( a < b )
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 44

```
{
```

```
    // інструкції, що мають виконуватись у циклі
    sum += a; a++;
```

```
}
```

Оператор циклу з після-умовою

```
int sum = 0;
```

```
int a = 10;
```

```
do
```

```
{
```

```
    // інструкції, що мають виконуватись у циклі
    sum += a; a++;
```

```
}
```

```
while( a < b )
```

Оператор-перемикач

```
switch (<вираз>)
```

```
{
```

```
    case <константний вираз 1>:
```

```
    {
```

```
        <оператори 1> ;
```

```
        break;
```

```
    }
```

```
    case <константний вираз 2>:
```

```
    {
```

```
        <оператори 2> ;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 45

```

        break;
    }

    ...

    case <константний вираз N>:
    {
        <оператори N> ;
        break;
    }

    [default:
    {
        <оператори>
    } ]
}

```

Перемикач (**Switch**) оберіє один з декількох варіантів (**case**) виконання дій. Виконання починається з обчислення **<вираз>**, після чого управління передається тому **<оператору>**, значення **<константного виразу>** якого дорівнює значенню **<виразу>**. Інструкція **break** для того, щоб після виконання **<оператору>** для обраного **case** не виконувались **<оператори>**, всі, що розташовані нижче (це особливість **switch-case**).

Якщо значення **<виразу>** не дорівнює жодному **<константному виразу>**, то управління передається **<оператору>**, відміченому ключовим словом **default**, звісно, якщо він є.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 46

//наприклад є змінна Choose (її значення ввів користувач, її тип char – символ)

```
switch (Choose)
{
// якщо Choose = a
    case 'a': { d = 0; break; }

// якщо Choose = b
    case 'b': { d = cos(a) ; break; }

// якщо Choose = z
    case 'z': { d = sin(a); break; }

// якщо Choose не = ні a, ні b, ні c
    default: cout << "Обраний варіант не передбачає
виконання дій";
}

```

Підготовка програми до виконання математичних обчислень

! Не забуваємо підключити відповідний файл заголовку:

```
#include "math.h"
```

! Не забуваємо, що змінні для математичних функцій мають тип переважно

```
double a, res;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 47

Математичні функції, доступні завдяки “math.h”

`res = sin(a); // синус кута, що заданий в радіанах`

`res = sinh(a); // гіперболічний синус кута, що заданий в радіанах`

`res = asin(a); // арксинус числа, результат
//в радіанах від 0 до π`

`res = cos(a); // косинус кута, що заданий в радіанах`

`res = cosh(a); // гіперболічний косинус кута, що заданий в радіанах`

`res = acos(a); // арккосинус числа, результат в радіанах
від $-\pi/2$ до $+\pi/2$`

`res = tanh(a); // гіперболічний тангенс кута, що заданий в радіанах`

`res = tan(a); // тангенс кута, що заданий в радіанах`

`res = atan(a); // арктангенс числа, результат в
// радіанах від $-\pi/2$ до $+\pi/2$`

`res = atan2(a,b); // арктангенс від b/a , результат
//в радіанах від $-\pi$ до $+\pi$`

`res = log(a); // натуральний логарифм \ln від числа a`

`res = exp(a); // функція експоненти,`

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 48

//число e - експонента у степені a

res = log10(a); // десятковий логарифм lg від числа a

res = pow(a, b); // зведення числа a у степінь b

res = fabs(a); // модуль (абсолютне значення) a

res = ceil(a); // найближче більше ціле від числа

res = floor(a); // найближче менше ціле від числа a

res = fmod(a,b); // залишок від ділення
//на цілу кількість частин a на b

res = _hypot(a,b); // корінь з суми (a² + b²)

res = sqrt(a); // корінь з a

Майже всі зазначені математичні функції дозволяють роботу також зі змінними типу

float aa; // менш точний за double

long double d; // точніший за double

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 49

Лабораторна робота № 4

Операції з масивами

Мета роботи:

Здобути навички використання командних інструкцій виведення даних **cout**, введення даних **cin**, циклу **do-while** (або **while**, або **for**) та командної інструкції перемикання варіанту дій **switch-case**, та інструкцій оголошення масивів одномірних та багатомірних масивів даних при створенні програм виконання операцій з масивами даних за вибором користувача і за допомогою меню користувача, у середовищі розробки Microsoft Visual C++.

Порядок виконання роботи.

Створити проект консольної програми Win32, яка виконує операції з масивами даних відповідно до вибору користувача і за допомогою меню користувача.

Скласти програму, яка реалізує роботу зі списком факультетів, списком груп для кожного з факультетів, списком студентів для кожної з груп відповідного факультету:

- а. В консоль виводиться меню користувача із зазначенням таких команд:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 50

- i. показати пояснення до команд: **«help»**
 - ii. вихід з програми **«exit»**
 - iii. зміна списку (**факультети, групи, студенти**): **«change level»**
 - iv. показати перелік елементів списку: **«show»** (повна реалізація у л.р. № 5)
 - v. додати елемент до списку: **«add»**
- b. Після меню виводиться рядок про поточний обраний список:
- наприклад: **«ОБРАНО СПИСОК: ФАКУЛЬТЕТИ»**
- c. Результати обробки даних виводяться у консоль після введення вибору користувача та його підтвердження клавішею Enter.
- d. Після виведення результату програма очікує натиснення будь-якої клавіші, потім очищує екран та знову виводить меню, обраний рівень та запит на введення команди (її номеру) користувачем.

Виконати інтерфейс спілкування програми з користувачем відповідно прикладу у файлі: **«lr_04_приклад.exe»**.

Короткі відомості, щодо порядку виконання роботи:

I Створити новий консольний проект Win32

II Приєднати файли заголовку та простори імен

Для стандартних функцій та операторів:

```
#include "stdafx.h"
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 51

Для використання операторів введення/виведення: `cin/cout`

`#include <iostream>`

Для встановлення бажаної кодової таблиці символів у консолі:

`#include <windows.h>`

Для використання рядкових змінних як окремих так і їх масивів:

`#include <string>`

Для використання скорочених командних інструкцій:

`using namespace std;`

III Оголосити потрібні змінні

Оголошуємо змінні після інструкції початку головної програми **`main { :`**

Рядкову змінну для збереження типу обраного списку («ФАКУЛЬТЕТИ», «ГРУПИ», «СТУДЕНТИ»):

`string sLevel;`

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 52

Символьну змінну для збереження введеного користувачем символу команди з меню («1», «2», «+», «-», «3», «4», «5», «6», «7», «8»):

char chChoose;

Рядкову змінну для збереження рядку даних введених користувачем (назва факультету, назва групи, ім'я студента):

string sInput;

1 вимірний масив для збереження даних про назви факультетів (I вимір – це номери факультетів, а у комірках масиву – рядки з назвами факультетів)

string sDepartmentsNames[10];

2 вимірний масив рядків для збереження даних про назви груп по факультетам (I вимір – це номери факультетів, II вимір – це номери груп, а у комірках масиву – рядки з назвами груп)

string sGroupsNames[10][50];

3 вимірний масив рядків для збереження даних про імена студентів за групами по факультетам (I вимір – це номери факультетів, II вимір – це номери груп, III вимір – це номери студентів, а у комірках масиву – рядки імен студентів):

string sStudentsNames[10][50][50];

Ціла змінна для збереження поточної кількості факультетів:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 53

int iDepartmentsNum;

1 вимірний масив для збереження поточної кількості груп по факультетам (I вимір – це номери факультетів, а у комірках масиву – цілі значення кількості груп):

int iGroupsNum[10];

2 вимірний масив для збереження кількості студентів за групами та по факультетам (I вимір – це номери факультетів, II вимір – це номери груп, а у комірках масиву – цілі значення кількості студентів):

int iStudentsNum[10][50];

Змінні цілого типу для використання у циклах:

int i, j;

Змінна цілого типу для збереження № поточного обраного факультету:

int iChoosedDepartmentNum;

Змінна цілого типу для збереження № поточної обраної групи:

int iChoosedGroupNum;

Змінна цілого типу для збереження № поточного обраного студента:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 54

int iChoosedStudentNum;

Нехай початковий стан програми буде передбачати роботу зі писком факультетів. Тому здійснимо ініціалізацію відповідної рядкової змінної **sLevel**:

sLevel = "ФАКУЛЬТЕТИ";

IV Порядок забезпечення сумісності програми з кириличними шрифтами:

Після оголошення змінних прописуємо інструкцію обрання кодової сторінки для даних, що виводяться у консоль, сумісної з кириличним шрифтом (кодова сторінка 1251):

SetConsoleOutputCP(1251);

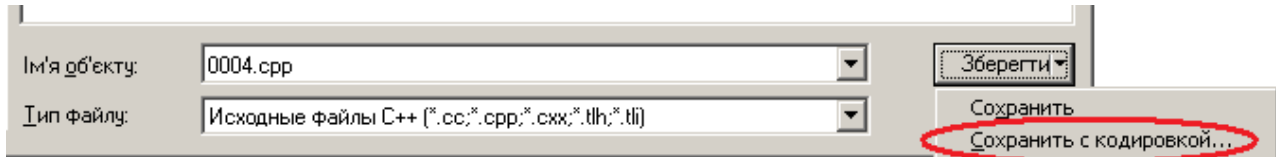
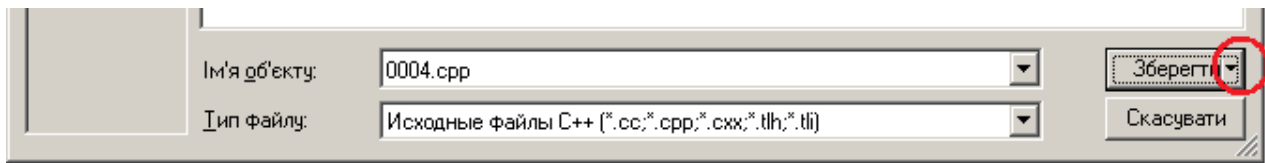
Прописуємо також інструкцію обрання кодової сторінки для даних, що вводяться у консоль користувачем, сумісної з кириличним шрифтом (кодова сторінка 1251):

SetConsoleCP(1251);

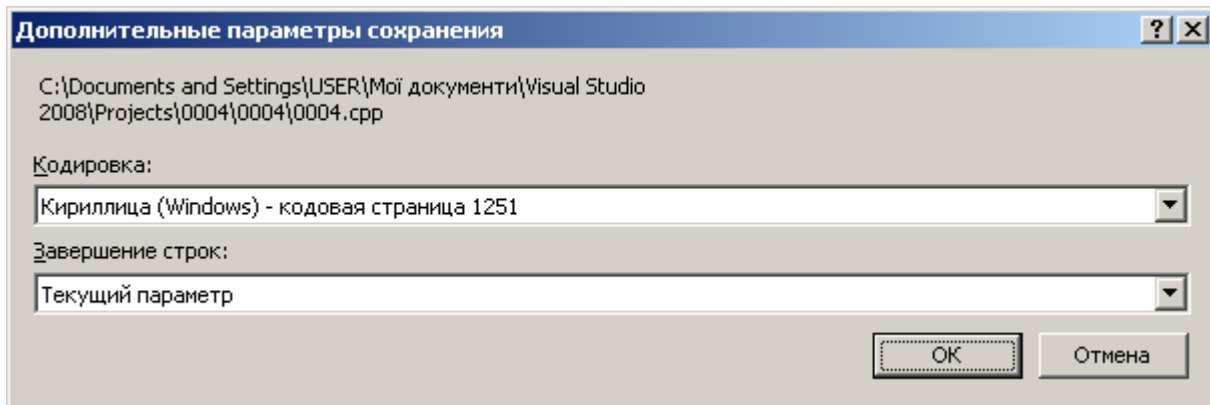
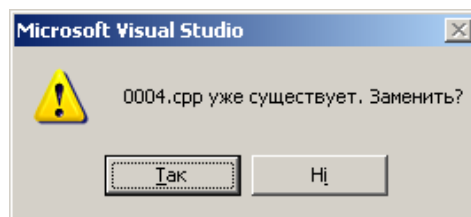
Далі зберігаємо наш файл основного коду проекту у кодах кодової сторінки 1251:

Головне меню -> Файл -> Зберегти Назва_проекту . сpp як ->

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 55



Підтвердити перезапис файлу:



Підтвердити рішення.

Додати команди встановлення для консолі шрифту, що підтримує кодування 1251 (шрифту Lucida Console).

Це дозволить нам не виконувати цей процес вручну після компіляції програми.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 56

Копіюємо цей фрагмент коду після команди **SetConsoleCP(1251)**; поки що без пояснень.

```

CONSOLE_FONT_INFOEX cfi;
int LucidaConsole[] = { 76, 117, 99, 105, 100, 97, 32, 67, 111, 110, 115, 111, 108,
101, 0};
cfi.cbSize = sizeof cfi; cfi.nFont = 0; cfi.dwFontSize.X = 0; cfi.dwFontSize.Y = 12;
cfi.FontFamily = FF_DONTCARE; cfi.FontWeight = FW_NORMAL;
i = 0; while (cfi.FaceName[i] = LucidaConsole[i]){i++;}
SetCurrentConsoleFontEx(GetStdHandle(STD_OUTPUT_HANDLE), FALSE, &cfi);

```

V. Порядок створення основної програми:

Оскільки програма виконуватиметься умовно нескінченно за бажанням користувача, тому оголошуємо нескінченний цикл.

Подумайте до чи після оператора «**return 0;**» основної програми, або до чи після основної програми «**main() { }**» потрібно оголошувати основний нескінченний цикл.

У будь-якому разі нескінченний цикл оголошуємо так:

```

do
    {

```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 57

// тут будуть розміщені ваші командні інструкції, які виконуватимуться

у циклі

}

while (true);

Далі будемо додавати команди до нескінченного циклу:

Спершу маємо вивести у консоль меню користувача.

Для цього спочатку очистимо екран консолі командою:

system("cls");

Далі вже виводитимемо меню. Наприклад, бажаємо, щоб меню мало такий вигляд:

```

+-----+-----+-----+
КОМАНДИ:| 1 help | 2 exit | +/- change level|
+-----+-----+-----+
| 3 show | 4 add  | 5 find | 6 edit | 7 del  |
+-----+-----+-----+

```

Для цього скористаємося блоком команд виведення даних у консоль:

```

cout << "          +-----+-----+-----+<<endl;
cout << " КОМАНДИ:| 1 help | 2 exit | +/- change level|"<<endl;
cout << "+-----+-----+-----+<<endl;
cout << "| 3 show | 4 add  | 5 find | 6 edit | 7 del  |"<<endl;
cout << "+-----+-----+-----+<<endl;

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 58

Далі відповідно завдання має бути виведено рядок поточного обраного списку для дій над ним:

```

+-----+-----+-----+
КОМАНДИ:| 1 help | 2 exit | +/- change level|
+-----+-----+-----+
| 3 show | 4 add  | 5 find | 6 edit | 7 del  |
+-----+-----+-----+
ОБРАНО РІВЕНЬ: ФАКУЛЬТЕТИ
+-----+-----+-----+
>

```

Врахуємо, що обраний список знаходиться у рядковій змінній «sLevel». Отже скористаємося такими командами:

```

cout << " ОБРАНО РІВЕНЬ: " << sLevel << endl;
cout << "+-----+-----+-----+-----+-----+"<<endl;

```

Виведемо у консоль також запит користувачу на введення номеру команди:

```

cout << " > ";

```

Після запиту має слідувати команда передачі введених користувачем даних у змінну.

Врахуємо, що введені команди зберігатимуться у символній змінній «chChoose».

Отже використаємо відповідні інструкції з використанням команди введення даних з консолі:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 59

cin >> chChoose;

Коли користувач ввів номер бажаної команди, потрібно далі обирати дії.

Використаємо оператор вибору дій відповідно значень, що у змінній

«chChoose»:

switch (chChoose)

```

{
    case :
    case :
}

```

Врахуємо типи команд, які можуть бути введені користувачем відповідно меню.

Це команди: «1», «2», «+», «-», «3», «4», «5», «6», «7», «8».

Отже ці варіанти потрібно занести у оператори варіантів **«case»**:

case '1':

case '2':

case '+':

case '-':

case '3':

case '4':

case '5':

case '6':

case '7':

case '8':

default:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 60

Після кожного «**case**» та «**default**» ітимуть набори операторів певних дій.

Тому передбачимо після кожного з них фігурні дужки блоків операторів:

case '1':

```
{
    // тут будуть командні інструкції дій програми,
    // при виборі користувачем команди меню «1»
}
```

case '2':

```
{
    // тут будуть командні інструкції дій програми,
    // при виборі користувачем команди меню «2»
}
```

і так само з іншими «**case**» та з «**default**».

Для того, щоб після виконання операторів по кожному «**case**» не виконувались оператори з інших «**case**», встановлюємо після кожного інструкцію «**break;**».

case '1':

```
{
  

}
```

break;

case '2':

```
{
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 61

}

break;

і так само з іншими «**case**» та з «**default**».

Опишемо оператори дій програми, при виборі команди меню «**1**» («**Help**»).

За це відповідає конструкція:

case '1':

{

}

Оператори розміщатимемо у ній.

Отже, має виводитись список пояснень до команд меню.

Перед виведенням інформації у консоль, очистимо її:

system("cls");

Далі використаємо звичайні оператори виведення у консоль.

cout << "HELP (сторінка 1):"<<endl;

cout << "1 help-описи команд"<<endl;

cout << "2 exit-вихід з програми"<<endl;

cout << "+/- change level-змінити рівень"<<endl;

cout << " + на рівень вище"<<endl;

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 62

```
cout << "          - на рівень нижче"<<endl;
cout << "3 show-список елементів рівня"<<endl;
cout << "4 add-додати елемент рівня"<<endl;
cout << "5 find-знайти елемент рівня"<<endl;
cout << "6 edit-змінити елемент рівня"<<endl;
cout << "7 del-видалити елемент рівня"<<endl;
cout <<endl;
```

Для затримки інформації у консолі до моменту натиснення користувачем будь-якої клавіші використаємо типову команду очікування та виведення сповіщення «Натисніть будь-яку клавішу ...» :

```
system("pause");
```

Подумайте де краще розмістити цю команду.

Рекомендуємо розмістити її у цій же конструкції «**case**» одразу ж після операторів виведення інформації-допомоги.

Опишемо оператори дій програми, при виборі команди меню «**2**» («**Exit**»).

За це відповідає конструкція:

```
case '2':
{

}
```

Оператори розміщатимемо у ній.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 63

Таким чином, необхідно виконати вихід з консольної програми.

Це можна зробити класичним оператором **«return»**:

```
return 0;
```

Опишемо оператори дій програми, при виборі команди меню **«+» («change level»)**.

Тут має відбуватись призначення поточного рівня **«ФАКУЛЬТЕТИ»**, якщо до цього був рівень **«ГРУПИ»**. А якщо поточним був рівень **«СТУДЕНТИ»**, то потрібно змінити його на **«ГРУПИ»**.

Таким чином, якщо ... якщо ... якщо.

Отже використаємо оператори **«if»**.

Розмістимо їх у конструкції:

```
case '+':
```

```
{
```

```
}
```

Врахуємо, що поточний обраний рівень зберігається у рядковій змінній **sLevel**. Її і будемо змінювати.

```
if( sLevel == "ГРУПИ" ){ sLevel="ФАКУЛЬТЕТИ"; }
```

```
else if( sLevel == "СТУДЕНТИ" ){ sLevel="ГРУПИ"; }
```

Аналогічні дії необхідні, при виборі команди меню **«-» («change level»)**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 64

Тільки тут вже має відбуватись призначення поточного рівня **«ГРУПИ»**, якщо до цього був рівень **«ФАКУЛЬТЕТИ»**. А якщо поточним був рівень **«ГРУПИ»**, то потрібно змінити його на **«СТУДЕНТИ»**.

Розмістимо оператори інструкцій **«if»** у конструкції:

case ‘-‘:

{

}

Набір команд **«if»** відповідно має бути такий:

```
if( sLevel == "ФАКУЛЬТЕТИ" ){ sLevel="ГРУПИ"; }
else if ( sLevel == "ГРУПИ" ){ sLevel="СТУДЕНТИ"; }
```

Опишемо оператори дій програми, при виборі команди меню **не передбаченою переліком меню**.

Для цього у операторі вибору варіантів була передбачена конструкція:

default:

{

}

У цій конструкції розмістимо відповідне повідомлення в консоль для користувача та оператор очікування реакції.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 65

```
cout << " така команда відсутня " << endl << " спробуйте ще " << endl;
system("pause");
```

На цьому етапі можливо вже відкомпілювати проект. Та спробувати дії програми на запрограмовані команди.

Розуміємо, що всі інші команди виконуватимуться особливо: в залежності від обраного рівня списку: «ФАКУЛЬТЕТИ», «ГРУПИ», чи «СТУДЕНТИ», що зберігається, нагадаємо, у змінній «**sLevel**».

Тому зробимо заготовки у всіх інших конструкціях «**case**» за прикладом:

```
case '5':
{
    if (sLevel=="ФАКУЛЬТЕТИ")
    {
        тут будуть команди для роботи зі списком факультетів
    }
    else if(sLevel=="ГРУПИ")
    {
        тут будуть команди для роботи зі списком груп
    }
    else if(sLevel=="СТУДЕНТИ")
    {
        тут будуть команди для роботи зі списком студентів
    }
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 66

}

Опишемо дії програми за умови вибору оператором команди «4» («add»).

Так програма повинна буде додати або факультет, або нову групу на факультеті, або нового студента у групі на факультеті.

Опишемо дії програми за умови вибору оператором команди «4» («add»), якщо вже обрано рівень «ФАКУЛЬТЕТИ».

Так програма повинна буде додати новий факультет у список факультетів.

Вже зрозуміло, що додаватимемо команди у конструкцію:

case '4':

```
{
    if (sLevel=="ФАКУЛЬТЕТИ")
    {
        знайдемо це місце і
        сюди додамо команди для роботи зі списком факультетів
    }
    ...
}
```

Команди, які ми передбачимо у цьому випадку, будуть такі:

1) Запит до користувача на введення імені нового факультету:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 67

cout << "Введіть назву нового факультету > ";

2) Очікування відповіді і передача її у змінну для збереження введених рядків:

cin >> sInput;

3) Тепер кількість факультетів збільшиться на 1:

iDepartmentsNum++;

4) Передаємо введене користувачем ім'я факультету у масив списку факультетів, у комірку, що відповідає останньому № факультету:

sDepartmentsNames[iDepartmentsNum] = sInput;

5) Виводимо сповіщення про успішно введений елемент:

cout << endl << "Новий факультет додано" << endl;

Опишемо дії програми за умови вибору оператором команди «4» («add»), якщо вже обрано рівень «ГРУПИ».

Так програма повинна буде додати нову групу на факультеті.

Вже зрозуміло, що додаватимемо команди у конструкцію:

case '4':

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 68

```

{
    ...
    else if(sLevel=="ГРУПИ")
    {
        тут будуть команди для роботи зі списком груп
    }
    ...
}

```

Додавання групи потребує вибору факультету.

Тому команди, які ми передбачимо у цьому випадку, будуть такі:

1) Запит на вибір факультету:

```

cout << "оберіть факультет > ";
cin >> sInput;

```

2) Присвоєння початкового значення обраному поточному номеру факультету.

```

iChosedDepartmentNum = 0;

```

3) Пошук введеного імені факультету у списку факультетів методом перебору і порівняння з усіма факультетами:

```

for (i = 1; i<=iDepartmentsNum; i++)
{
    if(sDepartmentsNames[i]==sInput)

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 69

```

    {
        iChooosedDepartmentNum = i;
    }
}

```

Знайдене значення № факультету у списку збережеться у змінній **iChooosedDepartmentNum**.

4) Якщо факультет у списку не знайдено, то змінна **iChooosedDepartmentNum** так лишиться нульовою, інакше

```

if (iChooosedDepartmentNum == 0)
{
    cout<<"Такого факультету не існує";
}
else
{
    тут будуть оператори додавання нової групи до факультету, № якого
    знайдено
}

```

5) У останню конструкцію

```

else
{
}

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 70

додамо такі оператори:

1) Запит на введення імені нової групи:

```
cout << "Введіть назву нової групи > ";
cin >> sInput;
```

2) Збільшення кількості груп у списку кількостей груп за факультетом. (Вже розуміємо, що № знайденого факультету буде у змінній

iChosedDepartmentNum, а кількості груп за факультетами у масиві **iGroupsNum[]**).

```
iGroupsNum[iChosedDepartmentNum]++;
```

3) Вносимо ім'я нової групи у масив імен груп за факультетами (№ факультету у змінній **iChosedDepartmentNum**, № нової групи = новій кількості груп на факультеті і знаходиться у масиві **iGroupsNum[iChosedDepartmentNum]**):

```
sGroupsNames[iChosedDepartmentNum][iGroupsNum[iChosedDepartmentNum]] = sInput;
```

4) Видаємо сповіщення про успішно додану групу:

```
cout << endl << "Нову групу додано" << endl;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 71

Аналогічні дії при виборі оператором команди «4» («add»), якщо вже обрано рівень «СТУДЕНТИ».

