

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 1

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ
для виконання лабораторних робіт
з навчальної дисципліни
«КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ ТА ПРОГРАМУВАННЯ»
для здобувачів вищої освіти освітнього ступеня «бакалавр»
спеціальності 141 «Електроенергетика, електротехніка та електромеханіка»
освітня програма: «Комп'ютеризоване управління енергетичними
системами»,
факультет комп'ютерно-інтегрованих технологій, мехатроніки та
робототехніки
кафедра робототехніки, електроенергетики та автоматизації
імені проф. Б.Б.Самотокіна

Схвалено на засіданні кафедри
робототехніки, електроенергетики та
автоматизації ім. проф. Б.Б.Самотокіна
27 серпня 2024 р.
протокол № 6

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

Житомир
2024 – 2025 н.р.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 2

Лабораторна робота 1. Ознайомлення з середовищем розробки Microsoft Visual Studio. Базові операції з даними засобами C++.

Мета роботи: Здобути навички використання середовища розробки Visual Studio. Здобути навички написання, компіляції, виправлення помилок та відлагодження програм. Вивчити команди мови C++ для виконання базових операцій над змінними числового та логічного типу.

1. Створення нового проекту

File --> New --> Project

У полі **All Languages** обрати C++.

Натиснути **Next**.

(показати власний скріншот з вікном шаблону тексту програми)

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

Скористайтесь запропонованим прикладом та наповніть головний файл новим текстом програми:

```
#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()

{ // відкриваюча дужка основної програми
  // далі інструкції користувача
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 3

```

// оголошення змінних

int a, b;
float operand_a, operand_b, result_c;

// елементарні дії над змінними

a = b + a;

a = ++b;

a = b++;

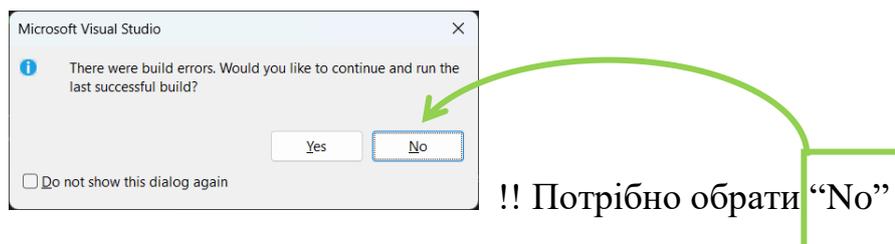
result_c = operand_a / operand_b;

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

```

Для того щоб відкомпілювати проект потрібно вибрати пункт меню **Build --> Build Solution**

Якщо, все ж таки, в програмному кодї є помилки тоді



У вікні **Output** (яке за замовчуванням знаходиться в нижній частині середовища розробки) буде відображатися інформація про хід процесу компіляції.

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)

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 4

(показати власний скріншот з повідомленнями вікна Output)

(показати власний скріншот з повідомленнями вікна Errors)

Подвійне натискання **на повідомленні про помилку** у закладці **вікна Errors** призводить до відкриття файлу з помилкою та встановленні курсору у рядок з помилкою.

(показати власний скріншот з вказанням місця помилки)

Спробуйте виправити помилку.

(поясніть, в чому була помилка)

(показати власний скріншот з виправленим текстом програми)

Після виправлення помилок та вдалої компіляції можна запустити створений файл на виконання, для чого потрібно натиснути **Ctrl + F5**.

Якщо в програмному коді відсутні помилки, тоді в результаті компіляції отримуємо повідомлення:

```
Compiling...
lab1.cpp
Linking...

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

(показати власний скріншот з повідомленнями вікна Output)

3. Відлагодження (Debug) програмного коду

Відлагодження – етап, який присутній при написанні будь-якої досить складної програми, та який є необхідним у випадку, якщо в результаті спроби запуску програми виникають сповіщення про помилки, виконання програми

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 5

було отримано неочікуваний результат або поведінка програми цілком є незрозумілою.

При відлагодженні програміст локалізує місце знаходження помилок у програмі шляхом аналізу змінних та порядку виконання команд.

1) Встановити точку зупинки (Breakpoint) в першому рядку з якого почнемо виконувати відлагодження:

- встановити в першому рядку програми курсор
- клавішею F9 встановити або видалити точку зупинки
- ліворуч від рядка з встановленою точкою зупинки з'являється зафарбоване червоним кольором коло

(показати власний скріншот з точкою зупинки)

2) Для того, щоб виконувати відлагодження потрібно запустити програму в режимі налагодження, це можна зробити за допомогою пункту меню **Debug --> Start Debugging** або **F5**.

(показати власний скріншот, як змінилась точка зупинки)

В цьому 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:

3) Виконуйте команди по чергово, натискаючи **F10**

При досягненні програмою рядка з встановленою точкою зупинки виконання програми тимчасово припиняється (програма знаходиться в режимі призупинення виконання). Рядок коду,

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 6

який виконується в даний момент відмічено жовтою стрілкою ліворуч від рядка.

(показати власний скріншот, як змінилась точка зупинки)

- 4) Щоб дізнатися поточне значення змінної наведіть на неї курсор, ім'я змінної та її значення, відобразиться у вікні спливаючі підказки.

(показати власний скріншот рядку з підказкою)

- 5) Для того, щоб переглянути значення змінної у закладці **вікна Watch** (знизу, біля вікон Errors та Output) можна виділити змінну і перетягнути її в дане вікно.

(показати власний скріншот вікна Watch)

- 6) Після зупинки виконання в точці зупинки переглянути значення потрібних змінних на закладці **вікна Autos**.

(показати власний скріншот вікна Autos)

- 7) Продовжити виконання програми в покроковому режимі, натискаючи **F10** (Step Over) або **F11** (Step Into), якщо є необхідність перевірки послідовності виконання команд, або продовжити виконання до наступної точки зупинки, натиснувши **F5**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 8

	<pre> a b; // операція логічного ПОРОЗРЯДНОГО АБО // тільки для цілих чисел // виконуємо в стовпчик 0110 0101 ! - тільки якщо 0 // 1011 0010 0 // в 10-вій системі це 247 → 1111 0111 тоді вихід 0, інакше 1 // результат true, тільки якщо перший АБО другий біт true a ^ b; // операція логічного ПОРОЗРЯДНОГО ВИКЛЮЧАЮЧОГО АБО // тільки для цілих чисел // виконуємо в стовпчик 0110 0101 // 1011 0010 // в 10-вій системі це 215 → 1101 0111 // результат true, якщо перший та другий біт не створюють пару 00 або 11 </pre>
?:	<pre> // операція ДІЇ ЗА УМОВОЮ ((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 Варіант 2: ((a) ? (b) : (c)); // результат операції буде рівний b, якщо a > 0, або a рівна true </pre>
=?:	<pre> d = ((a) ? (b) : (c)); // d стане рівна b, якщо a > 0, або a рівна true </pre>
+= - =	<pre> a += b; // те саме, що і a = a + b, тобто результат дії (a + b) записується у змінну a a -= b; // те саме, що і a = a - b, тобто результат дії (a - b) записується у змінну a </pre>
*= /=	<pre> 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 тільки для цілих чисел </pre>
<= >= == !=	<pre> (a <= b); // результат true, якщо a < або = b (a >= b); // результат true, якщо a > або = b (a == b); // результат true, якщо a дорівнює b (a != b); // результат true, якщо a НЕ дорівнює b </pre>
&= ^= =	<pre> (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) </pre>
<<= >>=	<pre> (a <<= 5); // те саме, що і a = (a << 5) тільки для цілих чисел (a >>= 5); // те саме, що і a = (a >> 5) тільки для цілих чисел </pre>
sizeof	<pre> sizeof(a); // операція, результат якої рівний числу байт, які займає змінна a </pre>

5. Розробити програму, яка обчислює всі операції з таблиці 2 та виводить дані у консоль, як показано у прикладі на наступній сторінці.

