Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

**про виконання лабораторних та практичних робіт блоку № 6**

На тему: «Динамічні структури (Черга, Стек, Списки, Дерево). Алгоритми обробки динамічних структур.»

***з дисципліни:*** «Мови та парадигми програмування»

до:

ВНС Лабораторної Роботи № 10

Алготестер Лабораторної Роботи № 5

Алготестер Лабораторної Роботи № 7-8

Практичних Робіт № 6

**Виконала:**

Студентка групи ШІ-11

Саліщева Юлія Августівна

# **Тема роботи:**

Ознайомитись з динамічними структурами: стеком, списками та деревами, зрозуміти в чому відмінність, попрактикуватися. Написати коди для практичних і лабораторних робіт з ВНС, а також задач з Algotester, результати закинути на github.

# **Мета роботи:**

* Theory Education Activities
* Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)
* Lab# programming: VNS Lab 10
* Lab# programming: Algotester Lab 5
* Lab# programming: Algotester Lab 7-8
* Practice# programming: Class Practice Task
* Practice# programming: Self Practice Task
* Result Documentation Report and Outcomes Placement Activities (Docs and Programs on GitHub)
* Results Evaluation and Release

# **Теоретичні відомості:**

1. Індивідуальний план опрацювання теорії:

* Тема №1: Theory Education Activities
  + Джерела Інформації:
  + https://prometheus.org.ua/cs50/sections/section6.html
    - Теорія з книжки С++ Primer 5th edition
    - Теоретичні відомості з лабораторних ВНС №10
  + Що опрацьовано:
    - Дізналась що таке черга, стек, списки та дерева та в чому їхня відмінність та їхні особливості.
    - Дізналась різні алгоритми обробки динамічних структур.
  + Статус: Ознайомлена
  + Початок опрацювання теми: 06.12
  + Звершення опрацювання теми: 08.12

# **Виконання роботи:**

## **1. Опрацювання завдання та вимог до програм та середовища:**

Завдання №1: VNS Lab 10

* Варіант завдання: 8
* Деталі завдання: Написати програму, у якій створюються динамічні структури й виконати їхню обробку у відповідності зі своїм варіантом.

Для кожного варіанту розробити такі функції:

1. Створення списку.

2. Додавання елемента в список (у відповідності зі своїм варіантом).

3. Знищення елемента зі списку (у відповідності зі своїм варіантом).

4. Друк списку.

5. Запис списку у файл.

6. Знищення списку.

7. Відновлення списку з файлу.

Порядок виконання роботи:

1. Написати функцію для створення списку. Функція може створювати порожній список, а потім додавати в нього елементи.

2. Написати функцію для друку списку. Функція повинна передбачати вивід повідомлення, якщо список порожній.

3. Написати функції для знищення й додавання елементів списку у

відповідності зі своїм варіантом.

4. Виконати зміни в списку й друк списку після кожної зміни.

5. Написати функцію для запису списку у файл.

6. Написати функцію для знищення списку.

7. Записати список у файл, знищити його й виконати друк (при друці повинне бути видане повідомлення "Список порожній").

8. Написати функцію для відновлення списку з файлу.

9. Відновити список і роздрукувати його.

10.Знищити список.

13.Записи в лінійному списку містять ключове поле типу \*char (рядок

символів). Сформувати двонаправлений список. Знищити з нього К перших

елементів. Додати елемент після елемента, що починається із зазначеного

символу.

Завдання №2: Algotester Lab 5

* Варіант завдання: 3
* Деталі завдання: У вас є карта гори розміром N × M. Також ви знаєте координати {x, y} , у яких знаходиться вершина гори. Ваше завдання - розмалювати карту таким чином, щоб найнижча точка мала число 0, а пік гори мав найбільше число. Клiтинки якi мають суміжну сторону з вершиною мають висоту на один меншу, сумiжнi з ними i не розфарбовані мають ще на 1 меншу висоту i так далi.

Завдання №3: Algotester Lab 78

* Варіант завдання: 2
* Деталі завдання: Ваше завдання - власноруч реалізувати структуру даних "Динамічний масив".