```
else if(sLevel=="СТУДЕНТИ")
```

```
{
```

```
    тут будуть оператори додавання нового студента
```

```
}
```

У середині цієї конструкції буде передбачуваний набір операторів:

```
cout << "оберіть групу > ";
```

```
cin >> sInput;
```

```
iChooosedGroupNum = 0;
```

```
for (i = 1; i<=iDepartmentsNum; i++)
```

```
    for (j = 1; j<=iGroupsNum[i]; j++)
```

```
    {
```

```
        if(sGroupsNames[i][j]==sInput)
```

```
        {
```

```
            iChooosedDepartmentNum = i;
```

```
            iChooosedGroupNum = j;
```

```
        }
```

```
    }
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 72

```

if (iChoosedGroupNum == 0) { cout<<"Такої групи не існує";}
else
{
    cout << "Введіть ім'я нового студента > ";
    cin >> sInput;
    iStudentsNum[iChoosedDepartmentNum][iChoosedGroupNum]++;

    sStudentsNames[ ][ ][ ]=sInput;
    cout << endl << "Нового студента додано" << endl;
}

```

Пусті номери елементів масиву імен студентів заповнимо потрібними номерами:

№ факультету - **iChoosedDepartmentNum**

№ групи - **iChoosedGroupNum**

№ студента - **iStudentsNum[iChoosedDepartmentNum][iChoosedGroupNum]**

Після конструкції:

```

else if(sLevel=="СТУДЕНТИ")
{

}

```

Розмістимо очікування реакції користувача на виконані дії.

```

system("pause");

```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 73

Дії при виборі оператором команди «3» («show»),
якщо вже обрано рівень «ФАКУЛЬТЕТИ».

Дії при виборі оператором команди «3» («show»),
якщо вже обрано рівень «ГРУПИ».

Дії при виборі оператором команди «3» («show»),
якщо вже обрано рівень «СТУДЕНТИ».

Вони абсолютно такі ж у плані необхідності шукати існування факультету та групи.

Проте, якщо вже знайдено факультет то використовується простий вивід у циклі (підказка, що ці оператори потрібно розмістити у конструкції «else{ }»):

Виконаємо це завдання у повній мірі на **наступній** лабораторній роботі.

Оператори виведення списку факультетів будуть включати такі елементи:

```
cout << endl;
```

```
for(i = 1; i<=iDepartmentsNum; i++)
{
    cout<<sDepartmentsNames[i]<<endl;
}
```

```
cout << endl;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 74

- оператори виведення списку груп:

```
cout << endl;
for(i = 1; i<=iGroupsNum[iChoosedDepartmentNum]; i++)
{
    cout<<sGroupsNames[iChoosedDepartmentNum][i]<<endl;
}
cout << endl;
```

- оператори виведення списку студентів:

```
cout << endl;
for(i = 1; i<=iStudentsNum[iChoosedDepartmentNum][iChoosedGroupNum]; i++)
{
    cout<<sStudentsNames[iChoosedDepartmentNum][iChoosedGroupNum][i]<<endl;
}
cout << endl;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 75

Лабораторна робота № 5 (продовження Л.р.№4)

Операції з масивами

Мета роботи:

Здобути навички використання операцій модуля підключення **<algorithm>** при створенні програм виконання операцій з масивами даних за вибором користувача і за допомогою меню користувача, у середовищі розробки Microsoft Visual C++.

Порядок виконання роботи.

Доповнити проект консольної програми Win32, яка виконує операції з масивами даних відповідно до вибору користувача і за допомогою меню користувача.

Скласти програму, яка реалізує додаткові операції роботи зі списком факультетів, списком груп для кожного з факультетів, списком студентів для кожної з груп відповідного факультету:

- е. В консолі користувач може викликати на виконання команди:
 - і. показати перелік елементів списку: **«show»**
 - ii. знайти елемент у списку: **«find»**
 - iii. відредагувати елемент зі списку: **«edit»**

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 76

Виконати інтерфейс спілкування програми з користувачем відповідно прикладу у файлі: «lr_05_приклад.exe».

Короткі відомості, щодо порядку виконання роботи:

Доповнення:

Виникала помилка при виведенні списку (команда - show), кількість елементів якого до цього не була визначена.

- при виведенні списку факультетів, коли не введено жодного факультету.
- при виведенні списку груп тільки-но створеного факультету.
- при виведенні списку студентів тільки-но створеної групи.

Причини помилок та їх усунення

1) При запуску не визначено початкова кількість факультетів.

- при оголошенні змінної «кількість факультетів» також виконаємо її ініціалізацію:

```
int iDepartmentsNum = 0;
```

2) При додаванні факультету, не було визначено кількість груп у новому факультеті.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 77

- після додавання факультету одразу заносимо дані про кількість груп у ньому:

case '4': // додати елемент рівня

if (sLevel=="ФАКУЛЬТЕТИ")

{

...

sDepartmentsNames[iDepartmentsNum] = sInput;

iGroupsNum[iDepartmentsNum]= 0;

...

3) При додаванні групи, не було визначено кількість студентів у новій групі.

case '4': // додати елемент рівня

{

...

else if(sLevel=="**ГРУПИ**")

{

...

sGroupsNames[iChosedDepartmentNum][iGroupsNum[iChosedDepartmentNum]] = sInput;

iStudentsNum[iChosedDepartmentNum][iGroupsNum[iChosedDepartmentNum]] = 0;

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 78

V Порядок створення програмного коду (продовження)

Дії при виборі користувачем команди «3» («show»), якщо вже обрано рівень «ФАКУЛЬТЕТИ».

Дії при виборі користувачем команди «3» («show»), якщо вже обрано рівень «ГРУПИ».

Дії при виборі користувачем команди «3» («show»), якщо вже обрано рівень «СТУДЕНТИ».

Пам'ятаємо, що у нас вже є заготовка:

```

case '3': {
    if (sLevel=="ФАКУЛЬТЕТИ") { // команди для роботи зі списком
    факультетів
        }
    else if(sLevel=="ГРУПИ") { // команд для роботи зі списком груп
        }
    else if(sLevel=="СТУДЕНТИ") { // команди для роботи зі списком
    студентів
        }
    }
    system("pause");
    break;
case '4':
...

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 79

Дії при виборі користувачем команди «3» («show»), якщо вже обрано рівень «ФАКУЛЬТЕТИ».

Отже, якщо поточний рівень факультети. То дописуємо зараз сюди (підправивши розміщення фігурних дужок):

```
case '3': {
    if (sLevel=="ФАКУЛЬТЕТИ")
    {
        тут необхідні команди для виведення списку факультетів
    }
    ...
}
```

За аналогією з попередніми пунктами включимо між { } код для виведення факультетів:

```
cout << endl;
for (i = 1; i<=iDepartmentsNum; i++)
    { cout<<sDepartmentsNames[i]<<endl;
    }
cout << endl;
```

Дії при виборі користувачем команди «3» («show»), якщо вже обрано рівень «ГРУПИ».

Виведення списку груп обіцяє бути дещо складнішим.

Дописувати код будемо у наступні { }:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 80

```

case '3': {
    if (sLevel=="ФАКУЛЬТЕТИ")
    {
        // вже є код
    }
    else if(sLevel=="ГРУПИ")
    {
        прописати команди для виведення списку груп
    }
    ...
}

```

Отже, програмно шукаємо номер факультету, групи якого необхідно вивести списком. Пошук буде за іменем факультету, що ввів користувач :

У **case '3'** після рядків

```
...
```

```
else if(sLevel=="ГРУПИ")
```

```
{
```

прописуємо:

```
// прописуємо команди
```

```
cout << "оберіть факультет > ";
```

```
cin >> sInput;
```

```
iChosedDepartmentNum = 0;
```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 81

```

for (i = 1; i<=iDepartmentsNum; i++)
    {
        if(sDepartmentsNames[i]==sInput)
            {
                iChosedDepartmentNum = i;
            }
    }

if (iChosedDepartmentNum == 0)
    { cout<<"Такого факультету не існує"<<endl<<endl;
    }
else
    {
        cout << endl;
        for(i = 1; i<=iGroupsNum[iChosedDepartmentNum]; i++)
            {

cout<<sGroupsNames[iChosedDepartmentNum][i]<<endl;

            }
        cout << endl;
    }

```

Дії при виборі користувачем команди «3» («show»), якщо вже обрано рівень «СТУДЕНТИ».

Оператори виведення списку студентів:

```

case '3': {
    ... // тут вже є оператори

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 82

```
else if (sLevel=="СТУДЕНТИ")
```

```
{
```

оператори для виведення студентів мають бути тут

```
}
```

Код для виведення списку студентів певної групи та факультету:

```
cout << "оберіть групу > ";
```

```
cin >> sInput;
```

```
iChosedGroupNum = 0;
```

```
for (i = 1; i<=iDepartmentsNum; i++)
```

```
    for (j = 1; j<=iGroupsNum[i]; j++)
```

```
    {
```

```
        if(sGroupsNames[i][j]==sInput)
```

```
        {
```

```
            iChosedDepartmentNum = i;
```

```
            iChosedGroupNum = j;
```

```
        }
```

```
    }
```

```
if (iChosedGroupNum == 0)
```

```
{
```

```
    cout<<"Такої групи не існує"<<endl<<endl;
```

```
}
```

```
else
```

```
{
```

```
    cout << endl;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 83

```

for(i = 1;
i<=iStudentsNum[iChooosedDepartmentNum][iChooosedGroupNum]; i++)
{

cout<<sStudentsNames[iChooosedDepartmentNum][iChooosedGroupNum][i]<<endl;

}
cout << endl;
}

```

Тепер :

Дії при виборі користувачем команди «5» («find»), якщо вже обрано рівень «ФАКУЛЬТЕТИ».

Дії при виборі користувачем команди «5» («find»), якщо вже обрано рівень «ГРУПИ».

Дії при виборі користувачем команди «5» («find»), якщо вже обрано рівень «СТУДЕНТИ».

Тепер окремо по кожній дії

Дії при виборі користувачем команди «5» («find»), якщо вже обрано рівень «ФАКУЛЬТЕТИ».

case '5': // знайти елемент рівня

if (sLevel=="ФАКУЛЬТЕТИ")

{

код для пошуку факультетів

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 84

}

...

Код пошуку факультетів повинен містити такі обов'язкові рядки:

```

cout << "введіть ім'я факультету для пошуку > ";
cin >> sInput;
iChooosedDepartmentNum = 0;
for (i = 1; i<=iDepartmentsNum; i++)
{
    if(sDepartmentsNames[i]==sInput)
    {
        iChooosedDepartmentNum = i;
    }
}

if (iChooosedDepartmentNum == 0)
{
    cout<<"Такого факультету не існує"<<endl<<endl;
}
else
{
    cout<<endl<<" Факультет
"<<sDepartmentsNames[iChooosedDepartmentNum];
    cout <<" знайдено"<< endl;

    int iStudentsOfDepartment = 0;
    for(q = 1; q<=iGroupsNum[iChooosedDepartmentNum]; q++)
    {

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 85

```

iStudentsOfDepartment+=iStudentsNum[iChooosedDepartmentNum][q];
}

```

```

cout<<" на факультеті: "<< endl;

```

```

cout<<" - "<<iGroupsNum[iChooosedDepartmentNum]<<"

```

```

груп"<<endl;

```

```

cout<<" - "<<iStudentsOfDepartment<<" студентів"<<endl;

```

```

cout << endl;

```

```

}

```

Пошук груп.

Дії при виборі користувачем команди «5» («find»), якщо вже обрано рівень «ГРУПИ».

```

case '5': // знайти елемент рівня

```

```

...

```

```

if (sLevel=="ГРУПИ")

```

```

{

```

```

// код пошуку груп вписати сюди

```

```

}

```

```

...

```

Ось цей код пошуку груп:

```

cout << " введіть ім'я групи для пошуку > ";

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 86

```

cin >> sInput;

iChoosedGroupNum = 0;

for (i = 1; i<=iDepartmentsNum; i++)
    for (j = 1; j<=iGroupsNum[i]; j++)
    {
        if(sGroupsNames[i][j]==sInput)
        {
            iChoosedDepartmentNum = i;
            iChoosedGroupNum = j;
        }
    }

if (iChoosedGroupNum == 0)
{
    cout<<"Такої групи не існує"<<endl<<endl;
}

else
{
    cout<<endl;

    cout<<" - знайдено таку групу на факультеті " <<
        sDepartmentsNames[iChoosedDepartmentNum] << endl;

    cout<<" у групі:
    "<<iStudentsNum[iChoosedDepartmentNum][iChoosedGroupNum] <<
    " студентів"<<endl;

    cout << endl;
}

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 87

Дії при виборі користувачем команди «5» («find»), якщо вже обрано рівень «СТУДЕНТИ».

case '5': // знайти елемент рівня

```

...
if (sLevel=="СТУДЕНТИ")
{
    // код пошуку студентів !! вписати сюди
}
...

```

І ось такий, абсолютно не складний код нижче:

```

cout << " введіть ім'я студента для пошуку > ";
cin >> sInput;
iChooosedStudentNum = 0;
for (i = 1; i<=iDepartmentsNum; i++)
    for (j = 1; j<=iGroupsNum[i]; j++)
        for (q = 1; q<=iStudentsNum[i][j]; q++)
            {
                if(sStudentsNames[i][j][q].find(sInput)!= string::npos)
                    {

                        iChooosedDepartmentNum = i;
                        iChooosedGroupNum = j;
                        iChooosedStudentNum = q;
                        cout<<endl;
                    }
            }

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 88

```

cout<<" - знайдено студента "<<sStudentsNames[i][j][q]
cout<<endl;

cout<<" на факультеті: "
cout<<sDepartmentsNames[iChooosedDepartmentNum]
cout << endl;
cout<<" у групі: ";
cout<<sGroupsNames[iChooosedDepartmentNum][iChooose
dGroupNum];
cout<<endl << endl;
}
}

if (iChooosedStudentNum == 0)
{
    cout<<"Такого студента не існує"<<endl<<endl;
}

```

Зверніть увагу, що тут використано функцію **.find(sInput)** алгоритмічної обробки даних масиву з підключеного модуля <algorithm>.

Значимо, що **string::npos** функція **.find** повертає, коли не знаходить у рядку відрізка, який вказаний змінною **sInput** у її параметрі **.find (sInput)**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 89

І у останню чергу розглянемо:

Дії при виборі користувачем команди «5» («edit»),
якщо вже обрано рівень «ФАКУЛЬТЕТИ».

Дії при виборі користувачем команди «5» («edit»),
якщо вже обрано рівень «ГРУПИ».

Дії при виборі користувачем команди «5» («edit»),
якщо вже обрано рівень «СТУДЕНТИ».

Подальші фрагменти кодів також по черзі будемо розміщувати між:

case '6':

{ // змінити елемент рівня

if (sLevel=="ФАКУЛЬТЕТИ")

{

команди для роботи зі списком факультетів

}

else if(sLevel=="ГРУПИ")

{

команд для роботи зі списком груп

}

else if(sLevel=="СТУДЕНТИ")

{

команди для роботи зі списком студентів

}

}

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 90

system("pause");

break;

Дії при виборі користувачем команди «5» («**edit**»), якщо вже обрано рівень «**ФАКУЛЬТЕТИ**».

(! структурувати код самостійно у редакторі MS VS)

```

cout << " Введіть діюче ім'я факультету > ";
cin >> sInput;
iChooosedDepartmentNum = 0;
for (i = 1; i<=iDepartmentsNum; i++)
{
if(sDepartmentsNames[i]==sInput)
{
iChooosedDepartmentNum = i;
}
}
if (iChooosedDepartmentNum == 0)
{
cout<<"Такого факультету не існує"<<endl<<endl;
}
else
{
cout<<endl<<" Ведіть нове ім'я факультету >";
cin >> sInput;
sDepartmentsNames[iChooosedDepartmentNum]=sInput;
cout << endl;
cout << "Зміни прийнято"<<endl;

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 91

```
cout << endl;
}
```

Дії при виборі користувачем команди «5» («edit»), якщо вже обрано рівень «ГРУПИ».

(структурувати код самостійно у редакторі MS VS)

```
cout << " Введіть діюче ім'я групи > ";
cin >> sInput;
iChooosedGroupNum = 0;
for (i = 1; i<=iDepartmentsNum; i++)
for (j = 1; j<=iGroupsNum[i]; j++)
{
if(sGroupsNames[i][j]==sInput)
{
iChooosedDepartmentNum = i;
iChooosedGroupNum = j;
}
}
if (iChooosedGroupNum == 0)
{
cout<<"Такої групи не існує"<<endl<<endl;
}
else
{
cout<<endl;
cout<<" Введіть нове ім'я групи > ";
cin >> sInput;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 92

```

sGroupsNames[iChosedDepartmentNum][iChosedGroupNum]=sInput;

cout << endl;

cout << " Зміни прийнято" << endl;

cout << endl;

}

```

Дії при виборі користувачем команди «5» («edit»), якщо вже обрано рівень «СТУДЕНТИ».

(! структурувати код самостійно у редакторі MS VS)

```

cout << " Введіть діюче ім'я групи > ";
cin >> sInput;
iChosedGroupNum = 0;
for (i = 1; i<=iDepartmentsNum; i++)
for (j = 1; j<=iGroupsNum[i]; j++)
for (q=1; q<=iStudentsNum[i][j]; q++)
{
if(sGroupsNames[i][j]==sInput)
{
iChosedDepartmentNum = i;
iChosedGroupNum = j;
}
}
if (iChosedGroupNum == 0)
{
cout<<"Такої групи не існує"<<endl<<endl;
}
else

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 93

```

{
cout<<" Введіть діюче ім'я студента > ";
cin >> sInput;
string * ptr_find;
string * ptr_begin =
&sStudentsNames[iChooosedDepartmentNum][iChooosedGroupNum][0];
string * ptr_end =
&sStudentsNames[iChooosedDepartmentNum][iChooosedGroupNum][0]+
iStudentsNum[iChooosedDepartmentNum][iChooosedGroupNum];
ptr_find = find(ptr_begin, ptr_end + 1 , sInput);
if (* ptr_find == sInput)
{
cout<<" Введіть нове ім'я студента > ";
cin >> sInput;
*ptr_find = sInput;
cout <<endl;
cout << " Зміни прийнято"<<endl;
cout << endl;
}
else
{
cout<<"Такого студента не існує"<<endl<<endl;
}
}

```

Уважний студент помітить у останньому фрагменті коду роботу з покажчиками та функціоналом підключеного модуля <algorithm>.

Далі буде.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 94

Лабораторна робота № 6 (продовження Л.р.№5)

Операції з масивами

Мета роботи:

Здобути навички використання командних інструкцій виведення даних операцій модуля підключення `<algorithm>` та `<fstream>` при створенні програм виконання операцій з масивами даних за вибором користувача і за допомогою меню користувача, у середовищі розробки Microsoft Visual C++.

Порядок виконання роботи.

Доповнити проект консольної програми Win32, яка виконує операції з масивами даних відповідно до вибору користувача і за допомогою меню користувача.

Скласти програму, яка реалізує додаткові операції зі списком факультетів, списком груп для кожного з факультетів, списком студентів для кожної з груп відповідного факультету:

- f. З консолі користувач викликає виконання таких додаткових команд:
 - i. видалити елемент зі списку: «**del**»
 - ii. зберегти списки у файл: «**save**»

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 95

- iii. прочитати списки з файлу: «**load**»
- iv. створити файл звіту: «**make report to file**»

Виконати інтерфейс спілкування програми з користувачем відповідно прикладу у файлі: «**lr_06_приклад.exe**».

Короткі відомості, щодо порядку виконання роботи:

V Порядок створення програмного коду (продовження)

Дії при виборі користувачем команди «7» («del»),
якщо вже обрано рівень «**ФАКУЛЬТЕТИ**»

Дії при виборі користувачем команди «7» («del»),
якщо вже обрано рівень «**ГРУПИ**»

Дії при виборі користувачем команди «7» («del»),
якщо вже обрано рівень «**СТУДЕНТИ**»

Зрозуміло використовуємо раніше прописані рядки коду:

Для зручності подальшого пошуку та налагодження виставимо одразу { }
вивірені по положенню відповідних операторів, а також **закоментуємо** початок
та кінець інструкцій «case '7'», if, else if:

////////////////////////////////кінєць case 6////////////////////////////////////

case '7':

if (sLevel=="ФАКУЛЬТЕТИ")

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 96

{

// тут розташуємо

// команди для видалення обраного факультету

// команди для видалення всіх груп факультету

// команди для видалення всіх студентів всіх груп факультету

}

////////////////кінєць case 7 видалення факультету////////////////////////////////////

else if(sLevel=="ГРУПИ")

{

// тут розташуємо

// команди для видалення обраної групи

// команди для видалення всіх студентів групи

}

////////////////кінєць case 7 видалення групи////////////////////////////////////

else if(sLevel=="СТУДЕНТИ")

{

// тут розташуємо

// команди для видалення обраного студента

}

////////////////кінєць case 7 видалення студента////////////////////////////////////

system("pause");

break;

////////////////кінєць case 7////////////////////////////////////

case '8': ...

Тепер вноситимемо по черзі блоки команд.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 97

Розпочнемо з найпростішого. Найпростіший перелік команд буде при видаленні одного обраного студента.

Дії при виборі користувачем команди «7» («del»), якщо вже обрано рівень «СТУДЕНТИ»

Команди вноситимемо у блок:

```
else if(sLevel=="СТУДЕНТИ")
```

```
{
```

```
    // тут розташуємо
```

```
    // команди для видалення обраного студента
```

```
}
```

```
////////////////////кінєць case 7 видалення студента////////////////////
```

Якщо не виникло бажання створити блок самостійно, можна скористатись командами, що наведені нижче:

```
    // команди для видалення обраного студента
```

```
// однакові студенти можуть бути в різних групах, тому оберемо групу студента
```

```
cout << " Введіть ім'я групи > ";
```

```
cin >> sInput;
```

```
// знайдемо групу за назвою (вона унікальна в університеті)
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 98

// переберемо по всім факультетам всі групи і зупинимось на співпадаючій

назві

```
iChooosedGroupNum = 0;
```

```
for (i = 1; i<=iDepartmentsNum; i++)
```

```
    for (j = 1; j<=iGroupsNum[i]; j++)
```

```
        for (q=1; q<=iStudentsNum[i][j]; q++)
```

```
        {
```

```
            if(sGroupsNames[i][j]==sInput)
```

```
                {    iChooosedDepartmentNum = i;
```

```
                    // фіксуємо номер факультету
```

```
                    iChooosedGroupNum = j;
```

```
                    // фіксуємо номер групи
```

```
                }
```

```
        }
```

```
if (iChooosedGroupNum == 0)
```

```
    {    cout<<endl<<"Такої групи не існує"<<endl<<endl;
```

```
    }
```

```
else
```

```
    {    // якщо група існує, то виберемо конкретного студента
```

```
        cout<<" Введіть ім'я студента для видалення > ";
```

```
        cin >> sInput;
```

```
// find() це засіб <algorithm>, він знайде покажчик на його ім'я у списку групи
```

```
    string * ptr_find;
```

```
    string * ptr_begin =
```

```
&sStudentsNames[iChooosedDepartmentNum][iChooosedGroupNum][0];
```

```
    string * ptr_end = ptr_begin + 50 - 1;
```

```
    ptr_find = find(ptr_begin, ptr_end + 1 , sInput);
```

```
    // якщо значення знайденого покажчика рівне введеному імені,
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 99

```

if (* ptr_find == sInput)
    {
        cout<<endl<<" студента знайдено у списку групи
" << endl;
// remove() це засіб <algorithm>, він видалить елемент масиву і змістить інші
remove(ptr_find, ptr_end+1, sInput);
// змінюємо кількість студентів на 1

iStudentsNum[iChooosedDepartmentNum][iChooosedGroupNum]--;
cout <<" та видалено " << endl<<endl;
    }
else
    {
        cout<<endl<<" Такого студента не існує" << endl<<endl;
    }
}

```

Щодо студента цього коду достатньо.

Для повного розуміння дамо відповідь на питання, чому при видаленні студента просто не присвоїти пустий символ " " елементу масиву імен студентів.

Візьмемося створити код для видалення групи.