(!! Для всіх операцій з таблиці 2 задати різні значення a та b перед кожною операцією в програмі)

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 9

Таблиця 2

Базові операції над змінними та приклади їх запису в програмі

+ -	* /	%	++ --	<< >>	& ^ 	?: =?:	+= -=	*= /= %=	&= ^= = <<= >>=	>= <=	sizeof
a+b a-b	a*b a/b	a%b	a++ ++a a-- --a	a<<b a>>b	a&b a^b a b	a?1:3 a?1.1:0.3 b=a?1.1:0.3	a+=b a-=b	a*=b a/=b a%=b	a&=b a^=b a =b a<<=b a>>=b	a>=b a<=b	sizeof(a)
		a,b- цілі числа		a,b- цілі числа	a,b- цілі числа			для % a,b- цілі числа	a,b- цілі числа		

Приклад того, що має виводити програма

```

*****
Лабораторна робота №1
Група: ЕТ-10
Студент: Добржанський Олександр Олексійович
Дата: 19.09.2024
Тема: Базові операції
*****
Вихідні дані операції « a + b »:
a = 11
b = 12
Результат виконання операції ( a + b ) = 23
*****
Вихідні дані операції « a - b »:
a = 10
b = 14
Результат виконання операції ( a - b ) = -4
*****
Вихідні дані операції « a * b »:
a = 18
b = 2
Результат виконання операції ( a * b ) = 36
*****
Вихідні дані операції « a / b »:
a = 18
b = 2
Результат виконання операції ( a / b ) = 9
*****
Вихідні дані операції « a % b »:
a = 11
b = 3
Результат виконання операції ( a % b ) = 2
*****
Вихідні дані операції « a ++ »:
a = 18
Результат виконання операції ( a ++ ) = 18

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 10

Вихідні дані операції « ++ a »:

a = 1

Результат виконання операції (++ a) = 2

і так до кінця, всі операції з таблиці 2

(!! Навести в звіті обчислення поданих операцій, тобто !! **Всі операції мають бути також обчислені також «вручну» та наведені у звіті**)

6. Висновок

Зробити висновок про те, які операції є базовими, та які функції вони виконують.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 11

Лабораторна робота 2.

Циклічні табличні обчислення засобами C++.

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 **Visual Studio** на мові програмування **C++**.

1. Ознайомитись із завданням:

Керуючись **порядком виконання** (див.нижче) створити **консольну програму**, яка виконує розрахунок значень функції **Y(X)** для діапазону значень **X** та виводить результати в **табличному** вигляді.

Після запуску у вікні програми перед обчисленням **користувач задає**

- **крок зміни X**
- **точність - кількість цифр** для відображення чисел в таблиці
- **ширину - кількість символів** для комірок в таблиці

(Результати виконання пунктів занотуйте у звіт про виконання лабораторної роботи)

2. З таблиці 1 за номером по журналу обрати

- **функцію Y(X)**,
- **діапазон зміни X**,
- **умову** для виведення **Y(X)** в таблицю (тобто виводитимуться лише ті Y, які задовольняють умову)

3. Переглянути **приклад** виконаної програми: файл

lr02_example_KTтаП_1.exe

Відобразіть **поки у текстовому вигляді власний інтерфейс.**

4. Переглянути функцію **Y(X)**.

Визначте оператори математичних операцій – запишіть оператори та назви цих операцій.

Визначити, який тип чисел **Y** та **X** може бути для математичних дій при обчисленні **Y(X)**.

Оберіть найбільш точний серед можливих типів.

5. Створити пустий проект консольної програми.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 12

6. Підключити бібліотеки
 - бібліотеку потоку введення / виведення (див. **попер. л.р.**)
 - бібліотеку функцій windows (див. **попер. л.р.**)

7. Підключити простір імен **std** (для скорочених команд)

8. Підключити підтримку української мови (див. **попер. л.р.**)

9. Придумати **ім'я**, обрати **тип** та **оголосити** змінну для збереження **Y(X)**.

Придумати **ім'я**, обрати **тип** та **оголосити** змінну для збереження **X**.

Придумати **ім'я**, обрати **тип** та **оголосити** змінну для збереження **кроку** для **X**.

Придумати **ім'я**, обрати **тип** та **оголосити** змінну для збереження **точності виведення чисел**.

Придумати **ім'я**, обрати **тип** та **оголосити** змінну для збереження **ширини комірки таблиці**.

Прогляньте таблицю у файлі.exe - прикладі.

Яку ще змінну необхідно оголосити в програмі для виведення таблиці у вигляді, як подано у прикладі ?

Придумати **ім'я**, обрати **тип** та **оголосити** цю змінну.

10. Вивести в програмі **Заголовок** текстового інтерфейсу за прикладом:

```
*****
* Лабораторна робота студента групи АТК-10 ФКІТМР 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 -70 < y(x) < +70 *
*****
```

Запустити, перевірити.

11. Після заголовку повинні виводитись запити користувачу для **введення даних** у змінні за допомогою команд **cout** та **cin** :

```
Введіть величину кроку зміни x: 1  ◀----- (користувач вводить число, тисне Enter)
Введіть кількість значущих цифр для відображення y(x): 4
Введіть ширину поля таблиці у знаках: 10  ◀-----
*****
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 13

Запустити, перевірити.

12. Згадати призначення команд та записати в звіт:

- cout << fixed;
- cout.precision()
- cout.width()

13. Написати команду встановлення кількості значущих цифр для чисел, що виводяться в консоль згідно того, що ввів попередньо користувач.

Підготовка до виведення таблиці:

14. Написати команди для виведення заголовків стовпчиків таблиці як у прикладі:

№з.п.	X	Y(X)
-------	---	------

Далі виведення рядків таблиці у циклі:

15. Вибрати та вставити далі в програму блок циклічних (повторюваних) операцій:

Можна **вибрати один з поширених способів** оголошення циклу в C++:

Цикл з явним налаштуванням	Цикл з передумовою	Цикл з постумовою
<pre>for(int i = 0; i < 10; i++) { // команди тіла циклу }</pre>	<pre>int i = 0; while(i < 10) { // команди тіла циклу i++; }</pre>	<pre>int i = 0; do { // команди тіла циклу i++; } while(i < 10)</pre>

Замінити тут змінну **i** на змінну, яку реально будемо змінювати, тобто змінну **X**.