Ви отримаєте Q запитів, кожен запит буде починатися зi слова-iдентифiкатора, пiсля якого йдуть його аргументи. Вам будуть поступати запити такого типу:

• Вставка:

Iдентифiкатор - insert

Ви отримуєте ціле число index елемента, на місце якого робити вставку.

Пiсля цього в наступному рядку рядку написане число N - розмір масиву, який треба вставити. У третьому рядку N цілих чисел - масив, який треба вставити на позицію index.

• Видалення:

Iдентифiкатор - erase

Ви отримуєте 2 цілих числа - index, індекс елемента, з якого почати видалення та n - кiлькiсть елементiв, яку треба видалити.

• Визначення розмiру:

Iдентифiкатор - size

Ви не отримаєте аргументів. Ви виводите кiлькiсть елементiв у динамічному масиві.

• Визначення кiлькостi зарезервованої пам'яті:

Iдентифiкатор - capacity

Ви не отримаєте аргументів.Ви виводите кількість зарезервованої пам'яті у динамічному масиві. Ваша реалiзацiя динамiчного масиву має мати фактор росту (Growth factor) рiвний 2.

• Отримання значення i-го елементу

Iдентифiкатор - get

Ви отримуєте ціле число - index, індекс елемента. Ви виводите значення елемента за індексом. Реалізувати використовуючи перегрузку оператора []

• Модифiкацiя значення i-го елементу

Iдентифiкатор - set

Ви отримуєте 2 цілих числа - індекс елемента, який треба змінити, та його нове значення. Реалізувати використовуючи перегрузку оператора []

• Вивiд динамічного масиву на екран

Iдентифiкатор - print

Ви не отримуєте аргументів. Ви виводите всi елементи динамічного масиву через пробіл. Реалізувати використовуючи перегрузку оператора <<

Завдання №4: Class Practice Task

* Деталі завдання: Бінарні дерева:
* Задача №4 - Віддзеркалення дерева

- використовувати цілі числа для значень у вузлах дерева

- реалізувати функцію, що проходить по всіх вузлах дерева і міняє місцями праву і ліву вітки дерева

- функція повертає нове дерево, передане в функцію дерево не модифікується

## Задача №5 - Записати кожному батьківському вузлу суму підвузлів

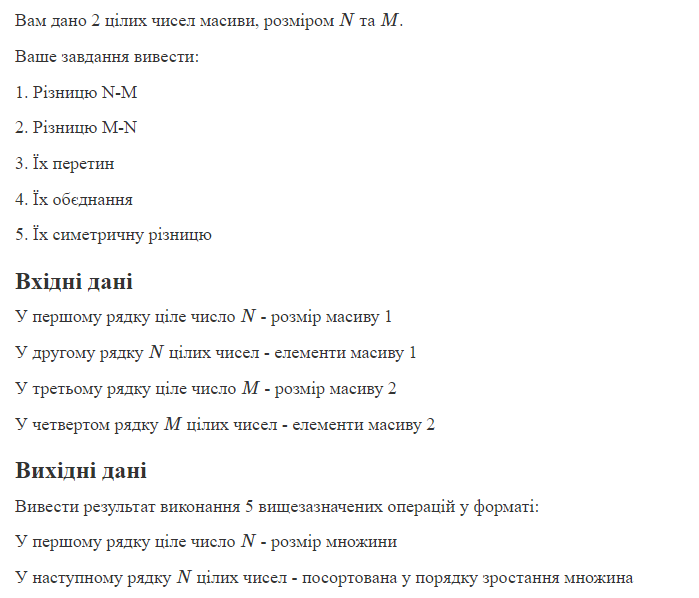
- використовувати цілочисельні значення у вузлах дерева;

- реалізувати функцію, яка ітеративно проходить по бінарному дереві і записує у батьківський вузол суму значень підвузлів