Дії аналогічні попереднім. Проте, видаливши групу, потрібно буде видалити ще і всіх її студентів зі списків.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 100

Дії при виборі користувачем команди «7» («del»), якщо вже обрано рівень «ГРУПИ»

Працюватимемо з блоком:

```

else if(sLevel=="ГРУПИ")
{
    // тут розташуємо
    // команди для видалення обраної групи
    // команди для видалення всіх студентів групи
}
////////////////////////////////кінєць case 7 видалення групи////////////////////////////////

```

Не питаючи якого-небудь дозволу, можете скопіювати у блок код, зазначений нижче, звертаючи увагу на коментарі.

```

// команди для видалення обраної групи
// команди для видалення всіх студентів групи

```

```

// 1 в 1 код пошуку групи, який раніше вже використали

```

```

cout << " Введіть ім'я групи для видалення > ";
cin >> sInput;
iChooosedGroupNum = 0;
for (i = 1; i<=iDepartmentsNum; i++)
    for (j = 1; j<=iGroupsNum[i]; j++)

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 101

{

if(sGroupsNames[i][j]==sInput)

{

iChosedDepartmentNum = i;

iChosedGroupNum = j;

}

}

if (iChosedGroupNum == 0)

{ cout<<"Такої групи не існує"<<endl<<endl;

}

else

{ // якщо група існує, сповістимо користувача

cout<<endl;

cout<<" знайдено групи " <<

sGroupsNames[iChosedDepartmentNum][iChosedGroupNum];

cout <<" на факультеті

"<<sDepartmentsNames[iChosedDepartmentNum]<< endl;

// видаляємо ім'я групи зі списку груп

string * ptr_begin = &sGroupsNames[0][0];

string * ptr_end = ptr_begin + 50-1;

remove(ptr_begin, ptr_end + 1, sInput);

// мітимо для видалення число зі списку кількості студентів по групах

// мітка = 100 (унікальна, бо кількість студентів не більша 49)

iStudentsNum[iChosedDepartmentNum][iChosedGroupNum]=100;

// видаляємо мічене

int * ptr_begin_ = &iStudentsNum[0][0];

int * ptr_end_ = ptr_begin_ + 50-1;

remove(ptr_begin_, ptr_end_+1, 100);

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 102

```
// готуємо список студентів для видалення (тільки обраної групи)
```

```
// мітимо їх унікальною міткою "Видалити"
```

```
for (i=0; i<=49; i++)
```

```
{
```

```
sStudentsNames[iChooosedDepartmentNum][iChooosedGroupNum][i]="Видалити";
```

```
}
```

```
// видаляємо мічених студентів
```

```
ptr_begin = &sStudentsNames[0][0][0];
```

```
ptr_end = ptr_begin + 50*50 - 1;
```

```
remove(ptr_begin, ptr_end + 1, "Видалити");
```

```
// зменшуємо на 1 кількість груп
```

```
iGroupsNum[iChooosedDepartmentNum]--;
```

```
// сповіщення користувачеві
```

```
cout <<endl<<" обрану групу та всіх її студентів видалено"<<endl<< endl;
```

```
}
```

Цікаво було б задатися питанням, чому у вищенаведеному кодї не використано засіб find() для пошуку групи.

Тепер найскладніший етап: видалення факультету з його групами та студентами.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 103

Дії при виборі користувачем команди «7» («del»), якщо вже обрано рівень «ФАКУЛЬТЕТИ»

Зараз додаватимемо до блоку:

```

if (sLevel=="ФАКУЛЬТЕТИ")
{
    // тут розташуємо
    // команди для видалення обраного факультету
    // команди для видалення всіх груп факультету
    // команди для видалення всіх студентів всіх груп факультету
}
////////////////////////////////кінєць case 7 видалення факультету////////////////////////////////

```

У більшості схожий на попередні, але дещо ускладнений код приведено нижче:

```

// команди для видалення обраного факультету
// команди для видалення всіх груп факультету
// команди для видалення всіх студентів всіх груп факультету

// обираємо факультет
cout << " введіть ім'я факультету для видалення > ";
cin >> sInput;
iChooosedDepartmentNum = 0;
for (i = 1; i<=iDepartmentsNum; i++)
{

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 104

```

if(sDepartmentsNames[i]==sInput)
{
    iChooosedDepartmentNum = i;
}
}
if (iChooosedDepartmentNum == 0)
{
    cout<<endl<<" Такого факультету не існує"<<endl<<endl;
}
else
{
    cout<<endl<<" Факультет
"<<sDepartmentsNames[iChooosedDepartmentNum]<<" знайдено"<< endl;
    // видалятимемо ім'я факультету зі списку
    string * ptr_begin = &sDepartmentsNames[0];
    string * ptr_end = ptr_begin + 10-1;
    remove(ptr_begin, ptr_end + 1, sInput);
    // видалятимемо дані про кількість груп на факультеті
    int * ptr_begin_ = &iGroupsNum[0];
    int * ptr_end_ = ptr_begin_ + 10-1;
    iGroupsNum[iChooosedDepartmentNum]=100;
    // число 100 -мітка даних, що видалятимуться
    remove(ptr_begin_, ptr_end_+1, 100);
    // мітимо міткою "Видалити" і видалятимемо дані про імена груп на
факультеті
    for (i=0; i<=49; i++)
{sGroupsNames[iChooosedDepartmentNum][i]="Видалити";}
    ptr_begin = &sGroupsNames[0][0];
    ptr_end = ptr_begin + (10*50)-1;
}

```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 105

```

remove(ptr_begin, ptr_end + 1, "Видалити");
// мітимо міткою 100 і видалятимемо дані
//про кількість студентів по групах факультету
for(i=0; i<=49; i++){iStudentsNum[iChooosedDepartmentNum][i]=100;}
ptr_begin_ = &iStudentsNum[0][0];
ptr_end_ = ptr_begin_ + (10*50)-1;
remove(ptr_begin_, ptr_end_+1, 100);
// мітимо міткою "Видалити" і видалятимемо дані
//про імена студентів груп факультету
for (i=0; i<=49; i++) for (j=0; j<=49; j++)
{sStudentsNames[iChooosedDepartmentNum][i][j]="Видалити";}
ptr_begin = &sStudentsNames[0][0][0];
ptr_end = ptr_begin + (10*50*50) - 1;
remove(ptr_begin, ptr_end + 1,"Видалити");
// зменшуємо кількість факультетів на 1
iDepartmentsNum--;
// даємо сповіщення користувачу;
cout << endl<< " обраний факультет та всі його групи видалено"<<endl<<
endl;
}

```

Приступимо до команд роботи з файлами.

Вони, на щастя, не залежать від рівнів факультет, студент чи група.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 106

Дії при виборі користувачем команди «8» («save»)

Дії при виборі користувачем команди «9» («load»)

Дії при виборі користувачем команди «0» («make report to file»)

Для прописування програмного коду по кожній дії у нас повинна бути готова така структура у єдиному для всієї програми операторі switch(sInput):

- з урахуванням коментарів структура виглядатиме так:

```
case '-':
```

```
...
```

```
case '8': // запис бази
```

```
    // тут розташуємо
```

```
    // команди запису файлу-бази
```

```
        system("pause");
```

```
        break;
```

```
////////////////кінєць case 8 запис файлу////////////////////////////////
```

```
case '9': // читання бази
```

```
    // тут розташуємо
```

```
    // команди читання файлу-бази
```

```
        system("pause");
```

```
        break;
```

```
////////////////кінєць case 9 читати файл////////////////////////////////
```

```
case '0': // запис звіту
```

```
    // тут розташуємо
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 107

// команди запису файлу-звіту

system("pause");

break;

////////////////////кінєць case 0 звіт у файл////////////////////

default:

....

Не переймаючись муками творення програмного коду використаємо запропоновані нижче блоки команд.

Спочатку:

Дії при виборі користувачем команди «8» («save»)

Наступні команди просто зберігають всі дані, що у головних змінних та масивах у файл.

// команди запису файлу-бази

// вибір імені файлу користувачем (без розширення)

cout<<endl<<" Введіть ім'я файлу > ";

cin>>slInput;

slInput+=" .txt";

// відкриття файлу для запису

my_file.open(&slInput[0], ios::out|ios::trunc);

// запис кількості факультетів

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 108

```
my_file<<iDepartmentsNum<<' ';
```

```
// запис всіх імен факультетів з кількістю груп
```

```
for(i=1;i<=iDepartmentsNum;i++)
```

```
{
```

```
my_file<<sDepartmentsNames[i]<<' '<<iGroupsNum[i]<<' ';
```

```
}
```

```
// запис всіх імен груп з кількістю студентів
```

```
for(i=1;i<=iDepartmentsNum;i++)
```

```
for(j=1;j<=iGroupsNum[i];j++)
```

```
{
```

```
my_file<<sGroupsNames[i][j]<<' '<<iStudentsNum[i][j]<<' ';
```

```
}
```

```
// запис всіх імен студентів
```

```
for(i=1;i<=iDepartmentsNum;i++)
```

```
for(j=1;j<=iGroupsNum[i];j++)
```

```
for(q=1;q<=iStudentsNum[i][j];q++)
```

```
{
```

```
my_file<<sStudentsNames[i][j][q]<<' ';
```

```
}
```

```
// закриття файлу
```

```
my_file.close();
```

```
// сповіщення користувачу
```

```
cout<<endl<<endl<<" дані збережено у файл: "<<sInput<<endl<<endl;
```

Примітимо, що при записі кожного елементу даних у файл, наступним є запис символу пробілу

«<<' '». Це зроблено для розділення окремих елементів даних у файлі.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 109

У наступному ці дані можна буде послідовно автоматично прочитати у змінні операторами читання з файлу «>>».

Дії при виборі користувачем команди «9» («load»)

Команди цього блоку повинні абсолютно відтворювати команди блоку «save», щодо послідовності зчитування змінних.

Різниця лише у напрямку переміщення даних: не «my_file<<», а «my_file>>».

Також не потрібні символи-розділювачі: пробіли ' '.

// команди читання файлу-бази

```
cout<<endl<<" Введіть ім'я файлу > ";
cin>>sInput;
sInput+=" .txt";
my_file.open(&sInput[0], ios::in);
my_file>>iDepartmentsNum;
for(i=1;i<=iDepartmentsNum;i++)
    {my_file>>sDepartmentsNames[i]>>iGroupsNum[i];
    }
for(i=1;i<=iDepartmentsNum;i++)
    for(j=1;j<=iGroupsNum[i];j++)
        {my_file>>sGroupsNames[i][j]>>iStudentsNum[i][j];
        }
for(i=1;i<=iDepartmentsNum;i++)
    for(j=1;j<=iGroupsNum[i];j++)
        for(q=1;q<=iStudentsNum[i][j];q++)
            {my_file>>sStudentsNames[i][j][q];
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 110

}

```
my_file.close();
```

```
cout<<endl<<endl<<" дані зчитано з файлу: "<<slnput<<endl<<endl;
```

Дії при виборі користувачем команди «0» («make report to file»)

Тут команди формують текст, по черзі виводячи імена факультетів, груп та студентів у текстовий файл у зручному для читання користувачем вигляді, так, якби це відбувалось виведенням у консоль.

Після розлогих попередніх пояснень цей фрагмент формування текстового файлу не потребує пояснень.

// команди запису файлу-звіту

```
cout<<endl<<" Введіть ім'я файлу > ";
```

```
cin>>slnput;
```

```
slnput+=" .txt";
```

```
my_file.open(&slnput[0], ios::out | ios::trunc);
```

```
my_file<<" ЗВІТ ПРО СТВОРЕНУ БАЗУ ДАНИХ СТУДЕНТІВ
```

```
УНІВЕРСИТЕТУ"<<endl<<endl;
```

```
my_file<<" Підготував студент гр.АТ-21 ФІКТ ЖДТУ Добржанський
```

```
О.О."<<endl<<endl;
```

```
my_file<<"Всього факультетів: "<<iDepartmentsNum<<endl;
```

```
my_file<<"Назви факультетів: "<<endl;
```

```
for(i=1;i<=iDepartmentsNum;i++)
```

```
{my_file<<" -"<<slDepartmentsNames[i]<<": "<<iGroupsNum[i]<<"
```

```
груп"<<endl;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. _ / 111

}

```
for(i=1;i<=iDepartmentsNum;i++)
```

```
{ my_file<<endl<<endl<<" Факультет: "<<sDepartmentsNames[i]<<endl;
```

```
for(j=1;j<=iGroupsNum[i];j++)
```

```
{ my_file<<endl<<" Група: "<<sGroupsNames[i][j];
```

```
my_file<<endl<<" Студенти: ";
```

```
for(q=1;q<=iStudentsNum[i][j];q++)
```

```
{my_file<<endl<<" -
```

```
"<<sStudentsNames[i][j][q];
```

```
}
```

```
}
```

```
}
```

```
my_file<<endl<<endl<<endl<<" Кінець звіту.";
```

```
my_file.close();
```

```
cout<<endl<<endl<<" звіт збережено у файл: "<<sInput<<endl<<endl;
```

Модифікуючи послідовність запису змінних у файл, обираючи відступи та формуючи рядки, студент може створити власний дизайн текстового файлу звіту.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 112

Приклад створеного звіту:

<p>ЗВІТ ПРО СТВОРЕНУ БАЗУ ДАНИХ СТУДЕНТІВ УНІВЕРСИТЕТУ</p> <p>Підготував студент гр.АТ-21 ФІКТ ЖДТУ Добржанський О.О.</p> <p>Всього факультетів: 3</p> <p>Назви факультетів:</p> <ul style="list-style-type: none"> -ФІКТ: 4 груп -ФІМ: 3 груп -ФЕМ: 3 груп <p>Факультет: ФІКТ</p> <p>Група: АТ-26</p> <p>Студенти:</p> <ul style="list-style-type: none"> - Тимченко - Савченко - Біденко - Вакуленко <p>Група: АТК-17</p> <p>Студенти:</p> <ul style="list-style-type: none"> - Калиновський - Садовий - Корчмар - Васюнін <p>Група: АТ-21-1м</p>

Студенти:

- Положинський
- Кирилів
- Кирейцев
- Порошенко

Група: АТ-22

Студенти:

- Лозинський
- Шарварчук
- Левченко
- Луценко

Факультет: ФІМ

Група: ХВ-2

Студенти:

- Канів
- Кучерук
- Королевська

Група: ММ-3

Студенти:

- Ганін
- Варейчук
- Львовчкін
- Тимошенко

Група: АГ-4

Студенти:

- Кучма

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 114

Факультет: ФЕМ

Група: ЕО-11

Студенти:

- Кирейцев
- Журахівський

Група: ЕМ-22

Студенти:

- Панченко
- Ключев

Група: ЕП-23

Студенти:

- Корчемлюк
- Тимченко

Кінець звіту.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 115

Приклад вмісту файлу бази даних:

3 ФІКТ 4 ФІМ 3 ФЕМ 3 АТ-26 4 АТК-17 4 АТ-21-1м 4 АТ-22 4 ХВ-2 3 ММ-3 4
АГ-4 1 ЕО-11 2 ЕМ-22 2 ЕП-23 2 Тимченко Савченко Біденко Вакуленко
Калиновський Садовий Корчмар Васюнін Положинський Кирилів
Кирейцев Порошенко Лозинський Шарварчук Левченко Луценко Канів
Кучерук Королевська Ганін Варейчук Львовчкін Тимошенко Кучма
Кирейцев Журахівський Панченко Ключев Корчемлюк Тимченко

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 116

Лабораторна робота № 7

Сучасні інженерні та математичні пакети для створення моделей приладів і систем вимірювань: Моделювання функціонування вимірювальних систем на базі мікроконтролерів

Мета роботи:

Ознайомитись з середовищем моделювання мікроконтролерів.

Короткі відомості, щодо порядку виконання роботи:

Відповідно до завдання на сьогоднішнє заняття, ми повинні вивчити такий функціонал мікроконтролера:

- завантаження коду у мікроконтролер
- редагування коду мікроконтролера
- забезпечувало можливість перегляд змін та помилок коду мікроконтролера
- виконання коду мікроконтролера покроково

Розпочнемо зі звичайного

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 117

1.1 Загальні відомості

TinkerCAD – це online-сервіс і середовище моделювання для роботи з 3D об'єктами і електронними схемами, що належить в даний час компанії Autodesk. Відмінними рисами є відкритість, безкоштовний доступ, багаті функціональні можливості редактора. Підтримується групова робота, обмін готовими результатами, інтеграція з популярними каталогами 3D-моделей і системами віддаленої 3D-друку.

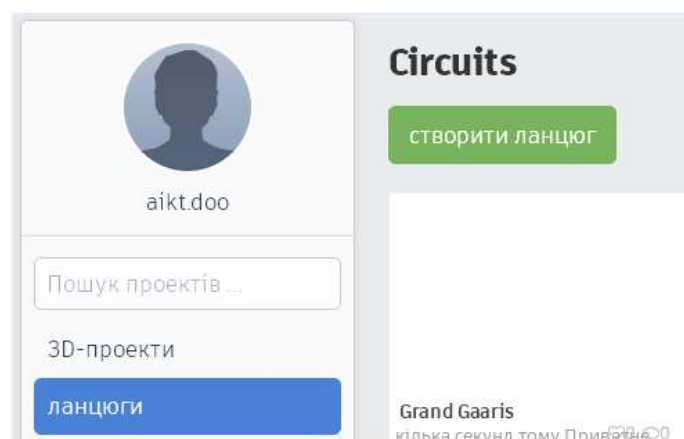
Для ознайомлення з мікроконтролерами використовуємо он-лайн віртуальну лабораторію **Tinkercad від Autodesk**.

Tinkercad дозволяє моделювати схеми на основі мікроконтролера Arduino.

Arduino непоганий контролер для навчання, оскільки підходи в багатьох мікроконтролерах схожі.

Реєструємося на порталі: <https://www.tinkercad.com/>

У особистому кабінеті обираємо моделювання мікросхем: Circuits та активуємо кнопку **Створити ланцюг**



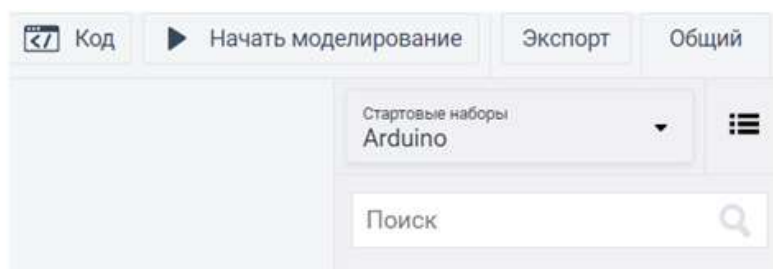
1.2 Робота з компонентами

Далі, ми можемо перетягувати на робоче поле Компоненти. Перелік компонент з правої частини вікна.

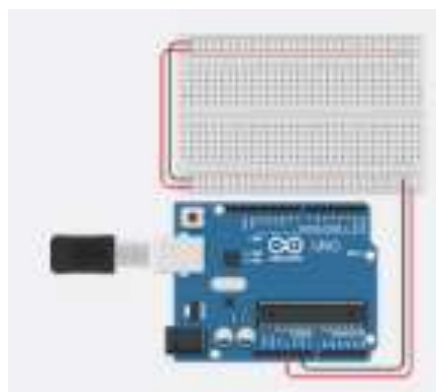
Клікнувши на окремих компонентах, можливо виділити їх для подальшого перетягування мишкою або видалення клавішею клавіатури delete.

Затиснувши Shift, утримуючи праву кнопку мишки, протягуючи вказівником мишки по робочі області, можливо виділяти сукупності компонентів.

При виборі компонентів можливо залишити **тип Базові**, але оберемо **Arduino**.



Почнемо з найпростішого, як для тих хто arduino бачить уперше. Візьмемо пустий arduino: тобто заготовку **Макетна плата**



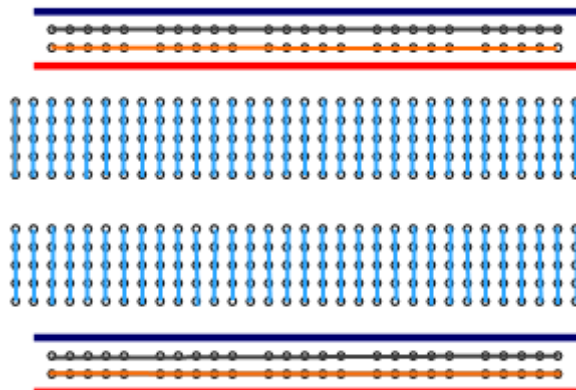
Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 119

Наведемо мишку на виходи arduino, до яких підключено проводи, та покрутимо коліщатко мишки, щоб наблизити їх.

Тут просто виконано розведення магістралі напруги живлення (від блока живлення плати arduino): GND = 0V та 5V на макетну плату.

Макетна плата призначена для зручного розміщення компонентів. Макетна плата дозволяє фізичне вставляння в роз'єми контактів компонентів чи інших невеликих приладів, для підключення до них напруги живлення, для з'єднання між собою та з arduino.

Якщо заглянути в середину макетної плати, ми побачимо яким чином електрично поєднані контакти:



Можна потренуватися підключати провідники, хоча б зараз до макетної плати. Коли при наведенні на певний вивід або роз'єм з'являється маленький

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 120

квадратик, то клікнувши на ньому можна далі витягувати провід до будь-якого іншого виводу або роз'єму, на якому для закінчення підключення потрібно також клікнути.

Потім, виділивши провід, можливо тягати за опорні точки для зміни його форми.

Макетна плата - це зручно, але не про неї зараз мова. Спробуємо поки обійтися без неї.

Для подальшого вивчення роботи самого arduino видалимо все додане та витягнемо на робоче поле заготовку Мерехтіння.



1.3 Запуск моделювання

Можна одразу ж **запустити проект кнопкою** меню  Почати моделювання.

Під час виконання програми в arduino, її можна перезапустити, натиснувши на платці arduino на кнопку reset:

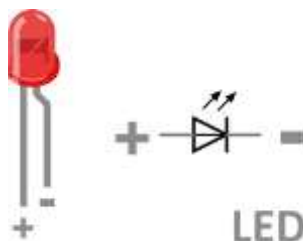
Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 121



Після отримання достатньої дози задоволення від блимання світло діода, можна зупинити проект кнопкою **Зупинити моделювання**.

Наведемо мишку на виходи arduino, до яких підключено світлодіод, та покрутимо коліщатко мишки, щоб наблизити їх.

Бачимо, що світлодіод підключено до дискретного виходу 13 та другий вихід до землі (зігнута ніжка означає + діода).



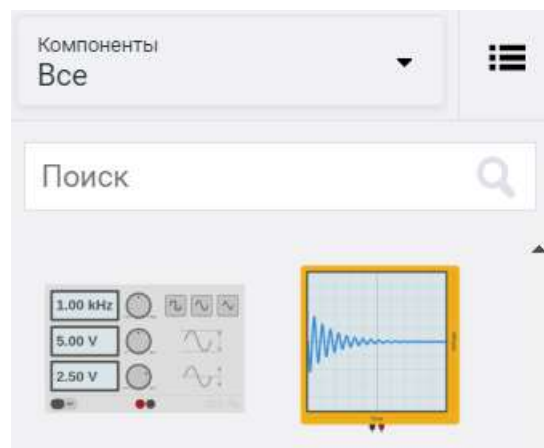
Коли програма в arduino включає дискретний вихід 13 у активний стан, arduino подає на вихід 13 напругу 5V. Якщо подати високий рівень напруги на + вхід діода, а – під'єднати на низький рівень напруги (GND), то діод відкриється і через нього потече струм. Якщо діод є LED, то він ще й засвітиться.

Але ж діод пульсує. Це означає імовірно певні зміни напруги на 13 виході arduino.

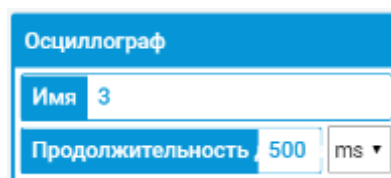
Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 122

Прямо зараз здійснимо майже технологічне вимірювання величини цієї напруги та її форми. Візьмемо віртуальний осцилограф.

Оберемо компоненти Всі та з них жовтенький осцилограф:



Витягнувши осцилограф, зробимо його налагодження: 500мс у 1 часовій поділці.




Тепер правильно приєднаємо виводи осцилографа до + та – діода до високого та низького рівня напруги, що на нього подається.

Запустимо, трохи зачекаємо, та отримаємо форму напруги на діоді. У режимі моделювання, на осцилографі відображаються шкали. За шкалами можливо визначити амплітуду та тривалість імпульсів.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 123

1.4 Програмування

Подивимося нарешті у нутроці arduino з позицій програміста. Тиснемо сміливо на клавішу  Код .

Якщо не висвітився текст програми, то змінюємо настройку Блоки на Текст:



Бачимо стандартний код arduino.

Дві такі собі чимось заповнені функції. Явно с++ або с походження.

Для того, щоб зрозуміти написане, обговоримо елементарний синтаксис кодів для arduino.

Звичайний багаторядковий коментар:

```
/*
```

```
Іфвафіа
```

```
фівафваі
```

```
*/
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 124

Звичайний однорядковий коментар:

// фівафіва

Функція **setup()**, яка виконується лише раз, на початку запуску програми arduino.

Її назва підказує, що у ній краще розташовувати початкові налагодження: ініціалізацію змінних, активацію портів, налагодження виводів arduino.

void setup()

```
{  
  
  
  
  
  
  
}
```

Функція **loop()**, яка буде виконуватись одразу після функції, **setup()**, і буде продовжувати повторне (циклічне) виконання до зупинки самого arduino.

void loop() // !! це автоматичний нескінченний цикл

```
{  
  
  
  
  
  
  
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 125

Налагодження виводу з arduino, для його використання як виходу (входу), тобто такий вивід можна буде вважати дискретним виходом (входом).

pinMode(3, OUTPUT);

pinMode(3, INPUT);

Як видно на платі 13 виводів у arduino. Кожен з них можна налагодити як вихід для видачі з arduino сигналів на інші пристрої, так і як вхід для подачі сигналів на arduino з датчиків та інших пристроїв.

Як дискретні входи можливо налагодити також аналогові входи arduino:

pinMode(A3, INPUT);

Коли налагоджені виходи, на них можливо надсилати високі та низькі рівні напруг (тобто включати/виключати):

digitalWrite(13, HIGH);

digitalWrite(13, LOW);

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 126

Для збереження певних значень можливо оголосити змінні. В arduino змінні оголошуються на початковому рівні часто глобальними (і робиться це перед **void setup()**):

```
int dd = 0;
```

```
float gg = 0.0;
```

```
void setup()
```

```
{
```

```
}
```

```
void loop()
```

```
{
```

```
}
```

Коли налагоджені входи, з них можливо у змінні зчитувати 1 або 0 при отриманні arduino високого чи низького рівнів сигналу на вході:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 127

```
dd = digitalRead(0);
```

```
dd = analogRead(A0);
```

Команда затримки виконання

```
delay(1000); // тут 1000 будь-яка кількість мілісекунд
```

Команди перевірки умови такі ж як і в c++:

```
if (dd == 1) {
```

```
    // певні дії
```

```
}
```

або

```
if (dd == 0) {
```

```
    // певні дії
```

```
} else {
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 128

// певні інші дії

}

Наприклад, можна перевірити чи натиснута кнопка, підключена до 6 входу arduino:

```
dd = digitalRead(6);
```

```
if (dd == HIGH) {
```

```
    // певні дії
```

```
}
```

або

```
dd = digitalRead(6);
```

```
if (dd == 1) {
```

```
    // певні дії
```

```
}
```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 129

Лабораторна робота № 8 (продовження л.р. 7)

Сучасні інженерні та математичні пакети для створення моделей приладів і систем вимірювань: Моделювання функціонування вимірювальних систем на базі мікроконтролерів

Мета роботи:

Отримати навички програмування мікроконтролерів.