Проглянути **діапазон X** по варіанту.

Присвоїти **X** початкове значення.

Написати **умову** для виконання **циклу** з урахуванням кінцевого значення **X**

X змінюється після кожного повтору циклу **не на 1**, як в прикладах, а з

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 14

кроком, який раніше задав користувач. Написати для циклу команду зміни **X**.

Далі у тілі циклу:

16.Написати команду обчислення **Y(X)** для поточного **X**.

17.Написати команди виведення **комірки номеру рядка**.

18.Написати команди виведення **комірки поточного X**.

19.Написати команди виведення **комірки поточного Y (X)**.
(! - Це остання команда виведення одного рядка)

Запустити, перевірити.

Якщо необхідно, дописати додаткові команди або скорегувати попередні.

Показати у звіті відображення вікна програми.

20.За варіантом необхідно виводити тільки **Y (X)**, які відповідають умові. В циклі використати оператор умовної дії **if**:

```
if( умова ){
    // дії 1
}
else {
    // дії 2
}
```

щоб виводити **Y (X)** в таблицю тільки тоді, коли виконується **умова** для виведення **Y(X)** за варіантом. Якщо **умова** не виконується, виводити - -

Запустити, перевірити.

Якщо необхідно дописати додаткові команди або скорегувати попередні.

Показати у звіті відображення вікна програми.

21.Перевірити, чи можна підібрати **крок X** так, щоб у таблицю виводилося не більше 50 рядків.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 15

Таблиця 1

**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	$(y \% 13) == 5$
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	$0 < y < 3$
26.	$y = x \& ((x << 3) + (x << 1))$	-127..127	$(y 17) >= 10$
27.	$y = 3.1*tg(-0.5*x) + 5*x^2 - 17$	-20..20	$y < 0 y > 300$
28.	$y = (x << 1) (x << 2) \& (x >> 3) x$	-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	$-17 < y < 17$

Приклад роботи програми:

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

Результат ітераційного обчислення значення функції:

$$y(x) = x^*x*x + \log_{10}(|x|) + \ln(|x|) + \sin(x) + |x|$$

на інтервалі значень $x = -6.28 \dots + 6.28$

за умови, що $y(x)$ буде у межах $-70 \dots +70$

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	--
2	-6.18	--
3	-6.08	--
...		
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	--
49	4.12	--
50	4.22	--
...		

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- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 17

Лабораторна робота 3.

Операції з текстовими файлами

та організація меню користувача засобами C++.

Частина 1. Розробка текстового меню та читання файлів.

Мета роботи: Здобути навички використання командних інструкцій читання символів, слів, рядків з текстового файлу, використання циклу **while**, та командної інструкції перемикачів між варіантами дій **switch-case**, при створенні програм виконання аналізу текстів за вибором користувача і за допомогою меню користувача, у середовищі розробки Microsoft Visual Studio.

1. Ознайомитись з завданням

Слід створити застосунок з таким функціоналом:

- Програма призначена для **аналізу вмісту текстових файлів**.
- Вибір кожної конкретно дії відбувається шляхом введення **№ команди з меню**.
- В меню **відображається ім'я текстового файлу**, з яким працює програма.

Порядок створення застосунку:

2. Підключити бібліотеки

- бібліотеку потоку введення / виведення (див. **попер. л.р.**)
- бібліотеку функцій windows (див. **попер. л.р.**)
- бібліотеку роботи з рядками (**<string>**)
- бібліотеку роботи з файлами на диску (**<fstream>**)

3. Підключити простір імен **std** (для скорочених команд)

4. Підключити підтримку української мови (див. **попер. л.р.**)

5. Придумати ім'я та оголосити змінну типу **Символ**, для збереження **обраної дії користувача**.

6. Придумати ім'я та оголосити змінну типу **Рядок**, для збереження **імені файлу**.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 18

7. Придумати ім'я та оголосити змінну типу **fstream**, для збереження зв'язку з файлом.

8. оголосити нескінченний цикл

```
while ( умова_нескінченності_циклу ) {
    Тіло циклу
}
```

У тілі циклу:

9. Прописати команду очищення вікна консолі **system("cls");**

10. Переглянути файл **.exe** прикладу програми
Вивести у консоль **рядок** для відображення **обраного файлу** (поки файл ми не знаємо)

```
*****
* ім'я файлу з текстом =
*****
```

11. В програмі після рядку * ім'я файлу з текстом = вивести змінну, яка зберігатиме ім'я текстового файлу (коли користувач введе нове ім'я файлу, у нас у меню ці зміни також відобразяться)

12. Вивести нижче **меню** користувача із **№ та скороч. призначенням команд** (**вигляд та зміст меню** має бути як у файлі-прикладі **.exe**)

```
*****
* ім'я файлу з текстом =
*****
* 0: вихід      і: автор      1: задати файл   2: друк файлу   3: кільк.слів   *
* 4: кільк.рядк. 5: кільк.голосн 6: кільк.пригол. 7: кільк.зад.симв. 8: кільк.зад.слів *
* 9: замін.слово
*****
Оберіть номер команди: █
```

Пояснення повних назв

- 0 - вихід з програми
- і - інформація про створювача
- 1 - задати ім'я файлу
- 2 - друк файлу у консоль

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 19

- 3 - лічба кількості слів у тексті
- 4 - лічба кількості рядків у тексті
- 5 - лічба кількості голосних у тексті
- 6 - лічба кількості появи заданого символу у тексті
- 7 - лічба кількості появи заданого слова у тексті
- 8 - заміна слова у тексті

13. Після меню вивести рядок-запит на введення користувачем команди "Оберіть номер команди > "

14. Застосувати `cin` для введення користувачем символу у змінну **обраної дії користувача** з пункту 4.

15. Прописати оператор **switch-case**, що керується змінною **обраної дії користувача** з пункту 4.

Приклад:

```
switch ( ) {
    case 'a': {
        команди;
    }
    break;
    case 'b': {
        команди;
    }
    break;
    default: {
        команди;
    }
}
```

В операторі switch-case:

16. Підлаштувати під наші потреби пусту структуру з **case** з попереднього пункту

- **!!** кейси для можливих команд '0'

'i'

'1'

'2'

і до '8' (одинарні символи)

17. Заповнити **case** для 'i' (i - інформація про створювача)

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 20

- прописати виведення у консоль даних про створювача:
ВНЗ, факультет, група, ПІБ, дата

- затримати напис у консолі до натиснення будь-якої клавіші за допомогою
system("pause");

18. Заповнити **case** для **'0'** (**0** - **вихід** з програми)

- застосувати звичайну команду виходу із програми (див. **попер. л.р.**)

19. Прописати **case** для випадків, коли **жоден case не спрацював**
(див. **п.14**)

- організувати у консоль повідомлення **"Невірно набрана команда"**
у випадку введення користувачем
символу,
якому не співставлена жодна команда
меню