- вузол-листок не змінює значення

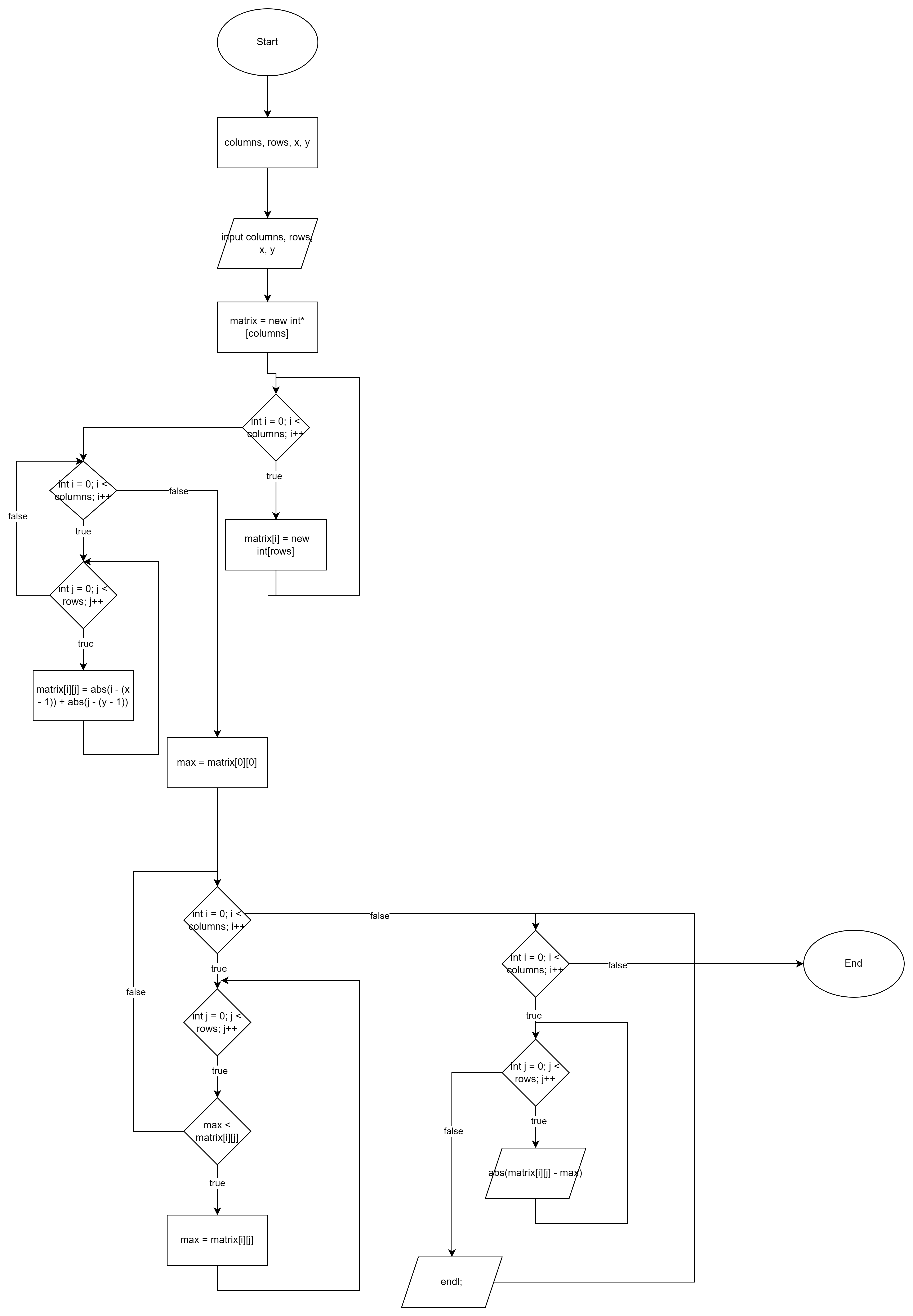
- значення змінюються від листків до кореня дерева