Короткі відомості, щодо порядку виконання роботи:

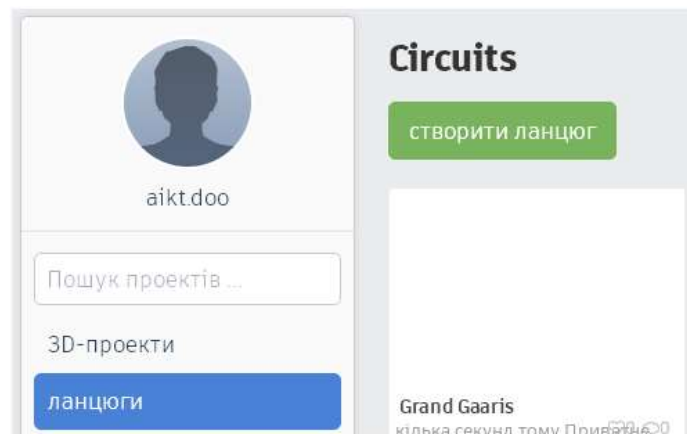
Відповідно до завдання на сьогоднішнє заняття, ми повинні вивчити такий функціонал мікроконтролера:

- створення коду для мікроконтролера
- виконання схемотехніки приєднання зовнішніх пристроїв до мікроконтролера

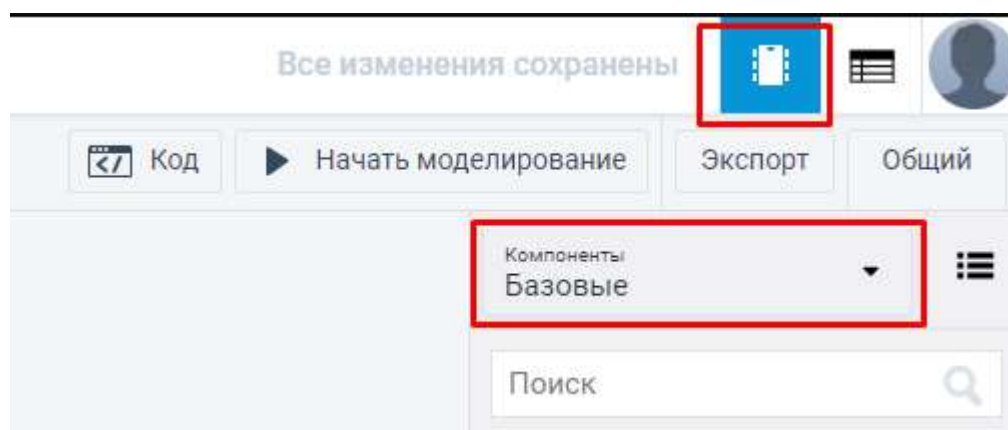
Розпочнемо

1.1 Порядок виконання роботи

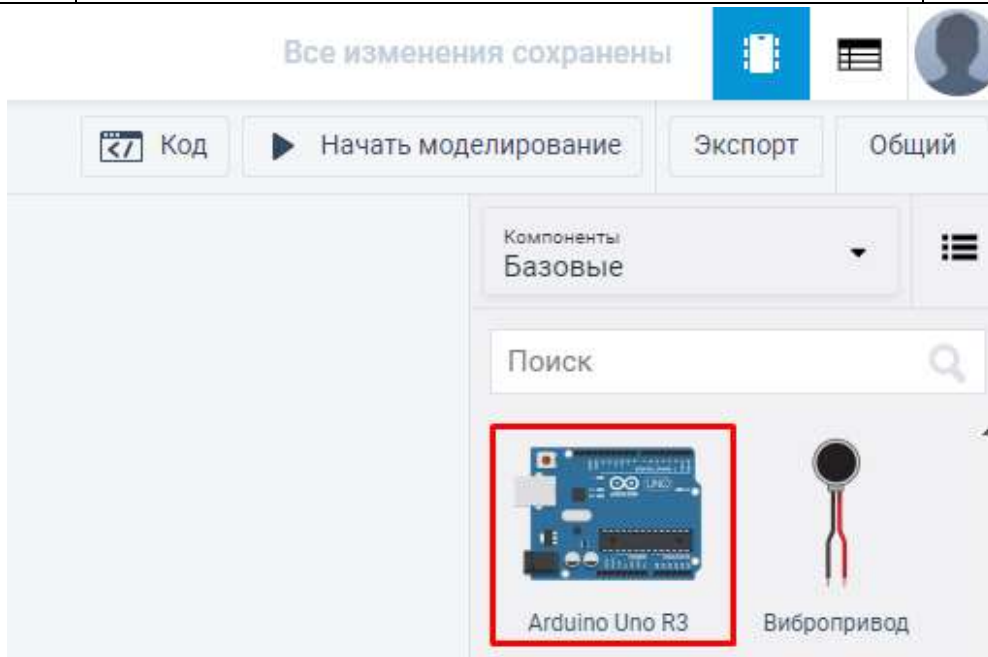
1. Створіть новий проект circuits у Tinkercad



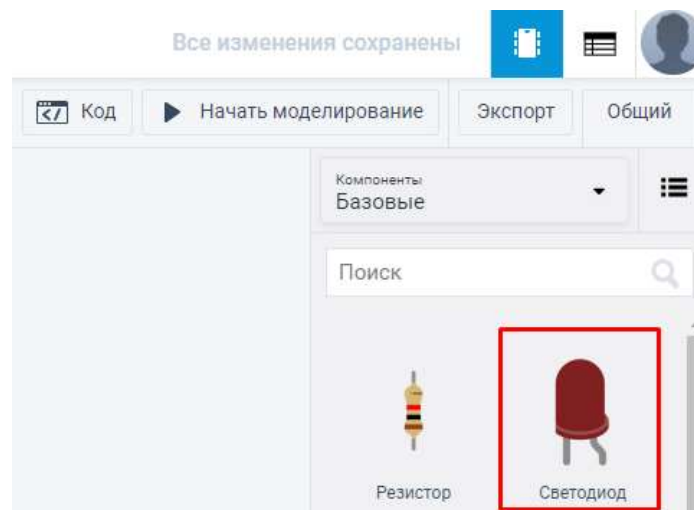
2. Переходимо в базові компоненти.



3. Обирати плату Arduino UNO. Щоб додати плату на робочий простір потрібно натиснути її та перетягнути з панелі компонентів на вільне місце.



4. Обрати LED, або інша його назва світлодіод. Додати до робочого простору.

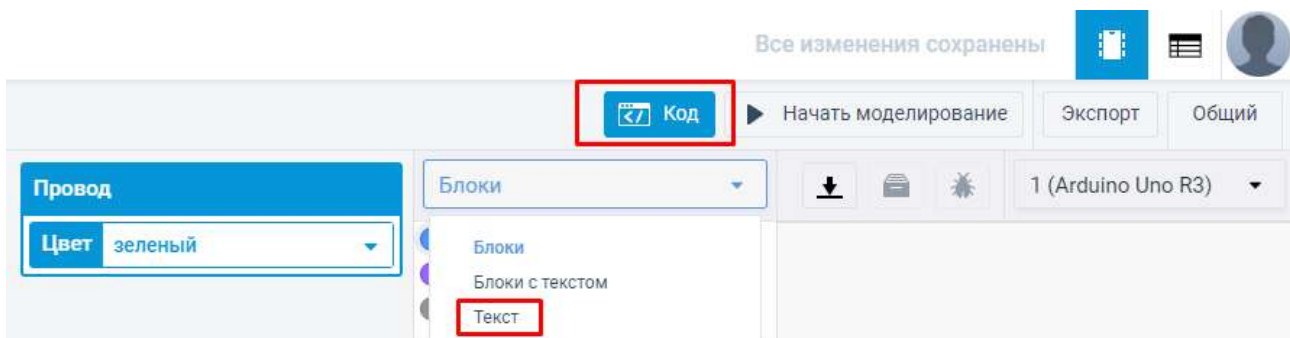


5. З'єднати світлодіод та плату Arduino UNO. Катодом до входу GND, а анодом до 13 входу.

(Підказка: Струм через діод протікатиме, коли на його + вивід (анод) подається вищий рівень напруги, а на – вивід (катод) подається нижчий рівень напруги)



6. Відкрити вікно програмування, обрати тип відображення текст.



7. Написати коду управляючої програми для світлодіода (у випадку, якщо код приклад не активувався автоматично).

Все изменения сохранены

Код ▶ Начать моделирование Экспорт Общій

Текст 1 (Arduino Uno R3)

```
1 void setup()
2 {
3   pinMode(13, OUTPUT);
4 }
5
6 void loop()
7 {
8   digitalWrite(13, HIGH);
9   delay(1000); // Wait for 1000 millisecond(s)
10  digitalWrite(13, LOW);
11  delay(1000); // Wait for 1000 millisecond(s)
12 }
```

8. Виконати моделювання коду.

Все изменения сохранены

Код ▶ Начать моделирование Экспорт Общій

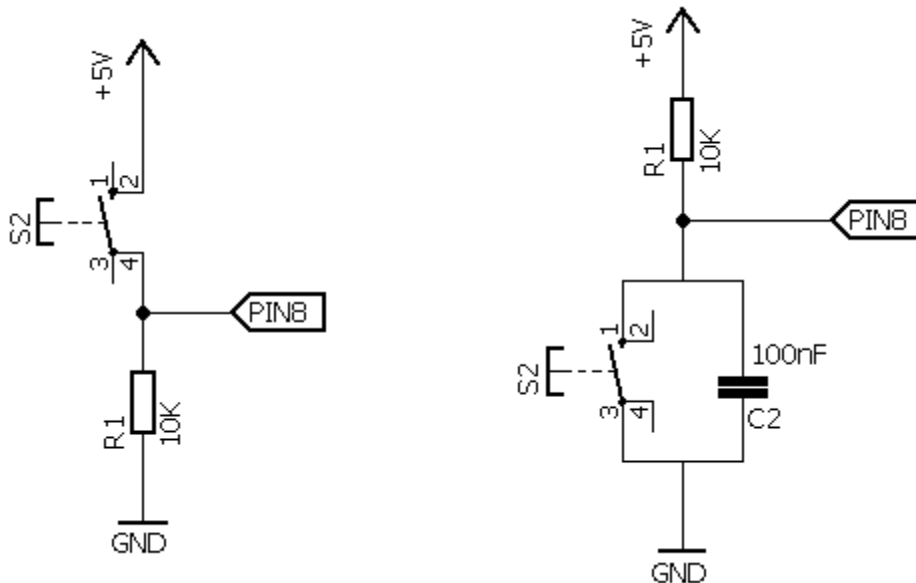
Текст 1 (Arduino Uno R3)

Скачать код

```
1 void setup()
2 {
3   pinMode(13, OUTPUT);
4 }
5
6 void loop()
7 {
8   digitalWrite(13, HIGH);
9   delay(1000); // Wait for 1000 millisecond(s)
10  digitalWrite(13, LOW);
11  delay(1000); // Wait for 1000 millisecond(s)
12 }
```

Додатково, підключити світлодіод через кнопку.

Кнопка має бути підключена за одним з варіантів до восьмого входу arduino:



Контрольне питання: Поясніть необхідність резистора підтягування до GND або підтягування до 5V з паралельно підключеним конденсатором.

У досліді з пульсуючим світлодіодом, як потрібно здійснити підключення, та яка має бути програма контролера, щоб світлодіод включався при періодичному виключенні (тобто подачі програмою періодичного низького рівня на дискретний вихід 13 контролера)? Продемонструйте у звіті схему та програму. Подайте необхідні пояснення.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 135

Лабораторна робота № 9 (продовження л.р. 8)

Сучасні інженерні та математичні пакети для створення моделей приладів і систем вимірювань: Моделювання функціонування вимірювальних систем на базі мікроконтролерів

Мета роботи:

Отримати навички програмування мікроконтролерів для виконання функцій вимірювання параметрів оточуючого середовища.

Короткі відомості, щодо порядку виконання роботи:

Відповідно до завдання на сьогоднішнє заняття, ми повинні вивчити такий функціонал мікроконтролера:

- створення коду для мікроконтролера
- виконання схемотехніки приєднання зовнішніх пристроїв до мікроконтролера (Підключення датчиків з аналоговим виходом до пристроїв програмованої логіки)

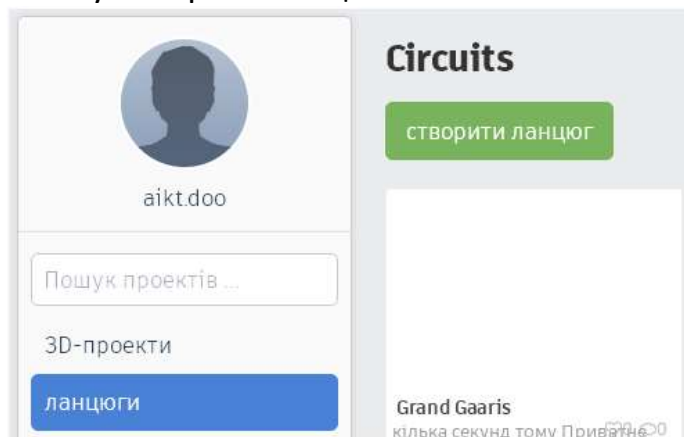
Розпочнемо

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 136

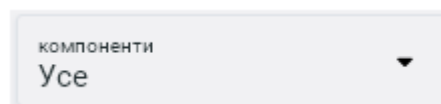
Виконання роботи:

1.1. Налагоджуємо систему моделювання (Tinkercad) та готуємо схему дослідження

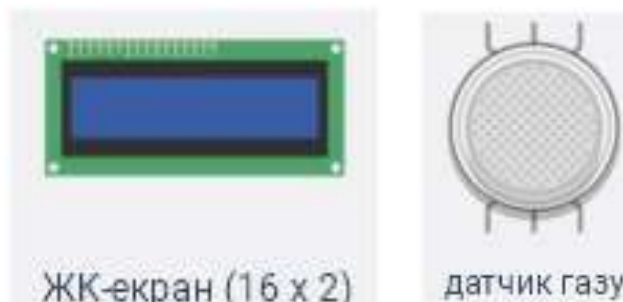
- Переходимо на порталі: <https://www.tinkercad.com/>
- У особистому кабінеті обираємо моделювання мікросхем: Circuits (Схеми) та активуємо кнопку Створити ланцюг

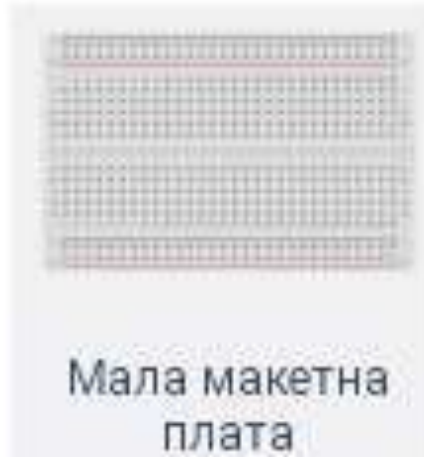


- Обираємо справа

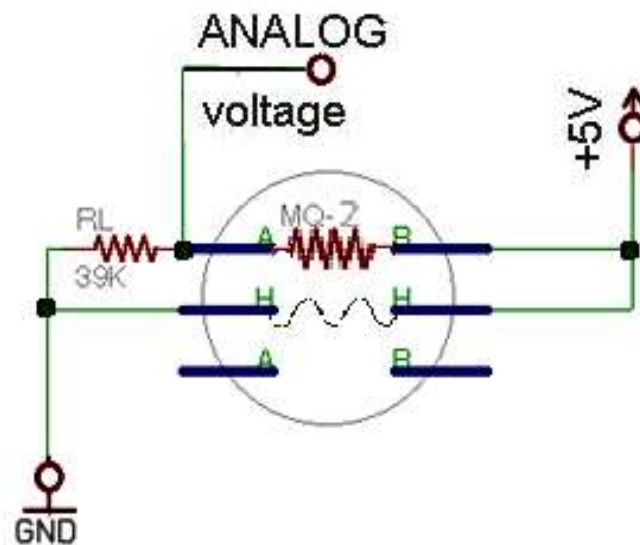


- Перетягуємо робоче поле компоненти:





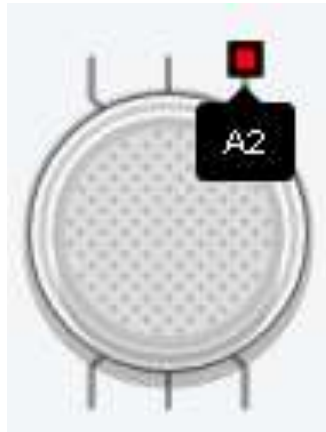
- Реалізуємо схему підключення датчика газу до Arduino




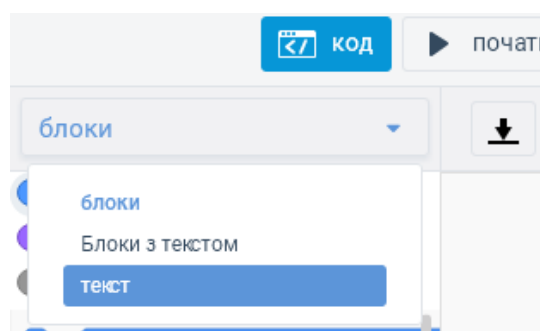
- навівши курсор на виводи датчика, можна побачити А, Н, В позначення, які відповідають схемі підключення (покрутивши

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 138

коліщатко миші можна збільшувати і зменшувати зображення на робочій області)



- резистор ззовні датчика ~~⚡~~ в схемі потрібен на 39 kΩ (клікнувши на ньому можливо викликати вікно для зміни його параметрів) інший в схемі позначений в середині датчика – його вставляти не потрібно
 - 5V взяти з виходу 5V Arduino
 - GND взяти з виходу GND Arduino
 - Вихід ANALOG voltage датчика направити на вхід A1 Arduino, або аналоговий вхід Arduino з іншим номером
 - інші виводи датчика приєднуємо відповідно до схеми підключення
-
- кінцеву схему відображаємо у звіті
-
- Увійдемо у систему програмування Arduino
 - тиснемо сміливо на позицію  Код
 - далі обираємо тип програми: «текст»



Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 139

- очищуємо програму до вигляду

```
void setup()
```

```
{
```

```
}
```

```
void loop()
```

```
{
```

```
}
```



- просто згадаємо:
 - у функцію `void setup()` пишемо лише команди, що потрібні для виконання лише одноразово і лише на початку програми
 - у функцію `void loop()` пишемо команди, для яких потрібен повтор виконання постійно, доки програма працює

1.2. Програмуємо мікроконтролер

- Виконуємо базовий код:

- на самому початку програми (перед `void setup()`) передбачаємо глобальну змінну з іменем «value» типу «int», для збереження потім у неї значення, зчитаного з датчика


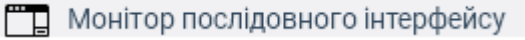
Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 140

- в програмі передбачаємо повторювані команди (обрати потрібні)
 - функція, що повертає значення сигналу з дискретного входу
digitalRead(номер задіяного дискретного входу A1, A2 тощо)
 - функція, що повертає значення сигналу з аналогового входу
analogRead(номер задіяного аналогового входу A1, A2 тощо)
 - класична команда умовного вибору
if (dd == 0) { // певні дії } else { // певні інші дії }
 - команда присвоєння значення змінній
value =
 - крапка з комою наприкінці кожної команди ;
- запускаємо проект, клікнувши на  ,
перевіряємо чи все працює без помилок
- зупиняємо проект для доопрацювання, клікнувши на 

- Знову звертаємося до коду Arduino, потрібен спосіб вивести для перегляду значення, виміряного на виході датчика газу

- однократно активуємо послідовний порт для виведення виміряних значень (обрати потрібну команду або команди та помістити у правильне місце програми)
 - pinMode(1, OUTPUT);
 - Serial.println(змінна або значення);
 - Serial Monitor(9600);
 - Serial.begin(9600);

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 141


- /*Serial Monitor*/
 - delay(100);
- застосовуємо повторювану команду виведення даних у послідовний порт (обрати потрібну команду або команди та помістити у правильне місце програми)
- pinMode(1, OUTPUT);
 - Serial.println(змінна або значення);
 - Serial Monitor(9600);
 - Serial.begin(9600);
 - /*Serial Monitor*/
 - delay(100);
- запускаємо проект, клікнувши на ,
- у режимі виконання відкриваємо вікно читання послідовного порту, клікнувши у вікні коду на позицію 
- занотовуємо (скріншотимо) побачене у звіт
- клікаємо на датчику газу і перетягуємо по екрану емуляцію газової суміші, спостерігаємо, чи змінюються виміряні значення у моніторі послідовного порту

1.3. Виконуємо перші вимірювання:

- Заповнюємо таблицю дослідних даних і розміщуємо її у звіті

Відстань від датчика до центру газової суміші	Виміряне мікроконтролером значення на виході датчика
---	--

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 142
5 діаметрів датчика		
4,5 діаметри датчика		
4 діаметри датчика		
3,5 діаметрів датчика		
3 діаметри датчика		
2,5 діаметри датчика		
2 діаметри датчика		
1,5 діаметри датчика		
1 діаметр датчика		
0,5 діаметра датчика		
0 діаметр датчика (центри датчика та газової суміші співпадають)		

- Зупиняємо проект для доопрацювання, клікнувши на 
- Дати відповідь на питання, чи справді виміряні мікроконтролером значення відповідають реальній фізичній концентрації газів у повітрі. Якщо ні, то що тоді виміряв мікроконтролер.

1.4. Доопрацьовуємо проект для корекції вимірянних значень

- Усвідомлюючи необхідність нормування сигналу для отримання реальних вимірянних значень, «калібруємо» створену систему вимірювання
 - Припустимо, маємо дані лабораторних вимірювань, що концентрація газів у суміші складає 10000 ppm

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 143

- При відсутності газової суміші над датчиком має бути виміряно 0 ppm (а виміряно
- При знаходженні датчика прямо у газовій суміші має бути виміряно 10000 ppm (а виміряно
- Для корекції виміряного значення застосуємо магічну функцію

змінна результат = $\max(X, \text{old_MIN}, \text{old_MAX}, \text{new_MIN}, \text{new_MAX})$;

де

X – це змінна, значення якої потрібно скоректувати

old_MIN – це мінімум сигналу, що існує

old_MAX – це максимум сигналу, що існує

new_MIN – це мінімальне значення, яке потрібне

new_MAX – це максимальне значення, яке потрібне

- Сформувати команду з \max та помістити у правильне місце програми (занотувати це у звіті)
 - Перевірити, чи потрібна корекція рядку, що виводить дані у послідовний порт (занотувати за потреби у звіті)
- Запустити і перевірити правильність роботи скоректованої програми

- Заповнюємо таблицю нових дослідних даних і розміщуємо її у звіті

Відстань від датчика до центру газової суміші	Виміряне мікроконтролером значення на виході датчика
5 діаметрів датчика	
4,5 діаметри датчика	
4 діаметри датчика	
3,5 діаметрів датчика	
3 діаметри датчика	
2,5 діаметри датчика	
2 діаметри датчика	
1,5 діаметри датчика	
1 діаметр датчика	
0,5 діаметра датчика	
0 діаметр датчика (центри датчика та газової суміші співпадають)	

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 145

Лабораторна робота № 10 (продовження л.р. 9)

Сучасні інженерні та математичні пакети для створення моделей приладів і систем вимірювань: Моделювання функціонування вимірювальних систем на базі мікроконтролерів

Мета роботи:

Отримати навички програмування мікроконтролерів для виконання функцій вимірювання параметрів оточуючого середовища.