- затримати напис у консолі до натиснення будь-якої клавіші за допомогою
system("pause");

20. Заповнити **case** для **'1'** (**1** - задати **ім'я файлу**)

- застосувати звичайну **cin** для запису користувачем імені файлу у
змінну **імені файлу**

- виконати перевірку наявності файлу для читання

Приклад:

```

підставити свою змінну зв'язку з файлом типу fstream
підставити свою змінну з іменем файлу
↙
↘
file.open( "ім'я файлу", ios::in );
if (!file) {
    cout << "Помилка відкриття файлу" << endl<<endl;
    system("pause");
} else{
    file.close();
}

```

↳ означає файл для читання

21. Заповнити **case** для **'2'** (**2** - **друк файлу** у консоль)

- виконати відкриття файлу для читання
підставити свою змінну **_зв'язку_з_файлом**

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 21

«ім'я файлу» у прикладі замінити на змінну з іменем файлу з пункту 5

```
file.open( "ім'я файлу" , ios::in );
```

- виконати читання файлу по рядкам та виведення кожного рядку у
КОНСОЛЬ

підставити свою змінну зв'язку з файлом типу **fstream**

Приклад:

підставити свою змінну типу **string**

```
while ( getline(file, data) ) {  
    cout << line << endl;  
}  
file.close();  
cout<< endl;
```

- затримати напис у консолі до натиснення будь-якої клавіші за допомогою

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

22.Заповнити case для '3' (3 - лічба кількості слів у тексті)

- виконати відкриття файлу для читання
підставити свою змінну зв'язку з файлом
«ім'я файлу» у прикладі замінити на змінну з іменем файлу з пункту 5

```
змінна_зв'язку_з_файлом.open( "ім'я файлу" , ios::in );
```

- далі керуватись прикладом

Приклад:

```
int wordCount = 0;  
string word;  
while ( file >> word ) {  
    wordCount++;  
}  
file.close();  
cout << "Кількість слів у тексті: " << wordCount << endl << endl;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 22

- затримати напис у консолі до натиснення будь-якої клавіші за допомогою

system("pause");

23. Заповнити **case** для **'4'** (4 - лічба кількості рядків у тексті)

- використати читання файлу рядками, як у **case** для **'2'**,

- лічбу рядків виконати як в **case** для **'3'**

- затримати напис у консолі до натиснення будь-якої клавіші за допомогою

system("pause");

24. Заповнити **case** для **'5'** (5 - лічба кількості голосних у тексті)

- використати читання файлу рядками, як у **case** для **'3'**,

але

- замінити тип змінної, куди пишемо з файлу, зі **string** на **char**

- команду збільшення лічильника на 1 прописати у команді **if (умова)**, де умова перевірки **string** буде складна (**чи == 'a'**) **АБО** (**чи == 'A'**) **АБО** (**чи == 'e'**) **АБО** (**чи == 'E'**) і так далі всі голосні великі та малі літери англійського алфавіту (уточнити в Internet)

- затримати напис у консолі до натиснення будь-якої клавіші за допомогою

system("pause");

25. (по можливості) Для кейсів де відбувається лічба, перед зверненням до файлу виконати **перевірку наявності файлу**, як у п.20

26. Запустити програму і перевірити, чи програма виконуватиметься за алгоритмом (послідовністю дій):

Початок циклу: очистити вікно консолі -> вивести початкові дані (поточне ім'я файлу) ->

-> вивести меню -> запит на введення обраного пункту меню -> очікувати введення -> виконати обраний пункт

меню -> очікувати натиснення будь-якої клавіші -> на Початок циклу

27. Висновок

Зробити висновок про те, які функції необхідно застосувати для того, щоб виконати заданий алгоритм роботи застосунку.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 23

Лабораторна робота 4.

Операції з текстовими файлами

та організація меню користувача засобами C++.

Частина 2. Інтеграція функцій обробки текстових даних.

Мета роботи: Здобути навички використання командних інструкцій читання символів, слів, рядків з текстового файлу, використання циклу **while**, та командної інструкції перемикачів між варіантами дій **switch-case**, при створенні програм виконання аналізу текстів за вибором користувача і за допомогою меню користувача, у середовищі розробки Microsoft Visual Studio.

- Заповнити **case** для **'6'** (6 - лічба кількості приголосних у тексті)
 - використати читання файлу по одному символу, як у **case** для **'5'**,
 - ознайомитись з функцією перевірки на те, чи знак є літерою алфавіту
Наприклад: **isalpha(введений_символ)**, функція повертає true, якщо символ є алфавітним, тобто лише літера, а не знак, чи цифра
 - скласти умову
(**чи це літера алфавіту**) і (**чи не = 'a'**) і (**чи не = 'A'**) і (**чи не = 'e'**) і (**чи не = 'E'**) і так далі
всі голосні великі та малі літери англійського алфавіту (як у **case** для **'5'**),
!! тобто все навпаки: якщо (**if**) справджується, що літера голосна, то нічого не робимо,
інакше (**else**) літера не голосна - значить приголосна, тоді збільшуємо лічильник на 1.
 - затримати напис у консолі до натиснення будь-якої клавіші за допомогою
system("pause");
- Заповнити **case** для **'7'** (7 - лічба кількості появи заданого символу у тексті)
 - оголосити додаткову змінну типу **char**, яка зберігатиме заданий символ
 - вивести в консоль запит до користувача **ввести заданий символ**
 - **зчитати введений користувачем символ** у додаткову змінну

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 24

- використати читання файлу по одному символу, як у **case** для '5',
- команду збільшення лічильника на 1 прописати у команді **if (умова)**, де умова передбачає перевірку, чи зчитаний символ рівний заданому
- затримати напис у консолі до натиснення будь-якої клавіші за

ДОПОМОГОЮ

system("pause");

3. Заповнити **case** для '8' (8 - лічба кількості появи заданого слова у тексті)

- оголосити **додаткову змінну типу string**, яка зберігатиме задане слово

- вивести в консоль запит до користувача **ввести задане слово**
- **зчитати введене користувачем слово** у додаткову змінну
- використати читання файлу по одному слову, як у **case** для '3',
- команду збільшення лічильника на 1 прописати у команді **if (умова)**, де умова передбачає перевірку, чи зчитане слово рівне заданому
- затримати напис у консолі до натиснення будь-якої клавіші за

ДОПОМОГОЮ

system("pause");

4. Заповнити **case** для '9' (9 - заміна заданого слова у тексті на інше задане слово-замінник)

- оголосити **додаткову змінну типу string**, яка зберігатиме слово-замінник

- вивести в консоль запит до користувача **ввести слово-замінник**
- **зчитати введене користувачем слово-замінник** у додаткову змінну
- оголосити **додаткову змінну типу string**, яка зберігатиме слово, яке слід замінити

- вивести в консоль запит до користувача **ввести слово, яке слід замінити**

- **зчитати введене користувачем слово, яке слід замінити** у додаткову змінну

- **відкрити додатковий файл-результату, для запису результату заміни**, причому ім'я файлу має містити рядок **"(new)"** + ім'я файлу, з якого читаємо

- використати читання основного файлу по одному слову, як у **case** для '3',

- команду **запису слова-замінника у файл-результату** та збільшення лічильника замін на 1 прописати у команді **if (умова)**, де умова передбачає перевірку, чи зчитане слово рівне слову, яке слід замінити
- команду **запису зчитаного слова без заміни у файл-результату** прописати у частині **else** команди **if**

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 25

- закрити файли: файл-результату так само, як і файл основний
- вивести у консоль повідомлення про результат, наприклад,
«виконано 7 замін»
- затримати напис у консолі до натиснення будь-якої клавіші за допомогою
`system("pause");`

5. (по можливості) Для кейсів де відбувається лічба, перед зверненням до файлу виконати перевірку наявності файлу, як у п.19

6. Висновок

Зробити висновок про те, які функції необхідно застосувати для того, щоб виконати заданий алгоритм роботи застосунку.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 26

Лабораторна робота 5.

Операції з масивами та текстовими файлами мовою C++.

Частина 1. Розробка текстового меню.

Мета роботи: Здобути навички використання командних інструкцій виведення даних **cout**, введення даних **cin**, циклу **do-while** (або **while**, або **for**) та командної інструкції перемикавання 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 Studio.

1. Ознайомитись із завданням: В ході роботи потрібно буде скласти консольну програму, яка реалізує роботу зі
 - списком факультетів,
 - списком груп для кожного з факультетів,
 - списком студентів для кожної з груп відповідного факультету

В консоль виводиться меню користувача:

```

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

```

із зазначенням таких команд:

- «**help**» - виведення пояснення до команд меню
- «**exit**» - вихід з програми
- «**change level**» - зміна списку (факультети, групи, студенти)
- «**show**» - показ елементів списку
- «**add**» - додавання елементу до списку

У нижній частині меню виводиться рядок про поточний обраний список, один з наступних:

- «ОБРАНО СПИСОК: ФАКУЛЬТЕТИ»
- «ОБРАНО СПИСОК: ГРУПИ»
- «ОБРАНО СПИСОК: СТУДЕНТИ»

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 27

Результати виконання команд меню виводяться у консоль після введення вибору користувача та його підтвердження клавішею Enter.

Після виведення результату програма

- очікує натиснення будь-якої клавіші,
- потім очищує екран
- знову виводить меню,
- назву поточного обраного списку та запит на введення команди (її номеру) користувачем.

2. Створити новий проект консольний застосунок Windows

3. Приєднати файли заголовку та простори імен

Для стандартних функцій та операторів:

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;
```

4. Оголосити потрібні змінні

Оголошуємо змінні перед інструкцією початку головної програми **main** { :

(якщо оголосити змінні перед описом **main** – це будуть глобальні змінні)

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- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 28

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:

int iDepartmentsNum = 0;

0 присвоюється одразу, оскільки поки що будь-які факультети відсутні, їх кількість = 0.

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:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 29

`int i, j;`

Змінна цілого типу для збереження № поточного обраного факультету:

`int iChooosedDepartmentNum;`

Змінна цілого типу для збереження № поточної обраної групи:

`int iChooosedGroupNum;`

Змінна цілого типу для збереження № поточного обраного студента:

`int iChooosedStudentNum;`

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

`sLevel = "ФАКУЛЬТЕТИ";`

5. Порядок забезпечення сумісності програми з кириличними шрифтами:

Після оголошення змінних у основній функції:

```
int main() {

    return 0;
}
```

починаючи з початку її опису, тобто після першої `{`, прописуємо першу інструкцію програми, яка необхідна для обрання кодової сторінки для даних, що виводяться у консоль, сумісної з кириличним шрифтом (кодова сторінка 1251):

`SetConsoleOutputCP(1251);`

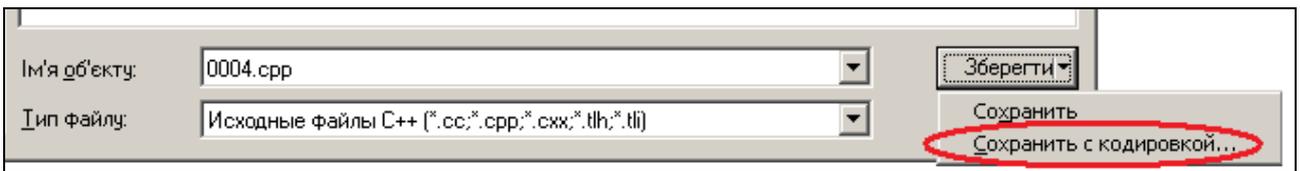
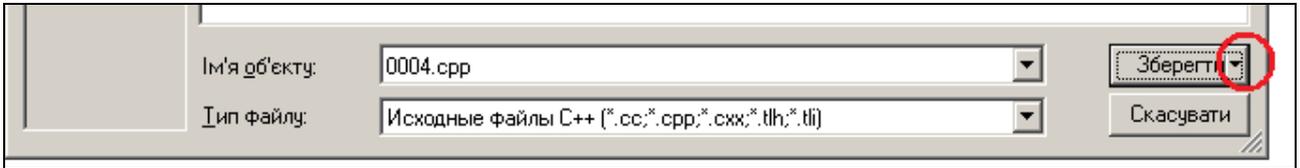
Прописуємо також інструкцію обрання кодової сторінки для даних, що вводяться у консоль користувачем, сумісної з кириличним шрифтом (кодова сторінка 1251):

`SetConsoleCP(1251);`

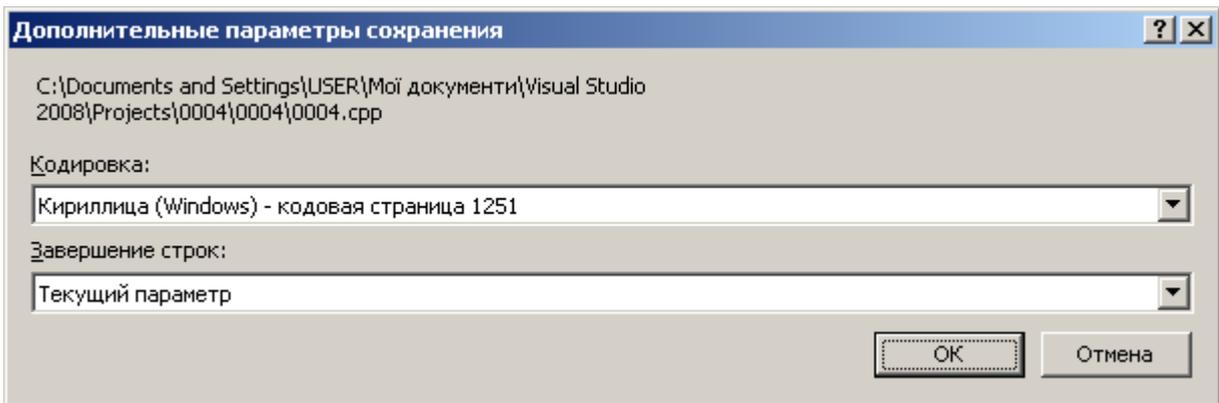
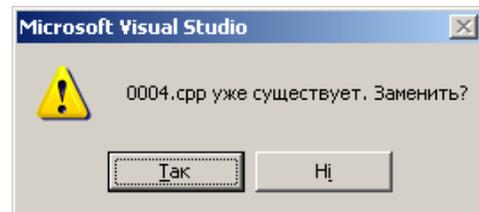
Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 30

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

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



Підтвердити **ПЕРЕЗАПИС (ЗАМІНУ)** файлу: **«ТАК»**



Підтвердити рішення.