Завдання №5: Self Practice Task

* 

## **2. Дизайн та планована оцінка часу виконання завдань:**

Програма №5 : Algotester lab5v3

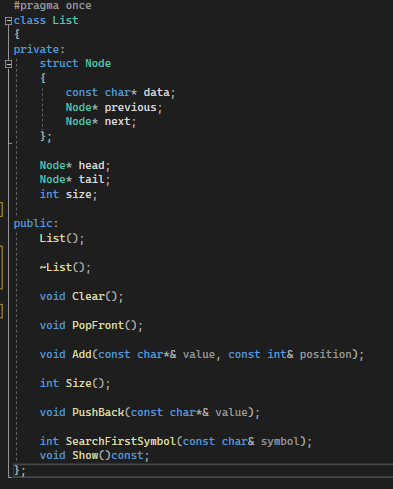


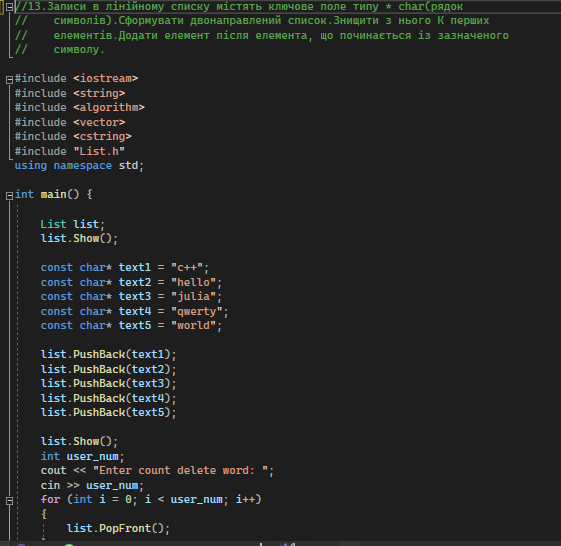
*Блоксхема до завдання. 1: Algotester lab5v3*

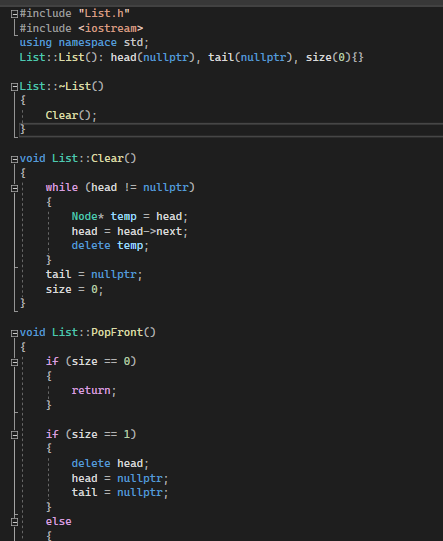
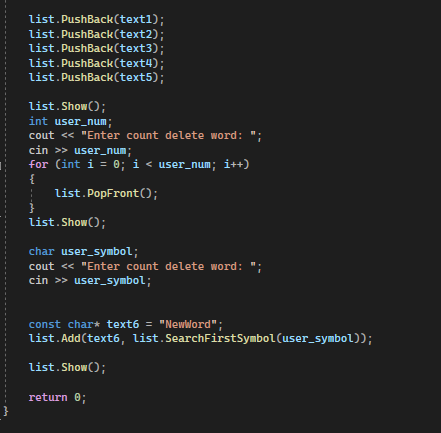
* Планований час на реалізацію: 20 хв

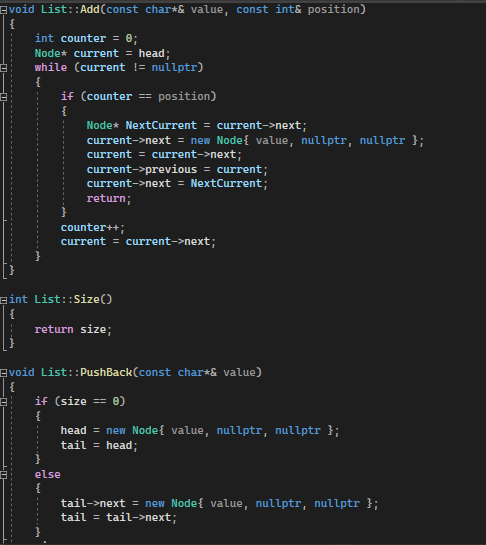
## **3. Код програм з посиланням на зовнішні ресурси:**