Короткі відомості, щодо порядку виконання роботи:

Відповідно до завдання на сьогоднішнє заняття, ми повинні вивчити такий функціонал мікроконтролера:

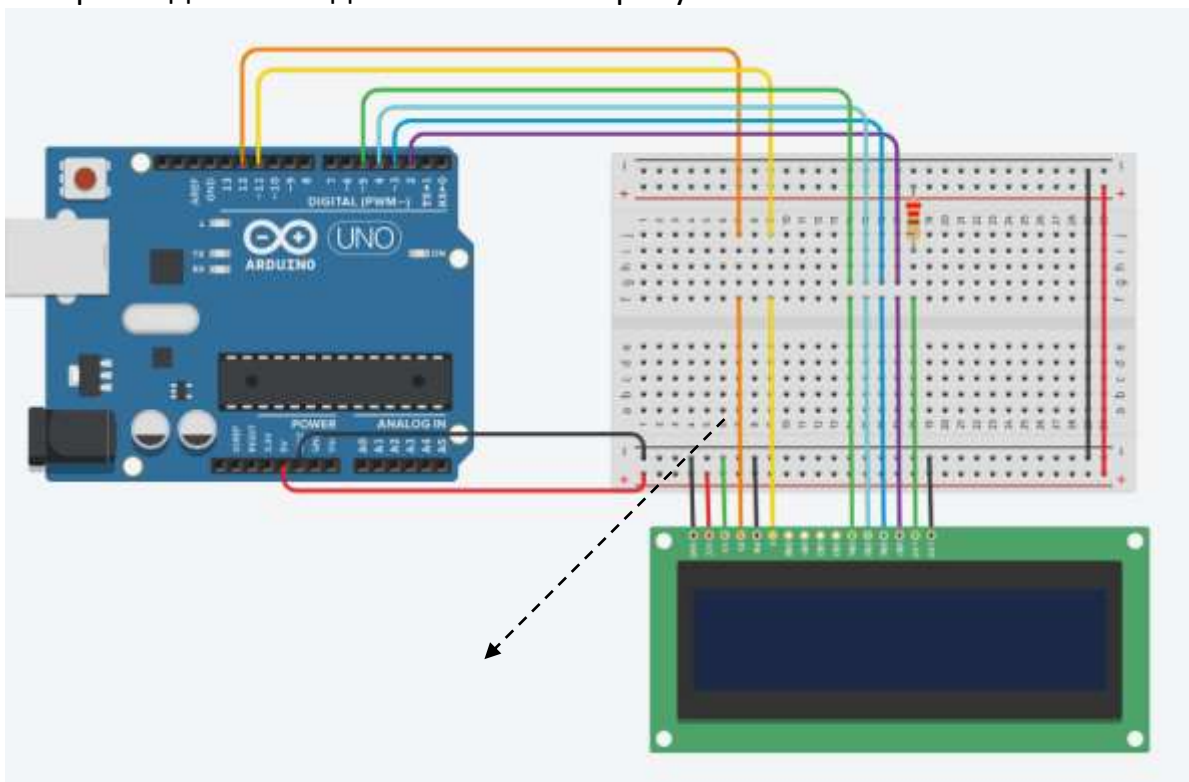
- створення коду для мікроконтролера
- виконання схемотехніки приєднання зовнішніх пристроїв до мікроконтролера (Підключення датчиків з аналоговим виходом до пристроїв програмованої логіки)

Розпочнемо

Виконання роботи:

1.1. Доопрацьовуємо проект додаванням індикаторів відображення інформації

- Використовуючи наведений нижче приклад виведення на ЖК-екран певного значення, запровадити у своїй схемі потрібні апаратні та програмні засоби для відображення на ЖК-екран виміряного значення та одиниць вимірювання. Результати роботи доопрацьованої програми та її код подати у звіті.
- Приклад схеми підключення ЖК-екрану



Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 147



- Приклад програми налагодження зв'язку програми з ЖК-екраном та виведення певного значення на нього

```
#include <LiquidCrystal.h> // долучення бібліотеки роботи з ЖК-екраном
```

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

```
// ініціалізація бібліотеки з номерами інтерфейсних виводів arduino,
```

```
// по порядку, до яких підключено виводи ЖК-екрану
```

```
// по порядку RS, E (Enable), DB4, DB5, DB6, DB7
```

```
void setup() {
```

```
    lcd.begin(16, 2); // встановлення для ЖК-екрану кількості колонок та  
    рядків
```

```
}
```

```
void loop() {
```

```
    lcd.setCursor(0, 1); // переміщення курсору на стовпчик 0 та рядок 1
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 148

// створюємо змінну з кусочків рядка

```
String text = "took " + String(1555) + "ms";
```

```
// функція String() перетворює число на шматочок рядка
```

```
lcd.print(text); // виведення на ЖК-екран
```

```
delay(10); // затримка необхідна для чіткості сповіщень, що виводяться
```

```
}
```



Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	<i>Екземпляр № 1</i>	<i>Арк __ / 149</i>

Ч А С Т И Н А І І

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 150

Лабораторна робота № 1

Операції з формами Windows

Основи GDI+

Мета роботи:

Здобути навички використання технології GDI+ при створенні програм відображення інформації у середовищі розробки Microsoft Visual C++.

Короткі відомості, щодо порядку виконання роботи:

Відповідно до завдання на сьогоднішнє заняття, ми повинні створити таке програмне застосування, яке виконувало б такі функції роботи з зображенням, яке містить прозоре тло.

1. Створення нового проекту:

Файл --- Створити --- Проект --- Visual C++ --- Windows Forms

Отже створено шаблон майже пустого проекту.

Можемо одразу ж запустити його на виконання.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 151

2. Виклик редактора редагування форми (конструктора форми):

Під час попередніх лабораторних робіт ми вже користувались **оглядачем рішення**. У ньому у вигляді структури-дерева відображено всі елементи проекту: основні файли коду, ресурсів тощо. У Visual Studio 2017 цей елемент типово зверху справа.

Знайдемо у ньому позицію Form1.h або щось подібне. Подвійний клік - і потрапляємо у **конструктор форми**.

3. Виклик вікна властивостей візуальних елементів форми

ПКМ --- і у **контекстному меню** обрати пункт **властивості**.

Проглянемо властивості форми та змінимо **колір форми** на власний смак. За допомогою редагування властивості **Back Color** на **панелі властивостей** задамо колір, що не зустрічатиметься у нас у рисунку. Рекомендується задати колір за назвою, наприклад, **Dark Magenta**.

Ми бажаємо, щоб форма була прозора під час роботи і видним було лише зображення завантаженого рисунку. Тому знайдемо властивість форми **Transparency Key** і задамо той самий колір, який обрали для тла форми.

Запустимо проект і впевнимся, що форма прозора.

Тепер задамо властивість, яка автоматично розтягуватиме форму на весь екран під час виконання програми. Це властивість **Window State**. Виставимо цю властивість у значення **Maximized**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 152

Тепер змінимо дещо стиль форми. Наприклад, ми хочемо форму, без стиля. Навіщо нам стиль, якщо наша форма буде невидимою. Задамо властивість **Form Border Style** як **None**.

Стрічка заголовку вікна зникла, тому перетягти вікно буде не можливо. Знову ж таки, навіщо перетягати форму, яку і так не видно.

Форма зовсім пуста, але тепер можемо додати мінімальні елементи. Додамо елемент відображення графічної інформації та елементи управління - кнопки.

У якості елемента виведення картинки додамо на форму елемент **imageBox**. Для того, щоб цей елемент розтягнувся по всій формі, зайдемо у його властивість **Dock** і задамо їй значення **Fill**. Цей елемент одразу ж займе всю площу форми.

У верхній правий кут форми додамо три кнопки. Перша кнопка «згорнути», тому змінимо її властивість **Text** на **_**. Друга кнопка «закрити» – її текст можна задати як **X**. І третя кнопка «пуск/стоп» нашої програми. Задамо їй текст **GO**.

Подвійним кліком на кнопці **_** відкриємо для редагування функцію обробки події кліку **buttonx_Click**. Акуратно додамо туди рядок зміни стану вікна:

```
Form1::WindowState::set(System::Windows::Forms::FormWindowState::Minimized);
```


Подвійним кліком на кнопці **X** відкриємо для редагування обробки події **buttonx_Click**. Додамо в цю функцію рядок виходу з програми:

```
Application::Exit();
```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 153

Запустимо програму і перевіримо правильність.

Для того, щоб кнопки були завжди у правому верхньому куті форми, задамо властивість кнопок **Anchor** як **Top, Right**.

Витягнемо на форму також елемент **таймера**  **Timer**. Він нам буде потрібний для керування зображенням.

Приступимо до задавання локальних та глобальних змінних у основній програмі.

Знайдемо такі рядки коду:

....

public:

// **шукане місце**

Form1(void)

{

....

Та додамо між **public:** та **Form1(void)** оголошення змінних:

static float x=100, y=100, dx=10.0, dy=10.0, sx=1.0, sy=1.0;

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 154

Призначення специфікатора **static** вивчимо дещо пізніше на лекціях. Змінні **x** та **y** відповідатимуть за координату, де відобразатиметься картинка. Змінні **dx** та **dy** будуть слугувати елементами зміни координат **x** та **y**. І, нарешті, **sx** та **sy**. Ці змінні показуватимуть знак зміни координати **x** або **y**. Якщо **sx** та **sy** додатні 1-ці, то координати **x** **y** збільшуватимуться. І, навпаки, від'ємні 1-ці сприятимуть зменшенню координат **x** **y**.

Працюватиме формула:

**нове значення координати = старе значення координати +
значення зміни координати * на знак зміни координати;**

Як бачимо тільки-но створеним змінним надано одразу значення ініціалізації.

Додамо на рядок нижче змінні-елементи:

```
static Image^ MyImage;  
static Graphics^ graph;
```

Перша змінна-елемент зберігатиме посилання на зображення. Друга змінна зберігатиме посилання на елемент, за допомогою якого і будемо виводити графічне зображення на форму. Типами тут є **Image^** та **Graphics^**. Елементами змінними будуть **MyImage** та **graph**. Нехай не лякає така форма запису зі знаком **^**. Це ознака спеціального шаблонного середовища розробки, яке підтримує **WindowsForms**.

Додамо нижче також таку змінну-елемент:

```
Bitmap^ myBitmap;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 155

Ця змінна **myBitmap** відіграватиме роль графічного буфера. Спочатку виводитимемо рисунок у ньому, а потім, виводитимемо буфер у графічний елемент **imageBox** на формі.

Тепер основні елементи готові і можемо запрограмувати основний алгоритм.

Зайдемо знову на форму **Form1.h** та викличемо її властивості. Знайдемо на панельці властивостей зверху іконку з блискавкою – це перелік подій з елементом, чиї властивості переглядаються.

Знайдемо для форми властивість **load**. Заповнимо поряд комірку назвою функції, що запуститься, коли форма завантажуватиметься. Задамо назву **Form1_load**. У вікні редактора коду задамо бажані рядки ініціалізації. Отже у функцію **Form1_load** додамо рядки:

```

SetStyle(ControlStyles::UserPaint, true);
SetStyle(ControlStyles::AllPaintingInWmPaint, true);
SetStyle(ControlStyles::DoubleBuffer, true);

```

Вони включають режим буферизації при виведенні зображень на форму.

Після них можна забезпечити рядок підготовки зображення з файлу у раніше підготовлену змінну-об'єкт.

```

MyImage = Image::FromFile("574.png");

```

Рисунок запропоновано у папці із завданням. Можна також скачати свій. Ефект буде більший, якщо рисунок буде з прозорим тлом.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 156

На цьому етапі ініціалізацію нашої форми для сприйняття графічних елементів можливо вважати виконаною.

Всім керуватиме таймер. Кнопкою **GO** ми запускатимемо його або призупинятимемо. Тому клікнувши на кнопку **GO** потрапимо до редагування функції обробки кліку по кнопці **button3_Click**:

```

if (Form1::timer1->Enabled == false)
    {
        Form1::timer1->Enabled = true;
        Form1::button3 ->Text = "STOP";
    }
else if (Form1::timer1->Enabled == true)
    {
        Form1::timer1->Enabled = false;
        Form1::button3 ->Text = "GO";
    }

```

Отже кнопка буде у нас перемикачем: **GO/ STOP**.

Тепер перейдемо до зображення форми і подвійно клікнемо по таймеру.

Попередньо таймеру виставимо інтервал лічби 50 мілісекунд.

У середовищі розробки коду скоректуємо функцію реакції на закінчення лічби таймеру: **timer1_Tick** .

Рядки будуть такими:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 157

x = x + (sx*dx) ;

y = y + (sy*dy);

Це рядки формування положення зображення на формі.

Для краси, ще додамо контроль за невиходом зображення за межі форми:

if ((x >= Form1::Width-100) || (x<=0)) sx=sx*(-1);

if ((y >= Form1::Height-100) || (y<=0)) sy=sy*(-1);

Коли сформовані і перевірені координати :

- задамо новий чистий графічний буфер для виведення картинки:

myBitmap = gcnew Bitmap(Form1::Width, Form1::Height);

- підготуємо об'єктний елемент, спеціально передбачений для роботи з графікою. Вище ми вже задали його з типом **Graphics^** . А зараз ще і ініціалізуємо у прив'язці до графічного буфера:

graph = Graphics::FromImage(myBitmap);

- рисуємо картинку у заданих координатах прямо у графічний елемент графічного буфера:

graph->DrawImage(MyImage, x, y, 100.0, 100.0);

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 158

(Цифри **100.0** означають тут ширину і висоту зображення, що відобразатиметься.)

- виводимо графічний буфер як картинку у елемент відображення графіки **pictureBox**:

Form1::pictureBox1->Image = myBitmap;

Запускаємо і перевіряємо.

Змінюючи властивість таймера (значення у мілісекундах), та значення приросту координат картинки dx та dy при ініціалізації, добиваємося бажаної швидкості та плавності переміщення картинки по екрану.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 159

Лабораторна робота № 2 (mini)

Використання пам'яті програмами в Windows

Мета роботи:

Здобути навички відстеження використання пам'яті програмами Windows та навички раціональної побудови коду для мінімізації використання пам'яті.

Короткі відомості, щодо порядку виконання роботи:

Відповідно до завдання до попередньої лабораторної роботи, ми створили таке програмне застосування, яке виконувало виведення зображення з певною періодичністю.

По кожному такту таймера виділяється новий (чистий) фрагмент графічної пам'яті **Bitmap** за допомогою команди **gcnew**.

Негатив – виділення все нових і нових фрагментів графічної пам'яті у циклі. Використані фрагменти при цьому прямо не звільняються.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 160

1. Перевіряємо та коректуємо за потреби код проекту

Ось так мають виглядати оголошені змінні:

```
static float x=100,y=100,dx=10.0,dy=10.0,sx=1.0,sy=1.0;
```

```
static Image^ MyImage;
```

```
static Graphics^ graph;
```

```
Bitmap^ myBitmap;
```

На формі встановлено елемент **pictureBox**. Він розміщений на задньому плані (що). Йому задана властивість **Dock** як **Fill**.

У функції (процедурі) **Form1_load** обробки первинного завантаження форми тільки 4 рядки:

```
SetStyle(ControlStyles::UserPaint, true);
```

```
SetStyle(ControlStyles::AllPaintingInWmPaint, true);
```

```
SetStyle(ControlStyles::DoubleBuffer, true);
```

```
MyImage = Image::FromFile("574.png");
```

У функції (процедурі) **timer1_Tick** обробки спрацювання таймеру всього 6 рядків:

```
if ((x >= Form1::Width-100) || (x<=0)) sx=sx*(-1);
```

```
if ((y >= Form1::Height-100) || (y<=0)) sy=sy*(-1);
```

```
myBitmap = gcnew Bitmap(Form1::Width, Form1::Height);
```

```
graph = Graphics::FromImage(myBitmap);
```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 161

graph->DrawImage(MyImage, x, y, 100.0, 100.0);

Form1::pictureBox1->Image = myBitmap;

2. Запуск готового проекту

Запустіть проект на відлагодження інструментом  Локальний відладчик.

Натисніть на кнопку **Go** запуску генерації графічних об'єктів **Bitmap**.

3. Перевірка оперативної пам'яті, виділеної під процес виконання програми:

3.1. Не зупиняючи програму, переглянути вікно у Visual Studio 2017, яке відповідає за виведення в режимі реального часу графіку використання пам'яті програмою.

Відобразити цей графік у звіті.

3.2. З Панелі завдань **Windows** запустити **Диспетчер завдань**. Відкрити там вкладку **Процесів**. Знайти там свій процес «назва_проекту.exe» і занотувати у звіт дані про використання оперативної пам'яті.

4. Модифікація програми

4.1. Видалимо з функції **timer1_Tick** виділені рядки:

```
if ((x >= Form1::Width-100) || (x<=0)) sx=sx*(-1);
```

```
if ((y >= Form1::Height-100) || (y<=0)) sy=sy*(-1);
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 162

```
myBitmap = gcnw Bitmap(Form1::Width, Form1::Height);
```

```
graph = Graphics::FromImage(myBitmap);
```

```
graph->DrawImage(MyImage, x, y, 100.0, 100.0);
```

```
Form1::pictureBox1->Image = myBitmap;
```

4.2. І перенесемо їх у функцію **Form1_load**, щоб вона виглядала так:

```
SetStyle(ControlStyles::UserPaint, true);
```

```
SetStyle(ControlStyles::AllPaintingInWmPaint, true);
```

```
SetStyle(ControlStyles::DoubleBuffer, true);
```

```
MyImage = Image::FromFile("574.png");
```

```
myBitmap = gcnw Bitmap(Form1::Width, Form1::Height);
```

```
graph = Graphics::FromImage(myBitmap);
```

Тепер виділення буферної графічної пам'яті **Bitmap** буде відбуватись лише одноразово – під час первинного завантаження форми.

4.3. Додамо до функції **timer1_Tick** виділений рядок, який замість виділення нової графічної пам'яті буде просто очищати ту, що виділена одноразово:

```
if ((x >= Form1::Width-100) || (x<=0)) sx=sx*(-1);
```

```
if ((y >= Form1::Height-100) || (y<=0)) sy=sy*(-1);
```

```
graph->Clear(Color::Transparent);
```

```
graph->DrawImage(MyImage, x, y, 100.0, 100.0);
```

```
Form1::pictureBox1->Image = myBitmap; У функції
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 163

4.4. Запускаємо програму і впевнюємося у її працездатності.

5. Перевірка оперативної пам'яті, виділеної під процес виконання модифікованої програми:

5.1. Не зупиняючи програму, переглянути вікно у Visual Studio 2017, яке відповідає за виведення в режимі реального часу графіку використання пам'яті програмою.

Відобразити цей графік у звіті.

5.2. З **Панелі завдань Windows** запустити **Диспетчер завдань**. Відкрити там вкладку **Процесів**. Знайти там свій процес «**назва_проекту.exe**» і занотувати у звіт дані про використання оперативної пам'яті.

5.3. Порівнюємо результати перевірок використання оперативної пам'яті до та після модифікації.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 164

Лабораторна робота № 3

Застосування GDI+ та Windows Forms

Мета роботи:

Здобути навички використання бібліотеки GDI+ при створенні програм на базі технології Windows Forms у середовищі розробки Microsoft Visual C++.

Короткі відомості, щодо порядку виконання роботи:

Відповідно до завдання на сьогоднішнє заняття, ми повинні створити таке програмне застосування, яке виконувало деякі функції, подібні до функцій графічного редактора.

1. Створення нового проекту:

Файл --- Створити --- Проект --- Visual C++ --- CLR Windows Forms

Як і годиться, створено шаблон майже пуского проекту.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 165

Щоб сформувались ресурси форми потрібно запустити його на виконання(відладку).

2. Виклик редактора редагування форми (конструктора форми):

Під час попередніх лабораторних робіт ми вже користувались **оглядачем рішення**. У ньому у вигляді структури-дерева відображено всі елементи проекту: основні файли коду, ресурсів тощо. У Visual Studio 2017 цей елемент типово зверху справа.

У верхньому правому куті VisualStudio2017 знайдемо **Оглядач рішення**. У **Оглядачі рішення** знайдемо позицію **Form1.h**. Подвійним кліком **Form1.h** запускаємо вкладку **конструктор форми**.

3. Корекція властивостей форми

3.1. ПКМ на графічному відображенні форми --- і у **контекстному меню** обрати пункт **властивості**.

3.2. Задамо колір форми на власний смак за допомогою властивості **BackColor**.

Запустимо проект і впевнимися, що форма правильно відображається.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 166

3.3. Призначення майбутньої програми все ж передбачає наявність певної області для дій користувача. Тому, потягнувши за правий нижній край графічного відображення форми, дещо розширимо її.

3.4. Зробимо новий підпис назви форми: елемент **Text** – значення **MiniPaint 1.0** (або інше, що вам до вподоби).

Поки що форма пуста. Додамо деякі елементи. Почнемо з меню.

4. Додавання діалогового вікна вибору кольору

Нам дещо пощастило з цим. **Windows** має стандартне вікно вибору кольору. Його лише потрібно додати як елемент у форму. Знайдемо серед переліку у **Панелі елементів** елемент **ColorDialog** і витягнемо його на форму.

5. Додавання рядка меню

5.1. Знайдемо у **Панель елементів** та додамо на форму рядок меню – елемент **stripMenu**.

5.2. За замовчуванням він розташовується зверху форми. Змінимо це для оригінальності відображення форми. Властивість **LayoutStyle** – значення

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 167

VerticalStackWithOverflow. Тепер це буде меню –стовпчик. Для надійності відображення вирівнюємо меню по лівому краю форми. Властивість **Dock** задамо як **Left**.

5.3. Клікаючи по першому та наступних елементах меню створимо його початкову структуру:

```
File
| _____ Exit
Color
Pen
```

5.4. Для більшої виразності меню змінимо його шрифт. Задамо властивість **Font** на **10** пунктів.

5.4. Подвійно клікнувши на пункт **Exit** задамо процедуру **ExitToolStripMenuItem_Click**, що обробляє вибір цього пункту користувачем:

```
Application::Exit();
```

Одразу ж перевіряємо, чи виконуються запрограмовані нами дії, запустивши нашу програму.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 168

5.5. Подвійно клікнувши на пункт **Color** задамо процедуру **ColorToolStripMenuItem_Click**, що обробляє вибір цього пункту користувачем, тобто викликає вікно вибору кольору.

5.5.1. Додаємо код у функцію **colorToolStripMenuItem_Click**

```
colorDialog1->ShowDialog();  
MyColor = colorDialog1->Color;
```

Перша команда відкриватиме вікно вибору кольору. Друга – записуватиме обраний користувачем колір у змінну.

Є одна проблема: змінної **MyColor** ми ще не оголошували.

5.5.2. Додамо змінну **MyColor**.

Додамо програмні змінні до програми роботи з формою як об'єктом.

Повернемось у **Оглядач рішення** і натиснемо **ПКМ** на елементі **Form1.h** дерева проекту. Оберемо команду **Перейти до коду**.

У кодї опису об'єкту **Form1** знайдемо такі рядки коду:

....

public:

```
Form1(void)  
{  
InitializeComponent();
```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 169

}

шукане місце

private: System::Windows::Forms::Tool ...

Та додамо у виділене місце оголошення програмних змінних. Для початку оголосимо змінну обраного кольору пера:

Color MyColor;

Color тут виступає як тип. Технологія **Windows Forms** передбачає широкий набір додаткових типів змінних, спеціального призначення. Тепер змінна **Color** зможе приймати лише стандартні системні кольори.

Запускаємо програму та перевіряємо, чи виконуються запрограмовані дії.

6. Підготовка елемента вибору інструменту малювання

6.1. Праворуч та знизу від стовпчика меню розмістимо елемент **Panel** середньої ширини. Задамо його властивості **Dock** як **Bottom** (тобто вирівнювання по основі форми).

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 170

6.2. У елемент **Panel** помістимо елемент **comboBox**. Задамо йому **Font** як **14** шрифт та вирівнювання **Dock** до низу **Bottom** та ліворуч **Left**. Можемо також задати стиль **FlatStyle** як **Flat**.

6.3. Елемент **comboBox** слугує для вибору користувачем певного варіанту. У нашому випадку варіантами будуть типи інструментів. Варіанти для вибору записуються у властивість **Items** цього елемента. Клікнемо на властивість **Items** і у вікні, що з'явилось введемо варіанти вибору:

Curve

Line

CirclingLines

Curve – це буде інструмент малювання кривої. **Line** – це буде інструмент малювання лінії. **CirclingLines** – інструмент малювання відцентрових ліній.

6.4. Початковий вибір **Line**, який буде безпосередньо після запуску програми, запишемо у властивість **Text** елемента **comboBox**.

Запустивши програму, перевіримо функціональність доданих елементів.

6.5. Якщо під час виконання виявилось, що у елемента **comboBox** недостатня ширина для виведення слів вибору, то розтягнемо елемент **comboBox** на потрібну ширину.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 171

Як бачимо на формі залишилося ще досить місця. Ми його лишили для елемента – поля редагування користувачем рисунку.

7. Додавання поля редагування рисунку

7.1. Додамо елемент **pictureBox** на вільну площу на формі.

7.2. Якщо задати для **pictureBox** властивість **Dock** як **Fill**. То він заповнить залишок місця на формі. І буде автоматично збільшуватись, якщо користувач змінюватиме розміри форми.

Запустимо програму і перевіримо функціональність.

Поступово почнемо реалізовувати основний код.