Одним блоком додати команди встановлення для консолі шрифту, що підтримує кодування 1251 (шрифту Lucida Console).

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

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;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 31

```
i = 0; while (cfi.FaceName[i] = LucidaConsole[i]){i++;}
SetCurrentConsoleFontEx(GetStdHandle(STD_OUTPUT_HANDLE), FALSE, &cfi);
```

6. 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
{
// 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 << "
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;
```

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
+-----+-----+-----+-----+
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 32

>

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 символьній змінній «chChoose». Отже використаємо відповідні інструкції з використанням команди введення даних з консолі:

```
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:
```

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 введенні користувачем команди меню «1»
}
case '2':
{
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 33

**тут будуть командні інструкції дій програми,
при введенні 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':
```

```
{
```

```
}
```

```
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 розмішатимемо 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;
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;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 34

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.

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- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 35

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.

```
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 '4':
{
    if (sLevel=="ФАКУЛЬТЕТИ")
    {
        тут будуть команди для роботи зі списком факультетів
    }
    else if(sLevel=="ГРУПИ")
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 36

```

    {
        тут будуть команди для роботи зі списком груп
    }
    else if(sLevel=="СТУДЕНТИ")
    {
        тут будуть команди для роботи зі списком студентів
    }
}
system("pause");
break;

```

Повторимо, що потрібно зробити те саме у всіх інших кейсах: **case '3'**, **case '5'**, **case '6'**, **case '7'**.

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:

```

У case '4':
{
у частині
    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:

```
cout << "Введіть назву нового факультету > ";
```

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

```
cin >> sInput;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 37

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:

У case '4':

у частині

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

Помістимо сюди команди для додавання групи у список

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++)
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 38

```
{
    if(sDepartmentsNames[i]==sInput)
    {
        iChoosedDepartmentNum = i;
    }
}
```

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

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

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

У останню конструкцію **else { }**

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

```
iGroupsNum[iChoosedDepartmentNum]++;
```

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

```
sGroupsNames[iChoosedDepartmentNum][iGroupsNum[iChoosedDepartmentNum]] = 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- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 39

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;
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";}
else
{
    cout << "Введіть ім'я нового студента > ";
    cin >> sInput;
    iStudentsNum[iChosedDepartmentNum][iChosedGroupNum]++;
    sStudentsNames[ ][ ]=sInput;
    cout << endl << "Нового студента додано" << endl;
}
```

Пусті номери індексів [][] елементів масиву імен студентів заповнимо потрібними номерами:

№ факультету - **iChosedDepartmentNum**
 № групи - **iChosedGroupNum**
 № студента - **iStudentsNum[iChosedDepartmentNum][iChosedGroupNum]**

Перевіримо, що після конструкції:

```
else if(sLevel=="СТУДЕНТИ")
{
    яка вже наповнена командами
}
```

наступним оператором є закриваюча дужка **}** оператора case '4',

після якої розміщено очікування реакції користувача на виконані дії.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 40

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

та оператор

```
break;
```

(Пояснити у звіті призначення оператора `break;` в командній-структурі `case`)

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;
```

- оператори виведення списку груп:

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

- оператори виведення списку студентів:

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

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 41

cout << endl;

Перевірка:

- Спробуйте створити список факультетів і вивести його, командою 3 (show)
- Спробуйте створити список груп і вивести його, командою 3 (show)
- Спробуйте створити список студентів і вивести його, командою 3 (show)

Відмітьте у звіті тип помилок виконання програми, якщо такі відбулись.

Виправлення помилок !!

Могла виникати 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:

(до вже існуючого коду додати виділені зеленим фрагменти)

```
int iDepartmentsNum = 0;
```

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

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

(до вже існуючого коду додати виділені зеленим рядки)

```
case '4':
```

```
    if (sLevel=="ФАКУЛЬТЕТИ")
```

```
    {
```

```
        ...
```

```
            sDepartmentsNames[iDepartmentsNum] = sInput;
```

```
            iGroupsNum[iDepartmentsNum]= 0;
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 42

...

3) При додаванні групи, не було визначено кількість студентів у новій групі.
(до вже існуючого коду **додати** виділені зеленим рядки)

У case '4':

```
{
  у частині
  else if(sLevel=="ГРУПИ")
  {
```

після

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

```
iStudentsNum[iChooosedDepartmentNum][iGroupsNum[iChooosedDepartmentNum]] = 0;
```

!! Ми поки створили лише кейси для команд 1, 2, +, -, 3, 4 наступні кейси
будемо створювати в наступних лабораторних роботах.

7. Висновок

Зробити висновок про те, які функції необхідно застосувати для того,
щоб виконати заданий алгоритм роботи застосунку.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 43

Лабораторна робота 6.

Операції з масивами та текстовими файлами мовою C++.

Частина 2. Інтеграція функцій роботи 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++.

Примітка:

У попередній л.р. вже було прописано інструкції для виведення списку студентів, груп, факультетів.

Але виводився абсолютно весь список, незалежно від приналежності групи факультету, або студента певній групі. виправимо це.

1. 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") {
        ОЧИСТИТИ ВМІСТ ПУНКТУ
    }
    else if(sLevel=="ГРУПИ") {
        ОЧИСТИТИ ВМІСТ ПУНКТУ
    }
    else if(sLevel=="СТУДЕНТИ") {
        ОЧИСТИТИ ВМІСТ ПУНКТУ
    }
}

```

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

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 44

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

```
case '3': {
    if (sLevel=="ФАКУЛЬТЕТИ")
    {
        тут будемо писати 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 { }:

```
case '3': {
    if (sLevel=="ФАКУЛЬТЕТИ")
    {
        вже є код
    }
    else if(sLevel=="ГРУПИ")
    {
        Тут будемо писати нові 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 команди:

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

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 45

```

iChosedDepartmentNum = 0;
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

    else if (sLevel=="СТУДЕНТИ")
    {
        Сюди додамо оператори для виведення студентів
    }
}

```

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;
    }

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 46

```

}
else
{
    cout << endl;
    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».