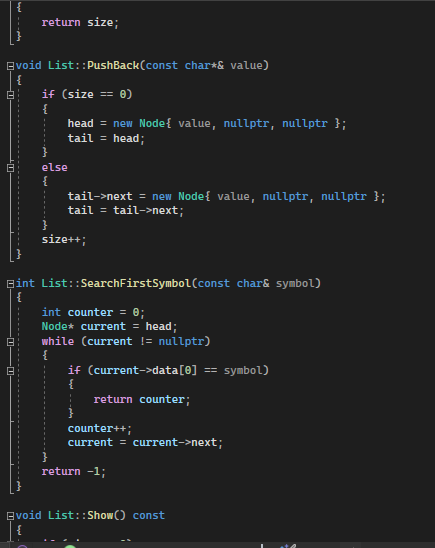
Завдання №1 VNS Lab 10

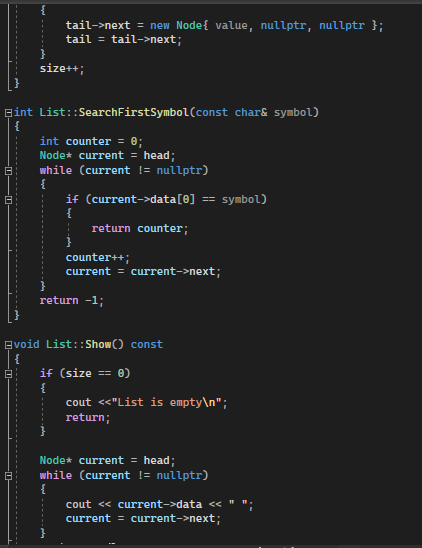


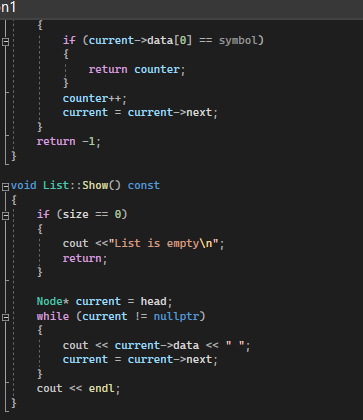


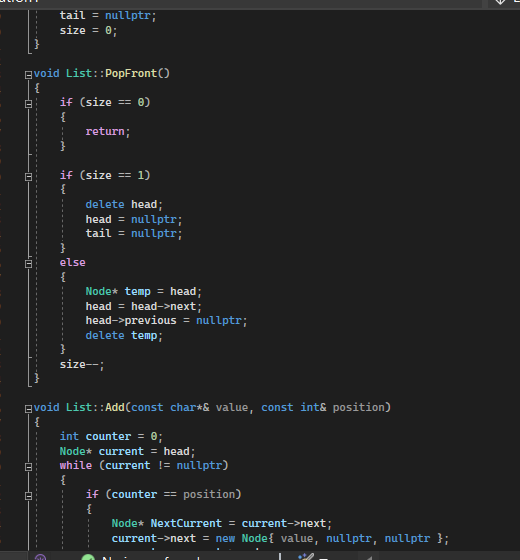






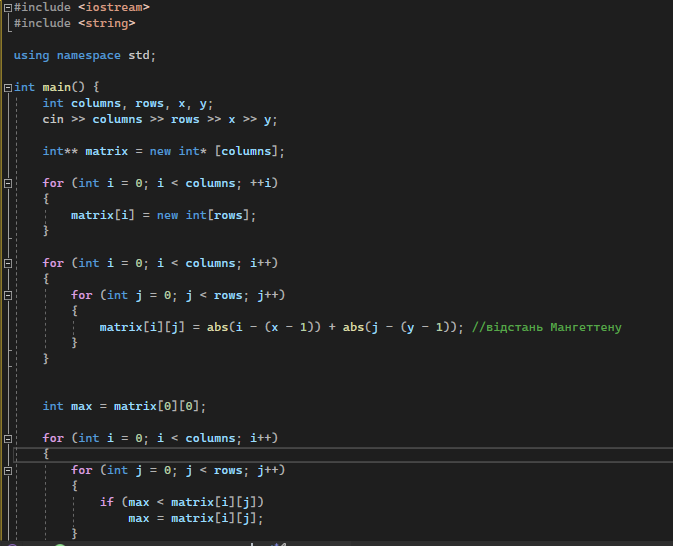


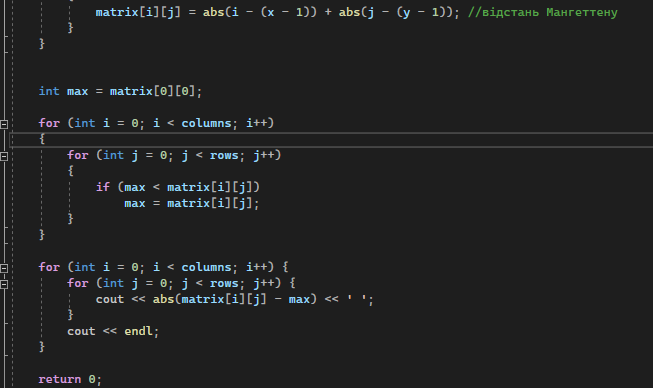




[ai\_programming\_playground/ai\_11/yuliia\_salishcheva/epic\_6/VNS\_lab10 at epic\_6\_practice\_and\_labs\_yuliia\_salishcheva · artificial-intelligence-department/ai\_programming\_playground (github.com)](https://github.com/artificial-intelligence-department/ai_programming_playground/tree/epic_6_practice_and_labs_yuliia_salishcheva/ai_11/yuliia_salishcheva/epic_6/VNS_lab10)