8. Задання початкового набору змінних

8.1. Знайдемо місце де задавали змінну **MyColor** і одразу після неї задамо інші:

Color MyColor;

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 172

Graphics^ MyGraph;

Graphics^ MyCloneGraph;

Bitmap^ MyBitmap;

Bitmap^ MyCloneBitmap;

Pen^ MyPen;

static System::Decimal MyPenWidth = 10;

Point ^ MyPrevPoint;

Point ^ MyCurrPoint;

static String ^ MyTool;

SolidBrush^ MyBrush;

Пояснимо призначення кожної змінної (для тих, хто особливо поспішає, можна пропустити цей фрагмент і перейти до пункту 8.2):

Color MyColor;

- зберігання даних про колір; змінна може приймати унормовані значення:

- повністю:

System::Drawing::Color::нормативна назва кольору

- скорочено:

Color::нормативна назва кольору;

Graphics^ MyGraph;

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 173

- об'єкт поверхні для малювання; викликається з більшості візуальних елементів WindowsForms або з буферних (невізуальних) об'єктів типу Bitmap;

Bitmap^ MyBitmap;

- об'єкт графічної пам'яті (невізуальний); часто використовується як буфер; після проведення всіх графічних операцій з буфером, буфер надсилається у візуальний елемент (підвищення швидкості та плавності виконання);

Bitmap^ MyCloneBitmap;

- аналогічно MyBitmap, але для зберігання проміжних станів картинки: попередніх та наступних, відносно поточного;

Graphics^ MyCloneGraph; - аналогічно MyGraph, але для роботи з буфером MyCloneBitmap;

Pen^ MyPen; - об'єкт стилю малювання ліній;

static System::Decimal MyPenWidth = 10;

- статична змінна для зберігання ширини малювання ліній; тип System::Decimal аналог типу int; використовується для деяких графічних WindowsForms;

Point ^ MyPrevPoint;

- змінна для зберігання точки початку малювання мишкою; тип Point передбачає розкладання змінної на MyPrevPoint::X та MyPrevPoint::Y; змінну можна задати і так MyPrevPoint = Point(X, Y);

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 174

Point ^ MyCurrPoint;

- аналогічно MyPrevPoint, але для зберігання поточної точки малювання;

static String ^ MyTool;

- змінна для зберігання рядку назви поточного інструменту малювання; тип String аналог типу string з бібліотеки <string>

SolidBrush^ MyBrush; - об'єкт стилю заливки;

Перевіримо правильність задання змінних, запустивши програму.

9. Опис функції початкової ініціалізації

Виконаємо початкову ініціалізацію прописавши функцію, для події завантаження форми. Для цього у конструкторі форми виділимо нашу форму – зайдемо у її властивості – оберемо перелік подій для форми за допомогою інструмента «блискавка» – знайдемо подію Load – подвійно клікнемо на ній і у опису функції **Form1_Load** пропишемо код:

```
MyBitmap = gnew Bitmap(
    Form1::pictureBox1->Width,
    Form1::pictureBox1->Height
);
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 175

MyGraph = Graphics::FromImage(MyBitmap);

MyGraph->SmoothingMode =

System::Drawing::Drawing2D::SmoothingMode::AntiAlias;

SetStyle(ControlStyles::UserPaint, true);

SetStyle(ControlStyles::AllPaintingInWmPaint, true);

SetStyle(ControlStyles::DoubleBuffer, true);

SetStyle(ControlStyles::OptimizedDoubleBuffer, true);

Пояснення:

Тут ми готуємо графічний буфер **MyBitmap** розмірами з елемент **pictureBox1** нашої форми, куди і будемо пізніше відображати картинку.

Потім дістаємо з нашої буферної картини об'єкт доступу до функцій графіки за допомогою функції **Graphics::FromImage**.

Далі ми включаємо режим згладженого відображення графічних елементів **SmoothingMode**.

І ще чотири рядки стандартного коду функції **SetStyle**, який у загальному призначений для зниження мерехтіння зображення та для максимально можливої плавності виведення на дисплей, але не є об'єктом нашого розгляду.

Тепер можна написати перший графічний інструмент.

Запустимо програму, та перевіримо правильність здійснених дій.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 176

10. Загальний принцип малювання

Подія «мишка натиснута» - захоплення мишки (відносно інших вікон та елементів), ініціалізація програмних інструментів малювання, фіксація початкової точки.

Подія «мишка рухається» - якщо мишка захоплена, то відображення за початковою та поточною точкою перебування мишки.

Подія «мишка відпущена» - якщо мишка була захоплена, то кінцеве відображення графіки та звільнення мишки (для інших вікон та елементів), завершення використання програмних інструментів малювання, переведення фокусу на область малювання.

11. Інструмент малювання кривих «Curve»

11.1. Звертаємося до нашої форми – клікаємо на **pictureBox1** – викликаємо його властивості – «блискавкою» відкриваємо події – обираємо подію **MouseDown** – подвійний клік на цій події – прописуємо код у функцію **pictureBox1_MouseDown**, що оброблює цю подію:

```
if(!pictureBox1->Capture) return;
```

```
MyTool = comboBox1->Text;
```

```
MyPen = gcnnew Pen(MyColor, (float) MyPenWidth);
```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 177

MyPrevPoint = Point(e->X, e->Y);

pictureBox1->Capture = true;

Пояснення:

Функцією **if** перевіряємо, чи захоплено мишку над елементом **pictureBox1**. Якщо ні (наприклад над іншим елементом), тоді виходимо з функції оператором **return**, тобто, не виконуємо ніяких дій. Інакше, переходимо до наступних операторів функції.

Змінній **MyTool** присвоюємо ім'я графічного інструменту, який користувач обрав у випадяючому списку **comboBox1**.

Функцією **Pen**, настраюємо тип ліній, що відобразатимуться, поки що тільки по кольору та ширині. Вірніше створюється об'єкт **MyPen** для наладки відображення ліній.

Далі у змінній **MyPrevPoint** зберігається позиція **X, Y** початкової точки малювання. Дані про точку знаходження мишки видобуваємо з системного об'єкта – **e**. Цей об'єкт не створений нами і передається автоматично операційною системою у функцію обробки події **MouseDown** (якщо придивитись до параметрів функції **pictureBox1_MouseDown**, то там і можна його знайти).

І останнє: надійне задання, того, що мишка захоплена, за допомогою властивості **Capture**, саме для елемента **pictureBox1**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 178

11.2. Аналогічно для події **MouseMove**, прописуємо код у функцію **pictureBox1_MouseMove**:

```
if(!pictureBox1->Capture) return;
```

```
if (MyTool == "Curve") {MyDrawingCurve(e);}
```

Тут, зрозуміло, нічого не виконуватиметься, якщо мишка не захоплена.

Якщо користувачем було обрано малювання кривої «**Curve**», то буде викликана функція її прорисовки.

Придумаємо назву цій функції **MyDrawingCurve()**. Опишемо цю функцію трохи згодом. Але передамо у якості параметра цій функції системний об'єкт – **e**, його системний тип **System::Windows::Forms::MouseEventArgs^**, з нього надалі братимемо інформацію про **X**, **Y** мишки, коли описуватимемо код функції **MyDrawingCurve(e)**.

11.3. Аналогічно для події **MouseUp**, прописуємо код у функцію **pictureBox1_MouseUp**:

```
if(!pictureBox1->Capture) return;
```

```
delete MyPen;
```

```
this->pictureBox1->Capture = false;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 179

Тут, зрозуміло, також нічого не виконуватиметься, якщо мишка не була захоплена раніше.

Видаляються об'єкт **MyPen**; властивостей відображення ліній.

Звільняється мишка для інших візуальних елементів.

11.4. Функція обробки переміщення мишки для інструменту «Curve» користувача

Цю функцію прописуємо одразу ж після функції **pictureBox1_MouseUp**

```
private: System::Void pictureBox1_MouseUp(
    System::Object^ sender,
    System::Windows::Forms::MouseEventArgs^ e)
{
    ...
}
```

тут буде наша функція малювання кривої

Функція викликатиметься постійно, доки мишка рухається.

Текст опису цієї функції:

```
private: void MyDrawingCurve(System::Windows::Forms::MouseEventArgs^ e)
{
    MyCurrPoint = Point(e->X, e->Y);
    MyGraph->DrawLine(MyPen, *MyPrevPoint, *MyCurrPoint);
    Form1::pictureBox1->Image = MyBitmap;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 180

```

    MyPrevPoint = MyCurrPoint;
}

```

Тут спочатку визначаємо поточну позицію мишки і зберігаємо у змінну **MyCurrPoint**.

Далі за допомогою функції **DrawLine** на графічмалюємо лінію (прямий відрізок) з наладкою стилю, взятою з **MyPen** від точки **MyPrevPoint** до **MyCurrPoint**.

Буфер з лінією відправляємо до візуального елемента **pictureBox1**, що на нашій формі.

Готуємо точку відліку для наступного відрізка: поточна точка стає попередньою.

Запускаємо програму і намагаємося щось намалювати.

12. Оцінка недоліків створеної програми

- після малювання змінимо розміри форми, ще щось намалюємо, значно збільшимо форму, знову спробуємо щось намалювати – відмітимо некоректності роботи;
- намалюємо, щось по всій поверхні форми, спробуємо напівприкрити форму, наприклад, вікном провідника, та знову прибрати покриваюче вікно – відмітимо некоректності роботи;

-> спробуйте пояснити причини поведінки програми, та некоректності, що виникають.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 181

Лабораторна робота № 4 (продовження л.р. №3)

Застосування GDI+ та Windows Forms

Мета роботи:

Здобути навички використання бібліотеки GDI+ при створенні програм на базі технології Windows Forms у середовищі розробки Microsoft Visual C++.

Короткі відомості, щодо порядку виконання роботи:

Відповідно до завдання на сьогоднішнє заняття, ми повинні продовжити створення такого програмного застосування, яке виконувало б деякі функції, подібні до функцій графічного редактора.

1. Згадаємо структуру створеного проекту

Події (далі у «») та їх обробка програмою:

після «завантаження форми»

- виділення частини пам'яті під графічний буфер

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 182

**- виклик графічного об'єкту для малювання у
графічному буфері**

після «вибір меню - Color»

- виклик модального вікна вибору кольору**

після «обрано колір»

- збереження обраного кольору**

після «мишка натиснута»

- визначення інструменту малювання**
- задавання кольору та товщини ліній**
- фіксація початкової точки**
- захоплення мишки (відносно інших вікон та елементів)**

після «мишка рухається»

- визначення поточної точки перебування мишки**
- малювання фігури у буфері за початковою та поточною точками мишки**
- виведення буферу на візуальний елемент форми**
- призначення попередній точці значення поточної**

після «мишка відпущена»

- малювання фігури кінцеве (якщо потрібно)**
- завершення використання об'єктів стилю малювання**
- звільнення мишки (для інших вікон та елементів)**
- переведення фокусу на область малювання**

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 183

Нами описані основні моменти, пов'язані з реалізацією програми малювання. Наша програма вже може малювати криві, але не вільна від деяких проблем. Одна з них – це незмінність області для малювання при збільшенні розмірів самої форми.

2. Зміна розміру області малювання

Спробуємо забезпечити незалежну від інших подій зміну розміру області малювання за бажанням користувача.

Попередньо для того, щоб бачити змінювану область на фоні форми, змінимо **Back color** для **pictureBox1** на зовсім трохи відмітний від кольору форми. Також елемент **pictureBox1** вже не повинен автоматично розтягуватись при зміні розміру форми, а тільки за бажанням користувача. Тому змінимо для **pictureBox1** властивість **Dock** на **None**.

На форму у **правий нижній кут** елемента **pictureBox1** (це наша область малювання) помістимо маленьку кнопку **button1** (кнопка може навіть трохи «сховатись» за границю області видимості форми).

Задамо для **button1** розміри **10x10** (властивість **Size**).

Навіщо нам ця кнопка ? Будемо використовувати її як інструмент для розтягування **pictureBox1**. Одночасно змінюватимемо розміри графічного буфера **MyBitmap**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 184

Ці дії варто виконувати, якщо **Mouse** (мишку) натиснуто на **button1** і відбувається **Move** (рух) мишки.

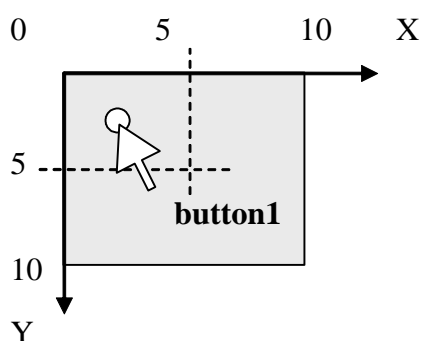
Опишемо функцію обробки події переміщення мишки по **Button1**:

button1 — ПКМ — властивості — «блискавка» — подія **MouseMove** — подвійний клік:

Автоматично створюється функція **button1_MouseMove**:

```
System::Void      button1_MouseMove(System::Object^      sender,
System::Windows::Forms::MouseEventArgs^ e)
{
    сюди писатимемо код функції
}
```

Звернемо увагу: **e** – параметр функції, звідки отримуватимемо координати переміщення мишки (але вже відносно лівого верхнього кута **button1**). Координати ми будемо отримувати у вигляді:



e->X та **e->Y** – координати положення мишки відносно 0.

А ми будемо використовувати:

e->X - 5 та **e->Y - 5** – координати положення мишки відносно центру кнопки.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 185

Тоді якщо мишка знаходитиметься так, як показано на рисунку, то $(X - 5) < 0$ та $(Y - 5) < 0$. Значить програма зміщуватиме кнопку **button1** та правий нижній край **pictureBox1** вгору та вліво. І навпаки.

Опишемо код функції **button1_MouseMove**.

Спочатку убезпечимо себе від хибного спрацювання. Тобто буде відбуватись одразу (**return**) вихід з функції, якщо не було захоплено (**Capture**) мишку на **button1**,

```
if(!button1->Capture) return;
```

Оголосимо локальні змінні для зберігання нових розмірів для **pictureBox1** та **MyBitmap**.

```
int NewWidth, NewHeight;
```

Далі визначатимемо нову ширину, але не меншу **1 піксела** та (**&&**) не ширшу ніж ширина вільної області на формі (цей параметр співпадає з шириною **panel1->Width**, яка під **pictureBox1**).

```
if (( 1 <= (pictureBox1->Width + e->X-5))&&( panel1->Width >=(pictureBox1->Width + e->X-5)))
```

```
{NewWidth=pictureBox1->Width + e->X-5;}
```

```
else
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 186

{NewWidth=pictureBox1->Width;}

Тут, якщо умови не виконуються, то новому розміру призначаємо старий розмір.

Далі те саме по висоті. Межею зменшення буде **1 піксел**. Межею збільшення прийемо положення **panel1->Location.Y**, яка знаходиться на формі під **pictureBox1**.

```

if (((panel1->Location.Y) >= (pictureBox1->Height + e->Y-5)) &&
      (1 <= (pictureBox1->Height + e->Y-5)))
    {NewHeight=pictureBox1->Height + e->Y-5;}
else
    {NewHeight=pictureBox1->Height;}

```

Тут так само, якщо умови не виконуються, то новому розміру призначаємо старий розмір.

Після визначення нових розмірів призначимо їх спочатку **pictureBox1**.

```

pictureBox1->Width = NewWidth;
pictureBox1->Height = NewHeight;

```

Зробимо границю **pictureBox1** видимою на час зміни розмірів:

```

pictureBox1->BorderStyle = BorderStyle::FixedSingle;

```

Перемістимо **button1** кнопку за мишкою (так вона буде завжди під мишкою):

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 187

button1->Location = Point(pictureBox1->Location.X+NewWidth-5,NewHeight-5);

Тепер попрацюємо над зміною розмірів графічного буфера **MyBitmap**.

Етапи будуть такі:

1) клонування основного буфера у тимчасовий буфер за допомогою функції

Clone:

```
MyCloneBitmap =  
MyBitmap->Clone(  
    Rectangle( 0, 0, MyBitmap->Width, MyBitmap->Height),  
    MyBitmap->PixelFormat  
);
```

Тут у тимчасовий буфер взято область зображення по границям, визначеним прямокутником **Rectangle**. Він покриває всю **MyBitmap** по ширині **Width** і висоті **Height**.

2) видалення буфера зі старими розмірами та його об'єкту поверхні для малювання

```
MyBitmap->~Bitmap();  
MyGraph->~Graphics();
```

3) створення буфера з новими розмірами та його об'єкту поверхні для малювання

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 188

MyBitmap = gcnw Bitmap(NewWidth, NewHeight);

MyGraph = Graphics::FromImage(MyBitmap);

4) розміщення зображення, що було, у новий буфер

MyGraph->DrawImage(MyCloneBitmap, Point(0,0));

5) видалення тимчасового буферу

MyCloneBitmap->~Bitmap();

6) виведення зображення на візуальний елемент форми

Form1::pictureBox1->Image = MyBitmap;

Закінчення зміни розмірів області малювання буде відбуватись по події кнопки **MouseUp**. Так само як і у попередньому випадку викличемо функцію **button1_MouseUp** обробки події **MouseUp** через інструмент «блискавка».

Код для цієї функції передбачає такі інструкції.

1) вихід з функції, якщо мишка не була захоплена до цього кнопкою **button1**

if(!button1->Capture) return;

2) звільнення мишки

button1->Capture = false;

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 189

3) повернення стилю елемента **pictureBox1** до попереднього

```
pictureBox1->BorderStyle = BorderStyle::None;
```

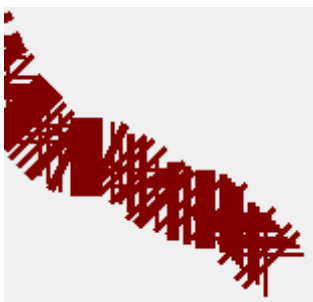
4) переведення фокусу з кнопки на область малювання **pictureBox1**

```
pictureBox1->Focus();
```

Запустимо програму і перевіримо функціональність.

Серед графічних інструментів користувача нами створено лише інструмент малювання кривої «**Curve**».

Графічний інструмент «**Curve**» працює, але крива відображається досить дивно – з проміжками.



Спробуємо виправити цю ситуацію.

Створимо інструмент **SmoothCurve**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 190

3. Створення графічного інструменту користувача для малювання згладжених кривих «SmoothCurve»

3.1. Повернемося у вкладку конструктора форми і для елемента **comboBox1** у властивості **Items** у Редакторі колекції рядків додамо рядок **SmoothCurve**.

Цей графічний інструмент буде використовувати стандартну функцію малювання **DrawCurve**.

3.2. Знайдемо місце у програмі, де ми оголошували всі змінні. Додамо після змінної **Point ^ MyCurrPoint**; нову змінну-масив

ArrayList^ MyCurveArray;

Ця змінна-масив буде зберігати поточні точки переміщення мишки. Масиву точок як параметру вимагає функція малювання кривої **DrawCurve**.

Тип масиву **ArrayList** обрано, через те, що розмір такого спец-масиву можна визначати динамічно (він не фіксований), адже ми не знаємо наперед, скільки точок пройде мишка.

3.3. Тепер визначимося з самим алгоритмом малювання кривої. Нами вже частково визначені головні функції

Події (далі у «») та їх обробка програмою:

після «мишка натиснута»

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 191

- визначення інструменту малювання (було)
- задавання кольору та товщини ліній (було)
- фіксація початкової точки (було)
- якщо обрано інструмент «SmoothCurve», (додамо)
 - виділення пам'яті під масив точок переміщення мишки
 - запис початкової точки у масив
- захоплення мишки (відносно інших вікон та елементів) (було)

після «мишка рухається»

- визначення поточної точки перебування мишки (було)
- якщо обрано інструмент «Curve» (додамо)
 - запис у масив поточної точки-позиції мишки
 - малювання фігури кривої у буфері за масивом точок
- виведення буферу на візуальний елемент форми (було)
- призначення попередній точці значення поточної (було)

після «мишка відпущена»

- якщо обрано інструмент «Curve» (додамо)
 - звільнення пам'яті від масиву точок переміщення мишки
- завершення використання об'єктів стилю малювання (було)
- звільнення мишки (для інших вікон та елементів) (було)
- переведення фокусу на область малювання (було)

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 192

Приступимо до редагування коду у файлі **Form1.h**. Для цього маємо вкладку **Form1.h**, або клікнемо по позиції **Form1.h** на дереві проекту у вікні **Оглядач рішення**.

3.4. Отже у функції **pictureBox1_MouseDown** допишемо рядок виклику функції початку малювання кривої після наявного рядку **MyPrevPoint = Point(e->X, e->Y);**

```
if (MyTool == "SmoothCurve") {MyBeginDrawingSmoothCurve(e);}
```

Зазначимо, що **MyBeginDrawingSmoothCurve(e);** функція яку ми назвали на власний розсуд та створимо самі. У якості параметра цієї функції ми передаємо **e** – об'єкт, з якого отримуватимемо координати переміщення мишки.

Далі зробимо додавання вже опису нової функції наприкінці файлу **Form1.h** перед знаком **};**:

тут буде наша функція початку малювання кривої

```
};  
}
```

Функція викликатиметься одноразово, при натисненні мишки.

Підготуємо основний оператор опису функції:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 193

private:

void

```
MyBeginDrawingSmoothCurve(System::Windows::Forms::MouseEventArgs^ e){
    тут описуватимемо оператори функції
}
```

Далі покроково опишемо функцію.

1) виділяємо пам'ять під динамічний масив точок (туди писатимемо позиції переміщення мишки)

```
MyCurveArray = gcnew ArrayList;
```

2) додаємо початкову точку до масиву

```
MyCurveArray->Add(MyPrevPoint);
```

3.5. А у функції **pictureBox1_MouseMove** допишемо рядок виклику функції неперервного малювання кривої. Функцію назвемо на власний розсуд **MyDrawingSmoothCurve** та додамо її після наявного рядку **if (MyTool == "Curve") {MyDrawingCurve(e);}**.

```
if (MyTool == "SmoothCurve") {MyDrawingSmoothCurve(e);}
```

Далі зробимо додавання вже опису нової функції **MyDrawingSmoothCurve** наприкінці файлу **Form1.h** перед знаком **};**:

тут буде наша функція малювання кривої

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 194

}

}

Функція викликатиметься неперервно, при утриманні та русі мишки.

Підготуємо основний оператор опису функції:

```
private: void MyDrawingSmoothCurve(
                System::Windows::Forms::MouseEventArgs^ e
                )
{
    тут описуватимемо оператори функції
}
```

Далі покроково опишемо функцію.

1) визначаємо поточну позицію мишки

```
MyCurrPoint = Point(e->X, e->Y);
```

2) додаємо поточну позицію мишки як наступний елемент масиву

```
MyCurveArray->Add(MyCurrPoint);
```

3) оголошуємо додатковий масив одномірний і надаємо йому фіксований розмір пам'яті відповідно до поточного розміру основного масиву (змушені до

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 195

такої проміжної операції, бо функція малювання кривої **DrawCurve** приймає масив точок тільки фіксованого розміру)

```
array<Point, 1> ^ MyCurveArray1 =
    gcnew array<Point, 1>(MyCurveArray->Count);
```

4) копіюємо вміст динамічного масиву у масив-копію фіксованого розміру

```
MyCurveArray->CopyTo(MyCurveArray1);
```

5) застосовуємо малювання кривої у графічний буфер функцією **DrawCurve** по масиву **MyCurveArray1**

```
MyGraph->DrawCurve(MyPen, MyCurveArray1);
```

6) видаляємо масив-копію

```
delete MyCurveArray1;
```

7) виводимо зображення з графічного буфера у візуальний елемент **pictureBox1**

```
pictureBox1->Image = MyBitmap;
```

3.6. Ну і нарешті у функції **pictureBox1_MouseUp** відпускання мишки після рядка

```
if(!pictureBox1->Capture) return;
```

прописуємо рядок

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 196

```
if (MyTool == "SmoothCurve") { MyEndDrawingSmoothCurve(e); }
```

3.7. Далі зробимо додавання опису нової функції **MyEndDrawingSmoothCurve** наприкінці файлу **Form1.h** перед знаком **};**:

тут буде наша функція кінця малювання кривої

```
};
```

```
}
```

Функція викликатиметься одноразово, при відпусканні мишки.

Підготуємо основний оператор опису функції:

```
private:                                                                                               void
MyEndDrawingSmoothCurve(System::Windows::Forms::MouseEventArgs^ e){
    тут описуватимемо оператори функції
}
```

Далі покроково опишемо функцію.

Тут потрібно звільнити пам'ять від динамічного масиву.

Вимагаються спеціальні команди.

1) обнулимо лічильник елементів масиву

```
MyCurveArray->Clear();
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 197

2) обрізаємо масив по поточному значенню лічильника елементів масиву (а він вже = 0)

MyCurveArray->TrimToSize());

Ось і всі оператори з дещо ускладненого алгоритму малювання кривої.

3.8. Запустимо і перевіримо ефект малювання за допомогою функції **DrawCurve**.

У звіт помістити два зображення для порівняння роботи інструментів **Curve** та **SmoothCurve**.

Спробуємо створити, ще кілька графічних інструментів користувача.

5. Створення графічного інструменту користувача для малювання ліній «Line»

5.1. Повернемося у вкладку конструктора форми і для елемента **comboBox1** і перевіримо, чи у властивості **Items** у Редакторі колекції рядків є рядок **Line**.

5.2. Щодо алгоритму малювання ліній, то потрібно буде додати оператори виклику функцій неперервного малювання та кінця малювання ліній. Тобто додати рядки у функції обробки подій «мишка рухається» та «мишка відпущена».

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 198

5.3. Приступимо до редагування коду у файлі **Form1.h**. Для цього маємо вкладку **Form1.h**, або клінемо по позиції **Form1.h** на дереві проекту у вікні **Оглядач рішення**.

Отже у функції **pictureBox1_MouseMove** допишемо рядок виклику іншої функції – функції малювання лінії після наявного рядку **if (MyTool == "SmoothCurve") {MyDrawingSmoothCurve(e);}**

```
if (MyTool == "Line") {MyDrawingLine(e);}
```

5.4. Далі зробимо додавання опису нової функції наприкінці файлу **Form1.h** перед знаком **};**:

тут буде наша функція малювання лінії

```
};  
}
```

Функція викликатиметься постійно, доки мишка рухається.

Підготуємо основний оператор опису функції:

```
private: void MyDrawingLine(System::Windows::Forms::MouseEventArgs^ e){  
    тут описуватимемо оператори функції  
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 199

Далі покроково опишемо функцію.

