ДНІПРОВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. О. ГОНЧАРА ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ МАТЕМАТИКИ ТА МАТЕМАТИЧНОЇ КІБЕРНЕТИКИ

Лабораторна робота №4

на тему «Узагальнене програмування»

з курсу «Об’єктно – орієнтоване програмування»

Варіант № 8

Виконав:

студент групи ПА-19-2

Ільяшенко Єгор

Дніпро, 2021

**Зміст**

[**Постановка задачі** 3](#_Toc73311939)

[**Індивідуальний варіант** 4](#_Toc73311940)

[**Опис розв’язку** 5](#_Toc73311941)

[**Література** 7](#_Toc73311942)

[**Висновки** 8](#_Toc73311943)

# **Постановка задачі**

Розробити програму – консольний застосунок для ОС Windows, що демонструє можливості мови C++, а також його стандартної бібліотеки по роботі з потоками, контейнерами та ітераторами. Використання всіх вищезгаданих можливостей є обов’язковим в кожному з варіантів індивідуальних завдань. В кожному з нижченаведених варіантів завдань Вами повинні бути реалізовані основні можливості бібліотеки стандартних шаблонів: ітератор, пошук, упорядкування, сума або щось аналогічне зі згоди викладача. При написанні програми слід враховувати необхідність забезпечення її зборки у двох режимах:

1. Створення виконуваного файлу із підключенням/використанням типів і алгоритмів бібліотеки стандартних шаблонів (STL).
2. Створення виконуваного файлу без застосування STL.

Наприклад: Ваше завдання – втілити алгоритм швидкого сортування для контейнеру «вектор»; в такому разі Ви мусите продемонструвати, як Ваша програма працює із Вашою реалізацією вектора, а також як вона працює із стандартною реалізацією вектора з бібліотеки STL. Увага: Ви пишете одну програму, а не дві. Необхідні зміни вихідного тексту для переналаштування з/на STL бажано робити за допомогою директив умовної компіляції.

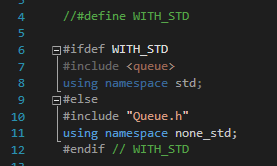
# **Індивідуальний варіант**

Варіант №8

Визначити чергу, використовуючи два стека. Визначити чергу таким чином, щоб вона повністю замінювала чергу із namespace std. Використовувати узагальнене програмування.

# **Опис розв’язку**

Для того, щоб зробити копію queue з std, але на двох stack, я використовував директиви препроцесора. А саме:



Коли ми за допомогую #define оголошуємо макрос WITH\_STD, програма у конструкції

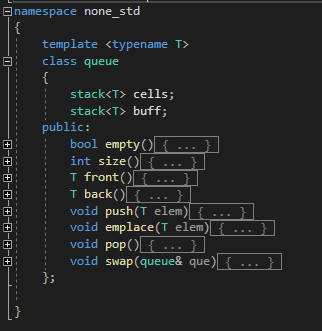
#ifdef WITH\_STD

…

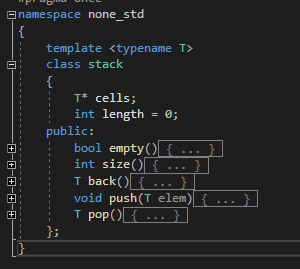
#else

…

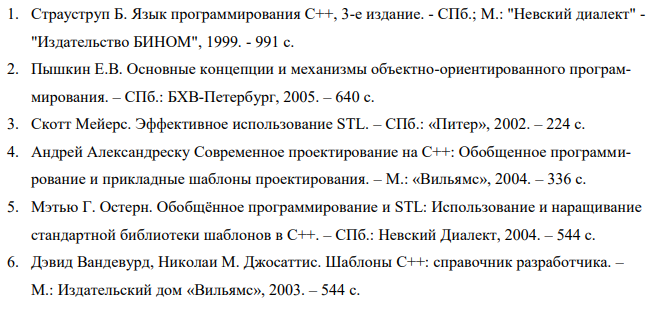
буде виконувати перший блок коду. Таким чином ми можемо в одній програмі написати 2 її версії. Для розділення свою чергу я написав у namespace none\_std.



Сама черга використовує 2 стека, як і було сказало у завданні. Перший стек – це сама черга, її елементи, розташовані у вірній послідовності. Другий стек необхідний, як буфер при додаванні нових елементів. При додаванні нового елемента ми забираємо елемент з 1 стека та ставимо його у інший. І так до кінця. Після чого додаємо новий елемент у буфер та переставляємо усі елементи у перший стек. Таким чином можна побудувати чергу на двох стеках.



# **Література**



# **Висновки**

Виконуючи лабораторну роботу №4 з Об’єктно-орієнтованного програмування, я дізнався про директиви препроцесора: #define, #ifdef, #else, #elif. Дізнався, що таке макроси у програмуванні, а також навчився використовувати парадигму програмування “Узагальнене програмування”, яка дозволяє робити методи для узагальненої змінної.