У case '5':

```

if (sLevel=="ФАКУЛЬТЕТИ")
{
    Сюди додамо код для пошуку факультетів
}

```

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

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 47

```

{
    cout<<endl<<" Факультет " << sDepartmentsNames[iChooosedDepartmentNum];
    cout <<" знайдено"<< endl;

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

У case '5':

```

if (sLevel=="ГРУПИ")
{
    Сюди додамо код пошуку груп
}

```

Ось цей код пошуку груп:

```

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

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 48

```

cout<<" - знайдено таку групу на факультеті " <<
    sDepartmentsNames[iChosedDepartmentNum] << endl;
cout<<" у групі: "<< iStudentsNum[iChosedDepartmentNum][iChosedGroupNum]
<< " студентів" << endl;
cout << endl;
}

```

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

У case '5':

```

...
if (sLevel=="СТУДЕНТИ")
{
    Сюди додамо код пошуку студентів
}

```

І ось такий, абсолютно не складний код нижче:

```

cout << " введіть ім'я студента для пошуку > ";
cin >> sInput;
iChosedStudentNum = 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)
                {
                    iChosedDepartmentNum = i;
                    iChosedGroupNum = j;
                    iChosedStudentNum = q;
                    cout<<endl;
                    cout<<" - знайдено студента "<<sStudentsNames[i][j][q]<<endl;
                    cout<<" на факультеті: "
                    cout<<sDepartmentsNames[iChosedDepartmentNum]<< endl;
                    cout<<" у групі: ";
                    cout<<sGroupsNames[iChosedDepartmentNum][iChosedGroupNum
                    ];
                    cout<<endl << endl;
                }
            }
        if (iChosedStudentNum == 0)
            {
                cout<<"Такого студента не існує"<<endl<<endl;
            }
}

```

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

!! Якщо ця бібліотека ще не підключена, то виконати це самостійно.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 49

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

І у останню чергу розглянемо:

Дії при виборі користувачем команди «6» («edit»), якщо вже обрано рівень «ФАКУЛЬТЕТИ».

Дії при виборі користувачем команди «6» («edit»), якщо вже обрано рівень «ГРУПИ».

Дії при виборі користувачем команди «6» («edit»), якщо вже обрано рівень «СТУДЕНТИ».

Подальші фрагменти кодів також по черзі будемо розміщувати між:

```

case '6':
{
    if (sLevel=="ФАКУЛЬТЕТИ")
    {
        Сюди додамо команди для роботи зі списком факультетів
    }

    else if(sLevel=="ГРУПИ")
    {
        Сюди додамо команди для роботи зі списком груп
    }

    else if(sLevel=="СТУДЕНТИ")
    {
        Сюди додамо команди для роботи зі списком студентів
    }
}
system("pause");
break;

```

Дії при виборі користувачем команди «6» («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)
    {

```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 50

```

iChosedDepartmentNum = i;
}
}
if (iChosedDepartmentNum == 0)
{
    cout<<"Такого факультету не існує"<<endl<<endl;
}
else
{
    cout<<endl<<" Ведіть нове ім'я факультету >";
    cin >> sInput;
    sDepartmentsNames[iChosedDepartmentNum]=sInput;
    cout << endl;
    cout << "Зміни прийнято"<<endl;
    cout << endl;
}

```

Дії при виборі користувачем команди «6» («edit»), якщо вже обрано рівень «ГРУПИ».

(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++)
{
    if(sGroupsNames[i][j]==sInput)
    {
        iChosedDepartmentNum = i;
        iChosedGroupNum = j;
    }
}
if (iChosedGroupNum == 0)
{
    cout<<"Такої групи не існує"<<endl<<endl;
}
else
{
    cout<<endl;
    cout<<" Введіть нове ім'я групи > ";
    cin >> sInput;
    sGroupsNames[iChosedDepartmentNum][iChosedGroupNum]=sInput;
    cout <<endl;
    cout << " Зміни прийнято"<<endl;
    cout << endl;
}

```

Дії при виборі користувачем команди «6» («edit»), якщо вже обрано рівень «СТУДЕНТИ».

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 51

(! 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
{
cout<<" Введіть діюче ім'я студента > ";
cin >> sInput;
string * ptr_find;
string * ptr_begin = &sStudentsNames[iChosedDepartmentNum][iChosedGroupNum][0];
string * ptr_end = &sStudentsNames[iChosedDepartmentNum][iChosedGroupNum][49];
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>**. Бібліотеку **<algorithm>** підключити самостійно.

Перевірка:

- створіть факультет (досить одного), наповніть його групами (досить однієї), групи наповніть студентами (досить двох-трьох)
- спробуйте викликати для певного студента функції з бібліотеки **<algorithm>**, а саме функцію 5 (Edit)

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	<i>Випуск 1</i>	<i>Зміни 0</i>	<i>Екземпляр № 1</i>	<i>Арк 62 / 52</i>

!! Задля уникнення помилок при використанні функцій бібліотеки **<algorithm>**, рекомендується, щоб змінні, які містять масиви даних про студентів, факультети та групи, були глобальними, тобто оголошеними до початку опису головної функції main(). Тому можна просто зробити всі змінні глобальними.

2. Висновок

Зробити висновок про те, які функції необхідно застосувати для того, щоб виконати заданий алгоритм роботи застосунку.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 53

Лабораторна робота 7-8.

Операції з масивами та текстовими файлами мовою C++.

Частина 3. Інтеграція функцій роботи з текстовими файлами.

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++.

Примітка:

!! Задля уникнення помилок при використанні функцій бібліотеки `<algorithm>`, рекомендується, щоб змінні, які містять масиви даних про студентів, факультети та групи, були глобальними, тобто оголошеними до початку опису головної функції `main()`. Тому можна просто зробити всі змінні глобальними.

1. 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**:

case '7':

```
if (sLevel=="ФАКУЛЬТЕТИ")
```

```
{
```

тут розташуємо

команди для видалення обраного факультету

команди для видалення всіх груп факультету

команди для видалення всіх студентів всіх груп факультету

```
}
```

```
//////////кінець частини case 7 видалення факультету//////////
```

```
else if(sLevel=="ГРУПИ")
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 54

```

{
    тут розташуємо
    команди для видалення обраної групи
    команди для видалення всіх студентів групи
}
//////////кінець частини case 7 видалення групи//////////
else if(sLevel=="СТУДЕНТИ")
{
    тут розташуємо
    команди для видалення обраного студента
}
//////////кінець частини case 7 видалення студента//////////
system("pause");
break;
//////////кінець case 7//////////
case '8':

```

Тепер вноситимемо по черзі блоки команд.

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
}
//////////кінець частини 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)
// переберемо по всім факультетам всі групи і зупинимось на співпадаючій назві
iChooosedGroupNum = 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- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 55