1) визначаємо поточну точку знаходження мишки

MyCurrPoint = Point(e->X, e->Y);

2) створюємо додатковий графчний буфер як клон від існуючого (по всій його ширині та довжині, що задано областю **Rectangle**) за допомогою функції-методу **Clone**; збережемо клон у нашу змінну **MyCloneBitmap**, яку створили раніше

**MyCloneBitmap = MyBitmap->Clone(
 Rectangle(0, 0, MyBitmap->Width, MyBitmap->Height),
 MyBitmap->PixelFormat
);**

Навіщо робити клон. При кожному переміщенні мишки у клон завантажується основне попереднє зображення і малюється лінія відповідно поточному положенню мишки. Якщо малювати тільки у основному буфері, то лінії накопичуватимуться по кожному переміщенні мишки. Але це вже буде не графічний інструмент «**Line**», а якась ...

3) створюємо об'єкт – поверхню малювання для клона графічного буфера

MyCloneGraph = Graphics::FromImage(MyCloneBitmap);

4) визначаємо згладжений режим малювання у клоні буфері

MyCloneGraph->SmoothingMode =

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 200

System::Drawing::Drawing2D::SmoothingMode::AntiAlias;

5) малюємо лінію у буфері клоні

MyCloneGraph->DrawLine(MyPen, *MyPrevPoint, *MyCurrPoint);

6) передаємо буфер клон на відображення у візуальний елемент форми **pictureBox1**

pictureBox1->Image = MyCloneBitmap;

5.5. Запустіть програму і спробуйте намалювати декілька ліній. Результат навряд чи задовольнив вас.

5.6. Скоректуємо ситуацію. Некоректність полягає в тому, що після малювання лінії у буфері клоні при кожному переміщенні мишки, при відпущенні мишки, кінцева лінія так і не переноситься у основний графічний буфер.

Виправимо це.

У функцію **pictureBox1_MouseUp** допишемо виклик функції, що завершуватиме процес малювання лінії. Допишемо її після рядка **if (MyTool == "SmoothCurve") {...};**. Назвемо цю функцію **MyEndDrawingLine**:

if (MyTool == "Line") {MyEndDrawingLine(e);}

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 201

Далі зробимо додавання опису нової функції наприкінці файлу **Form1.h** перед знаком **};** :

тут буде наша функція завершення малювання лінії

```
};  
}
```

Функція викликатиметься одноразово, коли кнопку мишки відпускає користувач.

Підготуємо основний оператор опису функції:

```
private: void MyEndDrawingLine(System::Windows::Forms::MouseEventArgs^  
e){  
    тут описуватимемо оператори функції  
}
```

Далі покроково опишемо функцію.

1) визначаємо кінцеву точку (захист, якщо функція переміщення не викликалаась)

```
MyCurrPoint = Point(e->X, e->Y);
```

2) включаємо режим згладженого малювання та малюємо лінію у основному буфері по кінцевим точкам

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 202

MyGraph->SmoothingMode =

System::Drawing::Drawing2D::SmoothingMode::AntiAlias;

MyGraph->DrawLine(MyPen, *MyPrevPoint, *MyCurrPoint);

3) виводимо в візуальний елемент **pictureBox1** основний графічний буфер **MyBitmap** з намальованою кінцевою лінією, на заміну зображення з буфера клону

Form1::pictureBox1->Image = MyBitmap;

Цих двох рядків достатньо.

5.7. Запустимо програму та перевіримо ефективність застосованого рішення.

6. Побудова нового графічного інструменту користувача

Під час побудови засобу малювання лінії була виявлена необхідність використання буфера клона. А що буде, якщо не застосувати клон, а малювати всі лінії при переміщенні мишки у основний буфер. Вийде доволі цікавий ефект. Використаємо це у наступному інструменті малювання.

6.1. Повернемося у вкладку конструктора форми і для елемента **comboBox1** і перевіримо, чи у властивості **Items** у Редакторі колекції рядків є рядок **CircligLines**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 203

6.2. Отже у функції `pictureBox1_MouseMove` допишемо рядок виклику нової функції – функції малювання всіх ліній у одному буфері. Додамо її після наявного рядку `if (MyTool == "Line") {MyDrawingLine(e);}`

Назвемо цю функцію `MyDrawingCirLines`.

```
if (MyTool == "CirclingLines") { MyDrawingCirLines(e);}
```

6.3. Зробимо додавання опису нової функції

після раніше доданої функції: десь наприкінці файлу `Form1.h` перед знаком `};`:

тут буде наша функція малювання новим графічним інструментом

```
};  
}
```

Функція викликатиметься, доки мишка рухається .

Підготуємо основний оператор опису функції:

```
private: void MyDrawingCirLines(System::Windows::Forms::MouseEventArgs^  
e){  
    тут описуватимемо оператори функції  
}
```

Далі покроково опишемо функцію.

1) визначаємо поточну точку знаходження мишки

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 204

MyCurrPoint = Point(e->X, e->Y);

2) включаємо режим згладженого малювання

MyGraph->SmoothingMode =

System::Drawing::Drawing2D::SmoothingMode::AntiAlias;

3) малюємо лінію від початкової точки, яку визначено при натисненні мишки, і до поточного положення мишки; малюємо у графічний буфер

MyGraph->DrawLine(MyPen, *MyPrevPoint, *MyCurrPoint);

4) виводимо вміст графічного буфера на екран

Form1::pictureBox1->Image = MyBitmap;

Обійшлися досить малою кількістю команд.

Запустимо програму і простежимо, чи достатньо такої малої кількості команд.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 205

Лабораторна робота № 5 (продовження л.р. №4)

Застосування GDI+ та Windows Forms

Мета роботи:

Здобути навички використання бібліотеки GDI+ при створенні програм на базі технології Windows Forms у середовищі розробки Microsoft Visual C++.

Короткі відомості, щодо порядку виконання роботи:

Відповідно до завдання на сьогоднішнє заняття, ми повинні продовжити створення такого програмного застосування, яке виконувало б деякі функції, подібні до функцій графічного редактора.

Навряд чи можливо обмежитись графічними інструментами, які ми вже створили. Створимо ще.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 206

1. Створення графічного інструменту користувача для малювання прямокутників «Rectangle»

1.1. Повернемося у вкладку конструктора форми і для елемента **comboBox1** у властивості **Items** у Редакторі колекції рядків додамо рядок **Rectangle**.

Цей графічний інструмент буде використовувати стандартну функцію-метод малювання **DrawRectangle**.

1.2. Отже у функції **pictureBox1_MouseMove** допишемо рядок виклику нової функції – функції малювання всіх ліній у одному буфері. Додамо її після останнього рядку **if (MyTool == "...") {...}**

Назвемо цю функцію **MyDrawingRectangle**.

```
if (MyTool == "Rectangle") { MyDrawingRectangle(e);}
```

1.3. Зробимо додавання опису нової функції після раніше доданої функції: наприкінці файлу **Form1.h** перед знаком **}:**

тут буде наша функція малювання новим графічним інструментом

```
}:
```

Функція викликатиметься, доки мишка рухається .

1.4. Підготуємо основний оператор опису функції:

```
private: void MyDrawingRectangle(
                System::Windows::Forms::MouseEventArgs^ e
            )
{
    тут описуватимемо оператори функції
}
```

Далі необхідно описати оператори функції.
Які ж оператори застосувати ?

Оскільки прямокутник малюватиметься за допомогою протягування мишки та по двом точкам, то це дуже схоже на реалізацію малювання лінії.

Тому знайдемо вже описану нами функцію малювання лінії:

```
private: void MyDrawingLine(System::Windows::Forms::MouseEventArgs^ e)
{
    MyCurrPoint = Point(e->X, e->Y);
    MyCloneBitmap = MyBitmap->Clone(
        Rectangle( 0, 0, MyBitmap->Width, MyBitmap->Height), MyBitmap-
        >PixelFormat
    );
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 207

```

MyCloneGraph = Graphics::FromImage(MyCloneBitmap);
MyCloneGraph->SmoothingMode =
System::Drawing::Drawing2D::SmoothingMode::AntiAlias;
MyCloneGraph->DrawLine(MyPen, *MyPrevPoint, *MyCurrPoint);
Form1::pictureBox1->Image = MyCloneBitmap;
}

```

Візьмемо вміст цієї функції і скопіюємо у функцію **MyDrawingRectangle**.

Функція малювання прямокутника така:

графічне поле -> **DrawLine(**
MyPen,
координата X верхньої лівої вершини,
координата Y верхньої лівої вершини,
ширина прямокутника,
висота прямокутника
); (це не смайлик, а кінцева дужка параметрів функції)

Причому параметри мають бути додані.

Отже, єдине, що потрібно змінити це рядок **MyCloneGraph->DrawLine(MyPen, *MyPrevPoint, *MyCurrPoint);**

на очевидний рядок:

```

MyCloneGraph->DrawRectangle(
MyPen,
MyPrevPoint->X,
MyPrevPoint->Y,
(MyCurrPoint->X - MyPrevPoint->X),
(MyCurrPoint->Y - MyPrevPoint->Y)
);

```

1.5. Запустимо програму і спробуємо намалювати прямокутник, поки не кінцево, а лише під час руху мишки.

Спробуйте намалювати прямокутник рухаючи мишку ↖ або ↗ або ↙ або ↘.

Якщо все виконано відповідно вказівок, то деякі дії не працюють.

1.6. Спробуємо обміркувати спосіб виправлення цієї ситуації.

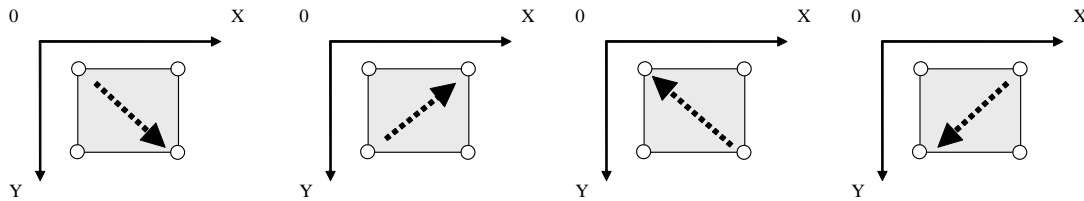
Якщо ми рухаємо мишку ↘, то координати правого верхнього кута прямокутника – початкова точка, завжди менші ніж координати кінцевої точки руху мишки. Тут і ширина і висота прямокутника виходять додатні:

```

(MyCurrPoint->X - MyPrevPoint->X) > 0
(MyCurrPoint->Y - MyPrevPoint->Y) > 0

```

Якщо виконуємо інші рухи: ↖ або ↗ або ↘, то якийсь з параметрів перестав бути додатнім, і точка верхнього лівого кута вже не є початковою точкою руху мишки.



З рисунку видно, що координати **X**, **Y** верхнього лівого кута мають найменше значення, при усіх варіантах руху мишки. Цю проблему вирішимо математичною функцією знаходження мінімуму.

Щодо можливих від'ємних ширини та висоти прямокутника, то тут нам допоможе функція знаходження абсолютного значення числа.

У системі імен **System** є математичні методи **Math**, які можливо викликати. Нам потрібні лише **Min**, **Abs**.

Тож зробимо необхідні зміни.

```
MyCloneGraph->DrawRectangle(
    MyPen,
    Math::Min(MyPrevPoint->X, MyCurrPoint->X),
    Math::Min(MyPrevPoint->Y, MyCurrPoint->Y),
    Math::Abs(MyCurrPoint->X - MyPrevPoint->X),
    Math::Abs(MyCurrPoint->Y - MyPrevPoint->Y)
);
```

1.7. Як і для малювання ліній потрібне завершальне перенесення прямокутника у основний графічний буфер. Тому потрібна функція закінчення малювання **MyEndDrawingRectangle**.

У функцію **pictureBox1_MouseUp** допишемо виклик функції, що завершуватиме процес малювання лінії. Допишемо її після останнього рядка **if (MyTool == "...") {...}**. Назвемо цю функцію **MyEndDrawingRectangle**:

```
if (MyTool == "Rectangle") {MyEndDrawingRectangle(e);}
```

Далі зробимо додавання опису нової функції наприкінці файлу **Form1.h** перед знаком **};**:

тут буде наша функція завершення малювання лінії

```
};
```

Функція викликатиметься одноразово, коли кнопку мишки відпускає користувач.

Підготуємо основний оператор опису функції:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 209

```
private: void MyEndDrawingRectangle(System::Windows::Forms::MouseEventArgs^ e){
    тут описуватимемо оператори функції
}
```

Оператори повністю повторюють оператори функції **MyEndDrawingLine**
Лише замінимо відповідний рядок на:

```
MyGraph->DrawRectangle(
    MyPen,
    Math::Min(MyPrevPoint->X, MyCurrPoint->X),
    Math::Min(MyPrevPoint->Y, MyCurrPoint->Y),
    Math::Abs(MyCurrPoint->X - MyPrevPoint->X),
    Math::Abs(MyCurrPoint->Y - MyPrevPoint->Y)
);
```

1.8. Запустимо програму для перевірки.

Далі, ще простіше.

2. Створення графічного інструменту користувача для малювання кіл та еліпсів «Ellipse»

Все знову повторюється:

2.1. Для елемента **comboBox1** у властивості **Items** у Редакторі колекції рядків додамо рядок **Ellipse**.

2.2. У функцію **pictureBox1_MouseMove** додаємо рядок

```
if (MyTool == "Ellipse") { MyDrawingEllipse(e);}
```

2.3. У функцію **pictureBox1_MouseUp** додаємо рядок

```
if (MyTool == "Ellipse") { MyEndDrawingEllipse(e);}
```

2.4. Після останньої функції у файлі **Form1.h** описуємо нашу **MyDrawingEllipse**.

Вона повністю така ж як і **MyDrawingRectangle**. Її можна скопіювати і лише замінити назву функції. А у описі операторів функції замінити назву методу з **DrawRectangle** на **DrawEllipse**.

2.5. Так само описуємо нашу **MyEndDrawingEllipse**.

Вона повністю така ж як і **MyEndDrawingRectangle**. Її можна скопіювати і лише замінити назву функції. А у описі операторів функції замінити назву методу з **DrawRectangle** на **DrawEllipse**.

3. Створення графічного інструментів користувача для малювання заливаних кіл та еліпсів «FillEllipse» та заливаних прямокутників «FillRectangle»

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 210

3.1. При малюванні заповнених фігур, необхідно використовувати об'єкт типу **Brush**. Він описує спосіб заповнення фігур. Додамо до вже оголошених змінних змінну

```
Brush ^ MyBrush;
```

У функції **pictureBox1_MouseDown** після рядка, де виділяється пам'ять під об'єкт **MyPen**, запишемо рядок виділення пам'ять під об'єкт **MyBrush**:

```
MyBrush = gcnew SolidBrush(MyColor);
```

У функції **pictureBox1_MouseUp** після рядка, де звільняється пам'ять з під об'єкту **MyPen**, запишемо рядок звільнення пам'яті з під об'єкту **MyBrush**:

```
delete MyBrush;
```

А далі знову повторення дій:

3.1. Для **comboBox1** у Редакторі колекції рядків додамо рядки **FillEllipse** та **FillRectangle**.

3.2. У функцію **pictureBox1_MouseMove** додаємо рядки

```
if (MyTool == "FillEllipse") { MyDrawingFillEllipse(e);}
if (MyTool == "FillRectangle") { MyDrawingFillRectangle(e);}
```

3.2. У функцію **pictureBox1_MouseUp** додаємо рядки

```
if (MyTool == "FillEllipse") { MyEndDrawingFillEllipse(e);}
if (MyTool == "FillRectangle") { MyEndDrawingFillRectangle(e);}
```

3.3. Після останньої функції у файлі **Form1.h** описуємо функції **MyDrawingFillEllipse**, **MyDrawingFillRectangle**. Для цього копіюємо повністю функції **MyDrawingRectangle** та **MyDrawingEllipse** разом з їх описами та замінюємо їх назви на **MyDrawingFillRectangle**, **MyDrawingFillEllipse**.

У створених функціях у описі операторів перед функціями-методами **DrawRectangle** та **DrawEllipse** додаємо відповідно методи **FillRectangle** й **FillEllipse**:

```
MyGraph->FillRectangle(
    MyBrush,
    Math::Min(MyPrevPoint->X, MyCurrPoint->X),
    Math::Min(MyPrevPoint->Y, MyCurrPoint->Y),
    Math::Abs(MyCurrPoint->X - MyPrevPoint->X),
    Math::Abs(MyCurrPoint->Y - MyPrevPoint->Y)
);
```

та

```
MyGraph->FillEllipse(
    MyBrush,
    Math::Min(MyPrevPoint->X, MyCurrPoint->X),
    Math::Min(MyPrevPoint->Y, MyCurrPoint->Y),
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 211

```
Math::Abs(MyCurrPoint->X - MyPrevPoint->X),
Math::Abs(MyCurrPoint->Y - MyPrevPoint->Y)
);
```

Чому ми **ДОДАЛИ** методи а не замінили?

Так ми зможемо малювати залитий прямокутник чи еліпс **З КОНТУРОМ**.

Зверніть увагу, що для малювання заповнених фігур використовується об'єкт стилю малювання **MyBrush** типу **Brush**.

3.4. Аналогічно виконуємо закінчення малювання. Копіюємо описи функцій **MyEndDrawingRectangle** та **MyEndDrawingEllipse** та перейменовуємо їх на **MyEndDrawingFillRectangle** та **MyEndDrawingFillEllipse**.

А у описі операторів функції перед методами **DrawRectangle** та **DrawEllipse** додаємо відповідно методи **FillRectangle** й **FillEllipse**, точно такі ж, як і у пункті 3.3.

3.5. Запускаємо, перевіряємо.

Намалювавши декілька прямокутників, бачимо, що їх контур не різниця за кольором з тілом прямокутника. Це тому, що ми задавали для **MyBrush** той самий колір **MyColor**, що і для **Pen**.

Тепер ми можемо навіть дещо намалювати.

Проте, як зберегти малюнок ?

4. Створення інструментарію збереження зображення

4.1. Перейдемо до **Form1.h[конструктор]** і у меню форми у категорії **File** створимо підкатегорію **Save**.

4.2. Витягнемо на форму елемент **saveFileDialog** – вікно вибору місця збереження та імені файлу, що зберігатиметься.

Задамо для нього властивість **Filter** у вигляді спеціального відформатованого рядка

```
JPEG | *.jpeg | PNG | *.png | GIF | *.gif | TIFF | *.tiff | BMP | *.bmp
```

По парам у рядок розташовуються **вираз для відображення при виборі | реальне розширення файлу**. Це дасть змогу користувачу обирати тип файлу (розширення файлу), який зберігатиметься. Визначено лише ті типи, які можливо зберегти засобами **GDI+**.

4.3. Подвійним кліком на підкатегорії **Save** зайдемо у функцію-відклик (**saveToolStripMenuItem_Click**). Задамо код для цієї функції.

- показ модального вікна **saveFileDialog**

```
saveFileDialog1->ShowDialog();
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 212

- передбачення відміни збереження (вихід з функції), якщо ім'я файлу не обрано

```
if (saveFileDialog1->FileName=="")return;
```

- якщо обрано перше розширення з зазначених у **Filter**, то зберігаємо з відповідним розширенням

```
if (saveFileDialog1->FilterIndex==1)
    MyBitmap->Save(
        saveFileDialog1->FileName,Imaging::ImageFormat::Jpeg
    );
```

- якщо обрано інше розширення з зазначених у **Filter**, то зберігаємо з цим розширенням

```
if (saveFileDialog1->FilterIndex==2)
    MyBitmap->Save(
        saveFileDialog1->FileName,Imaging::ImageFormat::Png
    );
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 213

```

if (saveFileDialog1->FilterIndex==3)
    MyBitmap->Save(
        saveFileDialog1->FileName, Imaging::ImageFormat::Gif
    );
if (saveFileDialog1->FilterIndex==4)
    MyBitmap->Save(
        saveFileDialog1->FileName, Imaging::ImageFormat::Tiff
    );
if (saveFileDialog1->FilterIndex==5)
    MyBitmap->Save(
        saveFileDialog1->FileName, Imaging::ImageFormat::Bmp
    );

```

Як видно з команд, **FilterIndex** – це поле об'єкту **saveFileDialog1**, яким дозволяє визначити обране розширення за порядковим номером у рядку у властивості **Filter**.

Обране розширення задавалося спеціальними системними змінними **Imaging::ImageFormat::розширення**.

Безпосередньо збереження файлу відбувається за допомогою функції-методу **Save** графічного об'єкту **MyBitmap**.

4.4. Запустимо, спробуємо зберегти малюнок у різних форматах, спробуємо відкрити файл за допомогою інших програм.

Ми навчилися зберігати файли малюнків. Тепер спробуємо створити засоби для відкриття файлів зображень.

5. Створення інструментарію відкриття файлу зображення

5.1. Перейдемо до **Form1.h[конструктор]** і у меню форми у категорії **File** створимо підкатегорію **Open**.

5.2. Витягнемо на форму елемент **openFileDialog** – вікно вибору шляху до файлу та імені файлу, що відкриватиметься.

Задамо для нього властивість **Filter** у вигляді спеціального відформатованого рядка

Файли зображень | *.jpeg;*.png;*.tiff;*.bmp

Це дасть змогу користувачу обирати тип файлів, який відобразатимуться для вибору конкретного файлу. Визначено лише ті типи, які можливо зберегти засобами **GDI+**.

5.3. Подвійним кліком на підкатегорії **Open** зайдемо у функцію-відклик (**openToolStripMenuItem_Click**). Задамо код для цієї функції.

- відкриває **openFileDialog**

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 214

openFileDialog1->ShowDialog();

- якщо не обрано жодного файлу, вихід з функції

if(openFileDialog1->FileName=="") return;

- звільнення пам'яті від попереднього графічного буфера

MyBitmap->~Bitmap();

- створення нового графічного буфера з файлу, шлях якого обрано у **openFileDialog**

MyBitmap = gcnew Bitmap(openFileDialog1->FileName,true);

- видалення попереднього графічного об'єкту області малювання

MyGraph->~Graphics();

- створення нової області малювання для нового графічного буфера

MyGraph = Graphics::FromImage(MyBitmap);

- завдання режиму згладжування при відображенні

**MyGraph->SmoothingMode =
System::Drawing::Drawing2D::SmoothingMode::AntiAlias;**

- завдання однакової з графічним буфером ширини та висоти області **pictureBox1**, у якому відображатимемо графічний буфер

**pictureBox1->Width = MyBitmap->Width;
pictureBox1->Height = MyBitmap->Height;**

- відображаємо графічний буфер у **pictureBox1**

Form1::pictureBox1->Image = MyBitmap;

6. Розділення кольорів, що заповнюють область фігур, та кольорів ліній та контурів

6.1. Оголосимо поряд з вже існуючою **MyColor** змінну **MyFillColor**

Color MyColor, MyFillColor;

6.2. Перейдемо до **Form1.h[конструктор]** і у меню форми у категорії **Color** створимо дві підкатегорії **For Lines** та **For Fill**

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 215

6.3. Подвійним кліком на категорії **Color** зайдемо у її функцію-відклик (**colorToolStripMenuItem_Click**) та **виріжмо** рядки коду з неї. Тепер вони нам потрібні при окремому задаванні кольору ліній.

6.4. Подвійним кліком на підкатегорії **For Lines** зайдемо у функцію-відклик (**forLinesToolStripMenuItem_Click**) та вставимо вирізані тільки-но рядки - це код виклик діалогового вікна вибору кольору та присвоєння обраного кольору змінній кольору ліній:

```
colorDialog1->ShowDialog();
MyColor = colorDialog1->Color;
```

6.5. Подвійним кліком на підкатегорії **For Fill** зайдемо у функцію-відклик (**forFillToolStripMenuItem_Click**) та вставимо подібний код, але вже для змінної кольору заповнення:

```
colorDialog1->ShowDialog();
MyFillColor = colorDialog1->Color;
```

6.6. На останок знайдемо у кодї (у функції **pictureBox1_MouseDown**) рядок **MyBrush = gcnw SolidBrush(MyColor);** та замінимо тут змінну **MyColor** на **MyFillColor**.

6.7. Перевіримо дієвість змін у режимі виконання програми. Спробуємо задати різні кольори для ліній та заповнення і малювати заповнені фігури з контурами.

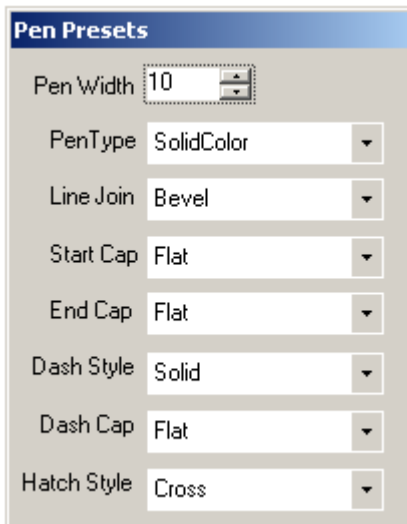
Далі деякі ускладнені можливості малювання ліній та фігур – малювання зі стилями.