```

for (j = 1; j<=iGroupsNum[i]; j++)
    for (q=1; q<=iStudentsNum[i][j]; q++)
    {
        if(sGroupsNames[i][j]==sInput)
            {
                iChosedDepartmentNum = i; // фіксуємо номер факультету
                iChosedGroupNum = j; // фіксуємо номер групи
            }
    }
if (iChosedGroupNum == 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[iChosedDepartmentNum][ iChosedGroupNum][0];
        string * ptr_end = &sStudentsNames[iChosedDepartmentNum][ iChosedGroupNum][49];
        ptr_find = find(ptr_begin, ptr_end + 1 , sInput);
        // якщо значення знайденого покажчика рівне введеному імені,
        if (* ptr_find == sInput)
            {
                // позначаємо студента міткою для видалення
                * ptr_find = "Видалити";
                cout<<endl<<" студента знайдено у списку групи "<<endl;
                // remove() це засіб <algorithm>, він видалить вказаний у параметрах
                елемент масиву
                // і змістить інші елементи масиву на пuste місце
                ptr_begin = &sStudentsNames[0][0][0];
                ptr_end = &sStudentsNames[9][49][49];
                remove(ptr_find, ptr_end+1, "Видалити");
                // змінюємо кількість студентів на 1
                iStudentsNum[iChosedDepartmentNum][iChosedGroupNum]--;
                cout <<" та видалено "<<endl<<endl;
            }
        else
            {
                cout<<endl<<" Такого студента не існує"<<endl<<endl;
            }
    }
}

```

Щодо студента цього коду достатньо.

Самостійно:

- Дати відповідь у звіті: Чому при видаленні імені студента просто не присвоїти пустий рядок " " для цього елемента масиву імен студентів?
- Дати відповідь у звіті: Чи можливо обійтися без функції find(), а лише функцією remove()? Як тоді виконуватиметься алгоритм при наявності двох однакових імен у групі?

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 56

Візьмемося створити код для видалення групи.

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

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 частини 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
та 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++)
        {
            if(sGroupsNames[i][j]==sInput)
                {
                    iChooosedDepartmentNum = i;
                    iChooosedGroupNum = j;
                }
        }
if (iChooosedGroupNum == 0)
{
    cout<<"Такої групи не існує"<<endl<<endl;
}
else
{
    // якщо група існує, сповістимо користувача
    cout<<endl;
    cout<<" знайдено групу " << sInput;
    cout <<" на факультеті "<<sDepartmentsNames[iChooosedDepartmentNum]<< endl;
    // видаляємо ім'я групи зі списку груп
    string * ptr_begin = &sGroupsNames[0][0];
    string * ptr_end = &sGroupsNames[9][49];
    remove(ptr_begin, ptr_end + 1, sInput);
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 57

```

// мітимо для видалення число зі списку кількості студентів по групах
// мітка = 100 (унікальна, бо кількість студентів не більша 49)
iStudentsNum[iChooosedDepartmentNum][iChooosedGroupNum]=100;
// видаляємо мічене
int * ptr_begin_ = &iStudentsNum[0][0];
int * ptr_end_ = &iStudentsNum[9][49];
remove(ptr_begin_, ptr_end_ +1, 100);
// 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 = &sStudentsNames[9][49][49];
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 засїб find() для пошуку групи.

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

Дїї при виборї користувачем команди «7» («del»), якщо вже обрано рївень «**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
}
//////////кінець частини case 7 видалення факультету//////////

```

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

команди для видалення обраного факультету
та команди для видалення всіх груп факультету
та ще й команди для видалення всіх студентів всіх груп факультету

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 58

```

// обираємо факультет
cout << " введіть ім'я факультету для видалення > ";
cin >> sInput;
iChooosedDepartmentNum = 0;
for (i = 1; i<=iDepartmentsNum; i++)
{
    if(sDepartmentsNames[i]==sInput)
    {
        iChooosedDepartmentNum = i;
    }
}
if (iChooosedDepartmentNum == 0)
{
    cout<<endl<<" Такого факультету не існує"<<endl<<endl;
}
else
{
    cout<<endl<<" Факультет " << sInput <<" знайдено"<< endl;
    // видалятимемо ім'я факультету зі списку
    string * ptr_begin = &sDepartmentsNames[0];
    string * ptr_end = &sDepartmentsNames[9];
    remove(ptr_begin, ptr_end + 1, sInput);
    // видалятимемо дані про кількість груп на факультеті
    int * ptr_begin_ = &iGroupsNum[0];
    int * ptr_end_ = &iGroupsNum[9];
    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 =&sGroupsNames[9][49];
    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_ = &iStudentsNum[9][49];
    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 = &sStudentsNames[9][49][49];
    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;
}

```

Перевірка:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 59

- створіть факультет (досить одного), наповніть його групами (досить однієї), групи наповніть студентами (досить двох-трьох) !! але студенти мають бути з однаковими іменами
- спробуйте викликати для певного студента функції з бібліотеки <algorithm>, а саме функцію 7 (Dell)
- після видалення спробуйте функцію показ списку групи (show) для групи, де був видалений студент
- поясніть спосіб роботи функції видалення елементів масиву з бібліотеки <algorithm>
- поясніть в чому некоректність запропонованого коду видалення студентів
- спробуйте самостійно виконати зміни коду, які уможливають видалення лише одного студента з однаковими іменами (наприклад першого, знайденого за іменем)

Самостійно

Доповнити меню програми командами згідно рисунку

```

+-----+-----+-----+-----+
КОМАНДИ: | 1 help | 2 exit | +/- change level |
+-----+-----+-----+-----+
3 show | 4 add | 5 find | 6 edit | 7 del |
+-----+-----+-----+-----+
8 save | 9 load | 0 make report to file |
+-----+-----+-----+-----+
ОБРАНО РІВЕНЬ: ФАКУЛЬТЕТИ
+-----+-----+-----+-----+
> |

```

Приступимо до команд роботи з файлами.

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

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 '8': // case для запису бази
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 60

тут розташуємо
команди запису файлу-бази
system("pause");
break;

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

case '9': // case для читання бази
тут розташуємо
команди читання файлу-бази
system("pause");
break;

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

case '0': // case для запису звіту
тут розташуємо
команди запису файлу-звіту
system("pause");
break;

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

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
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
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 61

```

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: "<<slnput<<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. 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 потрібно зчитувати з файлу символи-розділювачі: пробіли ' '.

команди читання файлу-бази

```

cout<<endl<<" Введіть ім'я файлу > ";
cin>>slnput;
slnput+=" .txt";
    далі
    цю частину команд виконати самостійно
    ( можна зкопіювати її з case 8 )
    тільки
    при відкритті файлу застосовуємо байти прапорців режиму відкриття як ios::in
    << міняємо на >>
    ' ' не зчитуємо (взагалі прибираємо його з коду)
і закінчуємо блок команд
командою
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.

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015			Ф-20.10- 05.02/141.00.1/Б/ОК13- 2024
	Випуск 1	Зміни 0	Екземпляр № 1	Арк 62 / 62

команди запису файлу-звіту

```

cout<<endl<<" Введіть ім'я файлу > ";
cin>>slInput;
slInput+="_report.txt";
my_file.open(&slInput[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<<" "<<sDepartmentsNames[i]<<": "<<iGroupsNum[i]<<" груп"<<endl;
    }
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: "<<slInput<<endl<<endl;

```

Самостійно:

- У «case 0» модифікуючи послідовність запису змінних у файл, обираючи відступи та формуючи рядки, додаючи певні пояснення, що записуються в файл, забезпечити запис файлу звіту з власним дизайном розміщення інформації у ньому.

2. Висновок

Зробити висновок про те, які функції необхідно застосувати для того, щоб виконати заданий алгоритм роботи застосунку.