7. Створення форм, для завдання значень стилів ліній та стилів заповнення фігур

7.1. Перейдемо до дерева проекту у **Оглядачі рішення**. ПКМ на папочці, де є **Form1.h** і обираємо команду **Додати – Створити елемент – UI – Форма**. Створимо два елементи: **MyForm1** та **MyForm2**.

7.2. У **Конструкторі** створених форм викликати властивості і задати **ControlBox** як **False**. Це потрібно для недопущення закриття форм, що призведе до видалення з пам'яті об'єктів форм. Також можна задати підпис форм – властивість **Text** як **Pen Presets** (для **MyForm1**) та **Brush Presets** (для **MyForm2**).

7.3. У **Конструкторі** створити елементи форм за зразком, скориставшись елементами **numericUpDown**, **lable** та **comboBox**



7.4. Задамо властивості елементів для першої форми **MyForm1**, яку назвали **PenPresets**

- для елемента **numericUpDown1** властивість **Border Style** як **None**
- для елемента **numericUpDown1** властивість **Value** як **10**
- для всіх елементів **comboBox** властивість **Flat Style** як **Flat**
- для **comboBox1**, за яким обиратиметься тип ліній, властивість **Items** як список назв типів

HatchFill
Solid

- для **comboBox2**, за яким обиратиметься тип з'єднання ламаних ліній у фігурах

Bevel
Miter
MiterClipped
Round

- для **comboBox3**, за яким обиратиметься форма початку ліній, та для **comboBox4**, за яким обиратиметься форма кінця ліній

AnchorMask
ArrowAnchor
Custom
DiamondAnchor
Flat
NoAnchor
Round
RoundAnchor
SquareAnchor
Triangle

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 217

- для **comboBox5**, за яким обиратиметься штриховий стиль лінії

Custom
Dash
DashDot
DashDotDot
Dot
Solid

- для **comboBox6**, за яким обиратиметься вигляд кінців штрихів

Flat
Round
Triangle

- для **comboBox7**, за яким обиратиметься тип графічного заповнення ліній

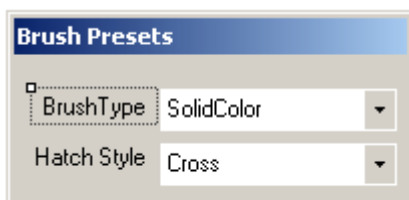
BackwardDiagonal
Cross
DarkDownwardDiagonal
DarkHorizontal
DarkUpwardDiagonal
DarkUpwardDiagonal
DarkVertical
DashedDownwardDiagonal
DashedHorizontal
DashedUpwardDiagonal
DashedVertical
DiagonalBrick
DiagonalCross
Divot
DottedDiamond
DottedGrid
ForwardDiagonal
Horizontal
HorizontalBrick
LargeCheckerBoard
LargeConfetti
LargeGrid
LightDownwardDiagonal
LightUpwardDiagonal
LightVertical
Max
Min
NarrowHorizontal
NarrowVertical
OutlinedDiamond
Percent05
Percent10
Percent20

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 218

Percent25
Percent30
Percent40
Percent50
Percent60
Percent70
Percent80
Percent90
Plaid
Shingle
SmallCheckerBoard
SmallConfetti
SmallGrid
SolidDiamond
Sphere
Trellis
Vertical
Wave
Weave
WideDownwardDiagonal
WideUpwardDiagonal
ZigZag

- початкові значення для всіх **comboBox** задати через властивість **Text** як показано на рисунку вище

7.5. Відредагуємо форму **MyForm2**, яку назвали **Brush Presets**, скопіювавши **comboBox1** та **comboBox7** у неї з **MyForm1**, та замінивши текст у підписах елементів **Label**.



Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 219

7.6. На початку файлу головної форми Form1.h додати рядки

```
#include "MyForm1.h"
#include "MyForm2.h"
```

для того, щоб мати доступ до полів та методів додаткових форм.

8. Створення функцій виклику відображення додаткових форм

8.1. Для відображення форм класів **MyForm1** та **MyForm2**, в яких будуть елементи завдання параметрів, потрібно створити змінні-показчики, які потім вказуватимуть на об'єкти-форми. Додамо змінні

```
MyForm1^ PenParamForm;
MyForm2^ BrushParamForm;
```

8.2. Створимо об'єкти форми класів **MyForm1** та **MyForm2**, та передамо показчики на них у відповідні змінні. Додамо ці дії у функцію, що викликається при завантаженні форми – **Form1_Load**.

```
PenParamForm = gcnw MyForm1();
BrushParamForm = gcnw MyForm2();
```

8.3. Повернемося до **Конструктора** нашої головної форми **Form1.h[Конструктор]**. Додамо у **menuStrip1** (меню форми) дві категорії: **Pen** і **Brush**.

8.4. Подвійно клікнемо на **Pen**. У функції відклику **penToolStripMenuItem_Click** прописуємо дії

```
if(!PenParamForm->Visible) PenParamForm->Show();
else if(PenParamForm->Visible) PenParamForm->Hide();
```

За цими діями форма **PenParamForm** з заголовком **Pen Settings** буде відображатись, якщо вона прихована, та приховуватись, якщо форма відображена. Тобто, відображатись і приховуватись, при повторних натисненнях на кнопку меню **Pen**.

8.5. Запустимо і перевіримо функціональність.

8.6. Подвійно клікнемо на **Brush**. У функції відклику **brushToolStripMenuItem_Click** прописуємо дії

```
if(!BrushParamForm->Visible) BrushParamForm->Show();
else if(BrushParamForm->Visible) BrushParamForm->Hide();
```

Так само форма з заголовком **Brush Settings** відображатись і приховуватись, при повторних натисненнях на кнопку меню **Brush**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 220

9. Зчитування значень параметрів інструменту Pen з додаткової форми

Об'єкт класу **Pen** відповідає за стиль відображення ліній.

Зчитуватимемо дані з додаткової форми параметрів малювання ліній. Цю форму правильно описати так:

- клас: **MyForm1**
- заголовок: **PenSettings**
- вказівник на об'єкт: **PenParamForm**

9.1. Для застосування скорочених імен деяких функцій використовуємо додатковий простір імен. Додаємо рядок до інших доданих просторів імен у файлі **Form1.h**:

```
using namespace System::Drawing::Drawing2D;
```

9.2. У функції **pictureBox1_MouseDown**, де створюється новий об'єкт **MyPen** малювання ліній, закоментуємо рядок створення цього об'єкту і додамо виклик власної функції **SetMyPen()** по створенню об'єкту **MyPen**.

```
//MyPen = gcnew Pen(MyColor, (float) MyPenWidth);  
SetMyPen();
```

9.3. У кінці файлу **Form1.h**, у кінці оголошення класу **Form1.h** перед символами

```
};  
}
```

додаємо нами написану функцію:

```
private: void SetMyPen()  
{  
  
}
```

між { } у функцію прописуємо команди

- зчитуємо з форми з елемента **numericUpDown1** значення товщини ліній і запишемо його у змінну **MyPenWidth**

```
MyPenWidth = PenParamForm->numericUpDown1->Value;
```

- зчитуємо дані з форми з елемента **comboBox1** дані про обраний тип ліній у проміжну змінну-рядок **Ch**

```
String ^ Ch;  
Ch = PenParamForm->comboBox1->Text;
```

- якщо обрано тип **Суцільна лінія**, то створюємо об'єкт класу **Pen**, із заданим раніше кольором та товщиною

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 221

```

if (Ch == "SolidColor")
{
    MyPen = gcnew Pen(MyColor, (float) MyPenWidth); }

```

- якщо обрано тип **Заштрихована лінія**, то зчитуємо стиль заповнення у **Ch** – проміжну змінну-рядок

```

else if (Ch == "HatchFill")
{
    Ch = PenParamForm->comboBox7->Text;
}

```

- залежно від обраного типу заповнення записуємо відповідну бібліотечну константу стилю заповнення у **St** – проміжну змінну-стиль

```

HatchStyle St; // оголошення змінної стилю заповнення
if (Ch == "BackwardDiagonal")
    St= HatchStyle::BackwardDiagonal;
else if (Ch == "Cross")
    St=HatchStyle::Cross;
else if (Ch == "DarkDownwardDiagonal")
    St=HatchStyle::DarkDownwardDiagonal;
else if (Ch == "DarkHorizontal")
    St=HatchStyle::DarkHorizontal;
else if (Ch == "DarkUpwardDiagonal")
    St=HatchStyle::DarkUpwardDiagonal;
else if (Ch == "DarkVertical")
    St=HatchStyle::DarkVertical;
else if (Ch == "DashedDownwardDiagonal")
    St=HatchStyle::DashedDownwardDiagonal;
else if (Ch == "DashedHorizontal")
    St=HatchStyle::DashedHorizontal;
else if (Ch == "DashedUpwardDiagonal")
    St=HatchStyle::DashedUpwardDiagonal;
else if (Ch == "DashedVertical")
    St=HatchStyle::DashedVertical;
else if (Ch == "DiagonalBrick")
    St=HatchStyle::DiagonalBrick;
else if (Ch == "DiagonalCross")
    St=HatchStyle::DiagonalCross;
else if (Ch == "Divot")
    St=HatchStyle::Divot;
else if (Ch == "DottedDiamond")
    St=HatchStyle::DottedDiamond;
else if (Ch == "DottedGrid")
    St=HatchStyle::DottedGrid;
else if (Ch == "ForwardDiagonal")
    St=HatchStyle::ForwardDiagonal;
else if (Ch == "Horizontal")
    St=HatchStyle::Horizontal;
else if (Ch == "HorizontalBrick")
    St=HatchStyle::HorizontalBrick;
else if (Ch == "LargeCheckerBoard")

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 222

```

St=HatchStyle::LargeCheckerBoard;
else if (Ch == "LargeConfetti")
    St=HatchStyle::LargeConfetti;
else if (Ch == "LargeGrid")
    St=HatchStyle::LargeGrid;
else if (Ch == "LightDownwardDiagonal")
    St=HatchStyle::LightDownwardDiagonal;
else if (Ch == "LightUpwardDiagonal")
    St=HatchStyle::LightUpwardDiagonal;
else if (Ch == "LightVertical")
    St=HatchStyle::LightVertical;
else if (Ch == "Max")
    St=HatchStyle::Max;
else if (Ch == "Min")
    St=HatchStyle::Min;
else if (Ch == "NarrowHorizontal")
    St=HatchStyle::NarrowHorizontal;
else if (Ch == "NarrowVertical")
    St=HatchStyle::NarrowVertical;
else if (Ch == "OutlinedDiamond")
    St=HatchStyle::OutlinedDiamond;
else if (Ch == "Percent05")
    St=HatchStyle::Percent05;
else if (Ch == "Percent10")
    St=HatchStyle::Percent10;
else if (Ch == "Percent20")
    St=HatchStyle::Percent20;
else if (Ch == "Percent25")
    St=HatchStyle::Percent25;
else if (Ch == "Percent30")
    St=HatchStyle::Percent30;
else if (Ch == "Percent40")
    St=HatchStyle::Percent40;
else if (Ch == "Percent50")
    St=HatchStyle::Percent50;
else if (Ch == "Percent60")
    St=HatchStyle::Percent60;
else if (Ch == "Percent70")
    St=HatchStyle::Percent70;
else if (Ch == "Percent80")
    St=HatchStyle::Percent80;
else if (Ch == "Percent90")
    St=HatchStyle::Percent90;
else if (Ch == "Plaid")
    St=HatchStyle::Plaid;
else if (Ch == "Shingle")
    St=HatchStyle::Shingle;
else if (Ch == "SmallCheckerBoard")
    St=HatchStyle::SmallCheckerBoard;
else if (Ch == "SmallConfetti")

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 223

```

St=HatchStyle::SmallConfetti;
else if (Ch == "SmallGrid")
    St=HatchStyle::SmallGrid;
else if (Ch == "SolidDiamond")
    St=HatchStyle::SolidDiamond;
else if (Ch == "Sphere")
    St=HatchStyle::Sphere;
else if (Ch == "Trellis")
    St=HatchStyle::Trellis;
else if (Ch == "Vertical")
    St=HatchStyle::Vertical;
else if (Ch == "Wave")
    St=HatchStyle::Wave;
else if (Ch == "Weave")
    St=HatchStyle::Weave;
else if (Ch == "WideDownwardDiagonal")
    St=HatchStyle::WideDownwardDiagonal;
else if (Ch == "WideUpwardDiagonal")
    St=HatchStyle::WideUpwardDiagonal;
else if (Ch == "ZigZag")
    St=HatchStyle::ZigZag;

```

- створюємо **MyBrush** – об’єкт заповнення відповідно до стилю заповнення, що зберігається у змінній **St**, і на базі цього об’єкту заповнення створюємо **MyPen** – об’єкт малювання заповнених ліній

```

MyBrush = gcnw HatchBrush(St,MyColor,MyFillColor);
MyPen = gcnw Pen( MyBrush, (float) MyPenWidth);

```

- закриваємо інструкцію **else if**

```

}

```

(запуск, перевірка)

- зчитуємо дані про стиль поєднання ламаних ліній у змінну-рядок **Ch**

```

Ch = PenParamForm->comboBox2->Text;

```

- відповідно до обраного стилю поєднання, зберігаємо відповідний стиль у змінну **St2**, яка має бути типу **LineJoin** (бібліотечний тип)

```

LineJoin St2;
if (Ch == "Bevel") St2= LineJoin::Bevel;
if (Ch == "Miter") St2= LineJoin::Miter;
if (Ch == "MiterClipped") St2= LineJoin::MiterClipped;
if (Ch == "Round") St2= LineJoin::Round;

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 224

(запуск, перевірка)

- задаємо обраний стиль у параметр **LineJoin** об'єкту **MyPen**

MyPen->LineJoin = St2;

- зчитуємо дані про обраний стиль кінця ліній у змінну-рядок **Ch**, присвоюємо відповідний стиль змінній стилю **St3**, задаємо обраний стиль у параметр **StartCap**, об'єкту **MyPen**

```
Ch = PenParamForm->comboBox3->Text;
LineCap St3;
if (Ch == "AnchorMask") St3= LineCap::AnchorMask;
if (Ch == "ArrowAnchor") St3= LineCap::ArrowAnchor;
if (Ch == "Custom") St3= LineCap::Custom;
if (Ch == "DiamondAnchor") St3= LineCap::DiamondAnchor;
if (Ch == "Flat") St3= LineCap::Flat;
if (Ch == "NoAnchor") St3= LineCap::NoAnchor;
if (Ch == "Round") St3= LineCap::Round;
if (Ch == "RoundAnchor") St3= LineCap::RoundAnchor;
if (Ch == "SquareAnchor") St3= LineCap::SquareAnchor;
if (Ch == "Triangle") St3= LineCap::Triangle;
MyPen->StartCap = St3;
```

- ті ж дії для іншого кінця лінії

```
Ch = PenParamForm->comboBox4->Text;
LineCap St4;
if (Ch == "AnchorMask") St4= LineCap::AnchorMask;
if (Ch == "ArrowAnchor") St4= LineCap::ArrowAnchor;
if (Ch == "Custom") St4= LineCap::Custom;
if (Ch == "DiamondAnchor") St4= LineCap::DiamondAnchor;
if (Ch == "Flat") St4= LineCap::Flat;
if (Ch == "NoAnchor") St4= LineCap::NoAnchor;
if (Ch == "Round") St4= LineCap::Round;
if (Ch == "RoundAnchor") St4= LineCap::RoundAnchor;
if (Ch == "SquareAnchor") St4= LineCap::SquareAnchor;
if (Ch == "Triangle") St4= LineCap::Triangle;
MyPen->EndCap = St4;
```

(запуск, перевірка)

- зчитуємо дані про обраний стиль малювання лінії штрихами у змінну-рядок **Ch**, присвоюємо відповідний стиль змінній стилю **St5**, задаємо обраний стиль у параметр **DashStyle**, об'єкту **MyPen**

```
Ch = PenParamForm->comboBox5->Text;
DashStyle St5;
if (Ch == "Custom") St5= DashStyle::Custom;
if (Ch == "Dash") St5= DashStyle::Dash;
if (Ch == "DashDot") St5= DashStyle::DashDot;
```


Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк __ / 225

```
if (Ch == "DashDotDot") St5= DashStyle::DashDotDot;
if (Ch == "Dot") St5= DashStyle::Dot;
if (Ch == "Solid") St5= DashStyle::Solid;
MyPen->DashStyle = St5;
```

(запуск, перевірка)

- зчитуємо дані про обраний стиль малювання штрихів при малюванні лінії штрихами, зчитуємо у змінну-рядок **Ch**, присвоюємо відповідний стиль змінній стилю **St6**, задаємо обраний стиль у параметр **DashCap** , об'єкту **MyPen**

```
Ch = PenParamForm->comboBox6->Text;
DashCap St6;
if (Ch == "Flat") St6= DashCap::Flat;
if (Ch == "Round") St6= DashCap::Round;
if (Ch == "Triangle") St6= DashCap::Triangle;
MyPen->DashCap = St6;
```

Запуск, перевірка: Спробуємо намалювати лінії різної товщини, стилів.

10. Зчитування значень параметрів інструменту Brush з додаткової форми

Об'єкт класу **Brush** відповідає за заповнення (заливку) суцільних фігур чи ліній.

10.1. Зчитуватимемо дані з додаткової форми параметрів заповнення фігур. Цю форму правильно описати так:

- клас: **MyForm2**
- заголовок: **BrushSettings**
- вказівник на об'єкт: **BrushParamForm**

10.2. У функції **pictureBox1_MouseDown**, де створюється новий об'єкт **MyBrush** малювання ліній, закоментуємо рядок створення цього об'єкту і додамо виклик власної функції **SetMyBrush()** по створенню об'єкту **MyPen**.

```
//MyBrush = gcnw SolidBrush(MyFillColor);
SetMyBrush();
```

10.3. У кінці файлу **Form1.h**, у кінці оголошення класу **Form1.h** перед символами

```
};
}
```

додаємо нами написану функцію:

```
private: void SetMyBrush()
{
}
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 226

між { } у функцію прописуємо команди

команди аналогічні тим, що додавали при написанні функції **SetMyPen**

```

String ^ Ch;
Ch = BrushParamForm->comboBox1->Text;
if (Ch == "SolidColor")
{
    MyBrush = gcnew SolidBrush(MyFillColor);
}
else
// same as BrushParamForm->comboBox1->Text == "HatchFill"
{
    Ch = BrushParamForm->comboBox7->Text;
    HatchStyle St;
if (Ch == "BackwardDiagonal") St=HatchStyle::BackwardDiagonal;
else if (Ch == "Cross") St=HatchStyle::Cross;
else if (Ch == "DarkDownwardDiagonal") St=HatchStyle::DarkDownwardDiagonal;
else if (Ch == "DarkHorizontal") St=HatchStyle::DarkHorizontal;
else if (Ch == "DarkUpwardDiagonal") St=HatchStyle::DarkUpwardDiagonal;
else if (Ch == "DarkVertical") St=HatchStyle::DarkVertical;
else if (Ch == "DashedDownwardDiagonal") St=HatchStyle::DashedDownwardDiagonal;
else if (Ch == "DashedHorizontal") St=HatchStyle::DashedHorizontal;
else if (Ch == "DashedUpwardDiagonal") St=HatchStyle::DashedUpwardDiagonal;
else if (Ch == "DashedVertical") St=HatchStyle::DashedVertical;
else if (Ch == "DiagonalBrick") St=HatchStyle::DiagonalBrick;
else if (Ch == "DiagonalCross") St=HatchStyle::DiagonalCross;
else if (Ch == "Divot") St=HatchStyle::Divot;
else if (Ch == "DottedDiamond") St=HatchStyle::DottedDiamond;
else if (Ch == "DottedGrid") St=HatchStyle::DottedGrid;
else if (Ch == "ForwardDiagonal") St=HatchStyle::ForwardDiagonal;
else if (Ch == "Horizontal") St=HatchStyle::Horizontal;
else if (Ch == "HorizontalBrick") St=HatchStyle::HorizontalBrick;
else if (Ch == "LargeCheckerBoard") St=HatchStyle::LargeCheckerBoard;
else if (Ch == "LargeConfetti") St=HatchStyle::LargeConfetti;
else if (Ch == "LargeGrid") St=HatchStyle::LargeGrid;
else if (Ch == "LightDownwardDiagonal") St=HatchStyle::LightDownwardDiagonal;
else if (Ch == "LightUpwardDiagonal") St=HatchStyle::LightUpwardDiagonal;
else if (Ch == "LightVertical") St=HatchStyle::LightVertical;
else if (Ch == "Max") St=HatchStyle::Max;
else if (Ch == "Min") St=HatchStyle::Min;
else if (Ch == "NarrowHorizontal") St=HatchStyle::NarrowHorizontal;
else if (Ch == "NarrowVertical") St=HatchStyle::NarrowVertical;
else if (Ch == "OutlinedDiamond") St=HatchStyle::OutlinedDiamond;
else if (Ch == "Percent05") St=HatchStyle::Percent05;
else if (Ch == "Percent10") St=HatchStyle::Percent10;
else if (Ch == "Percent20") St=HatchStyle::Percent20;
else if (Ch == "Percent25") St=HatchStyle::Percent25;
else if (Ch == "Percent30") St=HatchStyle::Percent30;
else if (Ch == "Percent40") St=HatchStyle::Percent40;

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 227

```

else if (Ch == "Percent50") St=HatchStyle::Percent50;
else if (Ch == "Percent60") St=HatchStyle::Percent60;
else if (Ch == "Percent70") St=HatchStyle::Percent70;
else if (Ch == "Percent80") St=HatchStyle::Percent80;
else if (Ch == "Percent90") St=HatchStyle::Percent90;
else if (Ch == "Plaid") St=HatchStyle::Plaid;
else if (Ch == "Shingle") St=HatchStyle::Shingle;
else if (Ch == "SmallCheckerBoard") St=HatchStyle::SmallCheckerBoard;
else if (Ch == "SmallConfetti") St=HatchStyle::SmallConfetti;
else if (Ch == "SmallGrid") St=HatchStyle::SmallGrid;
else if (Ch == "SolidDiamond") St=HatchStyle::SolidDiamond;
else if (Ch == "Sphere") St=HatchStyle::Sphere;
else if (Ch == "Trellis") St=HatchStyle::Trellis;
else if (Ch == "Vertical") St=HatchStyle::Vertical;
else if (Ch == "Wave") St=HatchStyle::Wave;
else if (Ch == "Weave") St=HatchStyle::Weave;
else if (Ch == "WideDownwardDiagonal") St=HatchStyle::WideDownwardDiagonal;
else if (Ch == "WideUpwardDiagonal") St=HatchStyle::WideUpwardDiagonal;
else if (Ch == "ZigZag") St=HatchStyle::ZigZag;

```

MyBrush = gcnew HatchBrush(St,MyColor,MyFillColor);

Перевірити роботу програми. Спробувати намалювати заповнені фігури з різними стилями ліній контурів, та різними стилями заповнення.

У звіті представити зображення різних стилів малювання ліній та заповнених фігур.

Перелік рекомендованої літератури

Основна література

1. Томас Кормен, Чарльз Лейзерсон, Рональд Рівест, Кліффорд Стайн Вступ до алгоритмів – К.: К.І.С., 2019. – 1288с. (див. посилання нижче)
2. Vjarne Stroustrup The C++ Programming Language (4th Edition), 2013 – 1366 p.
3. Грицюк Ю.І., Рак Т.Є. Програмування мовою С++ : навчальний посібник. – Львів : Вид-во Львівського ДУ БЖД, 2011. – 292 с.
4. Грицюк Ю.І., Рак Т.Є. Об'єктно-орієнтоване програмування мовою С++ : навчальний посібник. – Львів : Вид-во Львівського ДУ БЖД, 2011. – 404 с.
5. Adam Sawicki C++/ CLI Tutorial, 2011. - 30 p.
6. Андрій Будаї Дизайн-патерни, 2016. – 90 с.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-20.10- 05.02/2/152.00.1/Б/ОК13- 2022
	Екземпляр № 1	Арк. __ / 228

Допоміжна література

- інформаційні ресурси

1. <http://programming.in.ua/> - сторінка україномовних матеріалів з програмування для початківців
2. https://www.w3schools.com/CPP/cpp_intro.asp - електронний підручник мови програмування C++ від спільноти W3C (Консорціуму Всесвітнього Павутиння) з можливістю представлення українською мовою (інструмент головного меню сайту)
3. www.microsoft.com – Internet портал виробника MS Visual Studio
4. <https://docs.microsoft.com> - сервер документації Microsoft з безлічі питань сфери програмування з можливістю представлення українською мовою (інструмент головного меню сайту)
5. <https://docs.microsoft.com/en-us/cpp/?view=msvc-160>
- портал виробника Microsoft для розробників програмного забезпечення на мові C++
6. <https://docs.microsoft.com/en-US/visualstudio/get-started/visual-studio-ide?view=vs-2019>
- портал виробника Microsoft для розробників програмного забезпечення у редакторі програм Visual Studio 2019
7. www.wikipedia.org – довідкова енциклопедична інформація про мову програмування C/C++
8. <https://play.google.com/books/reader?id=HkzYDwAAQBAJ&pg=GBS.PP1&hl=uk> - перші 229 сторінок книги «Томас Кормен, Чарльз Лейзерсон, Рональд Рівест, Кліффорд Стайн Вступ до алгоритмів»
9. <https://www.youtube.com/c/VirtuAka/featured> - україномовний youtube-канал з питань програмування «Віртуальна Академія».