Завдання №2: Algotester Lab 5

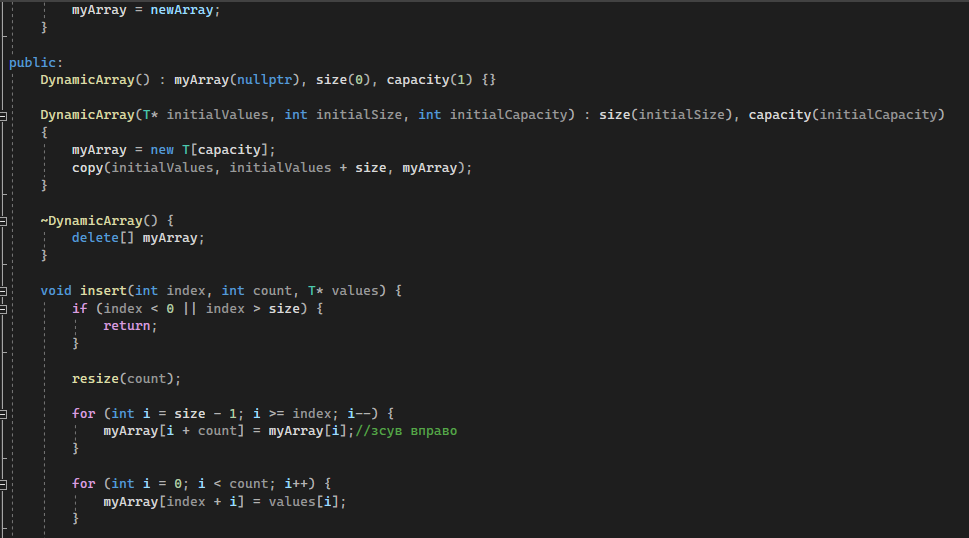


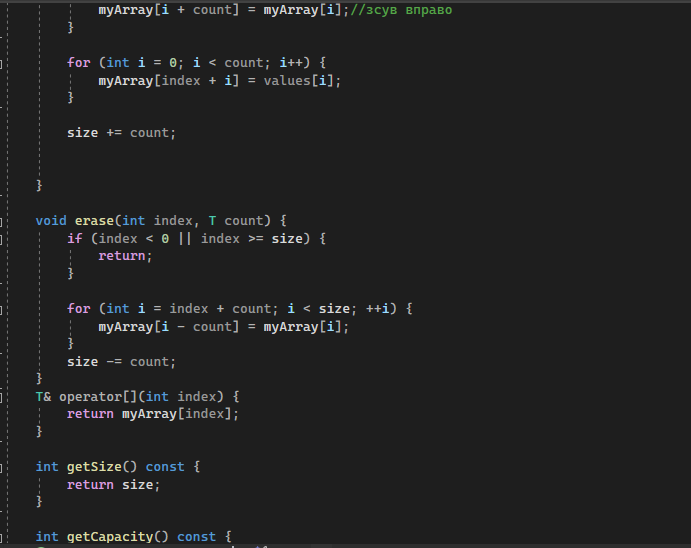


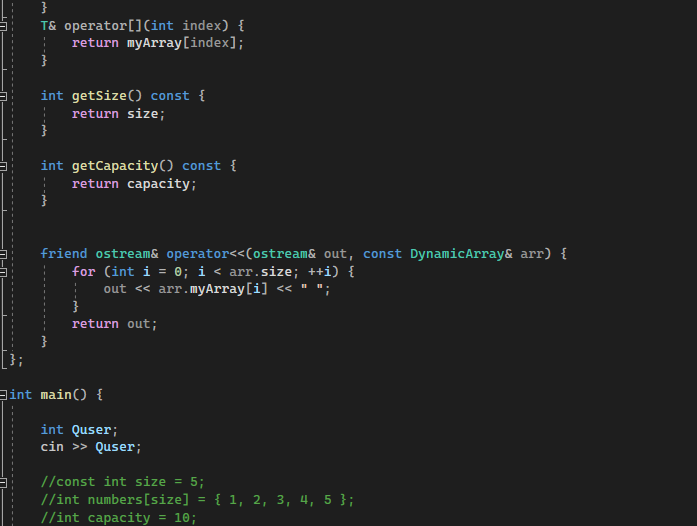
[ai\_programming\_playground/ai\_11/yuliia\_salishcheva/epic\_6/algotester\_lab\_5\_variant\_3\_yuliia\_salishcheva.cpp at epic\_6\_practice\_and\_labs\_yuliia\_salishcheva · artificial-intelligence-department/ai\_programming\_playground (github.com)](https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_6_practice_and_labs_yuliia_salishcheva/ai_11/yuliia_salishcheva/epic_6/algotester_lab_5_variant_3_yuliia_salishcheva.cpp)

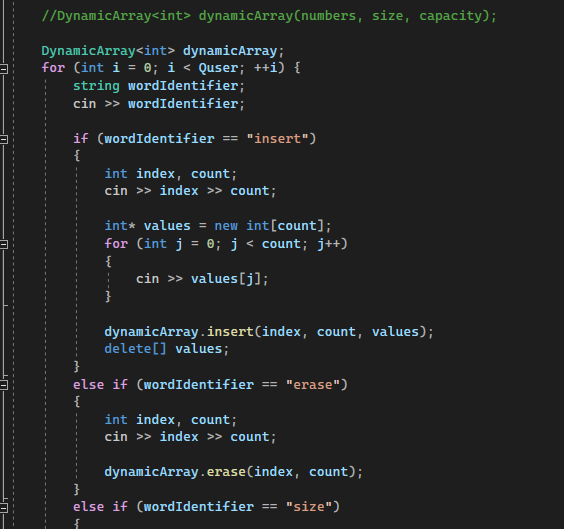
Завдання №3: Algotester Lab 78

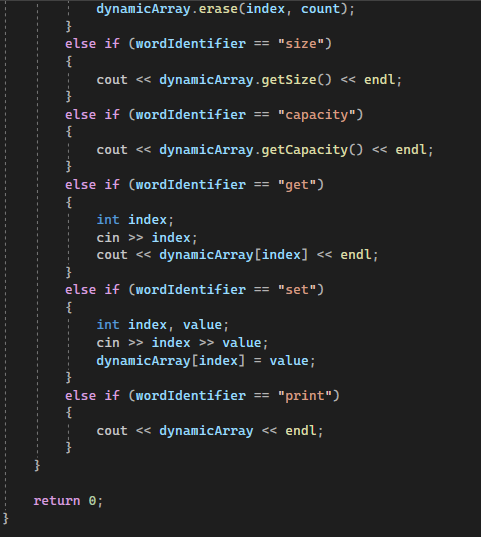






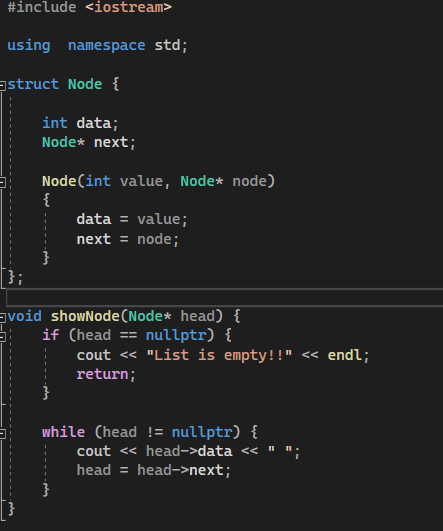


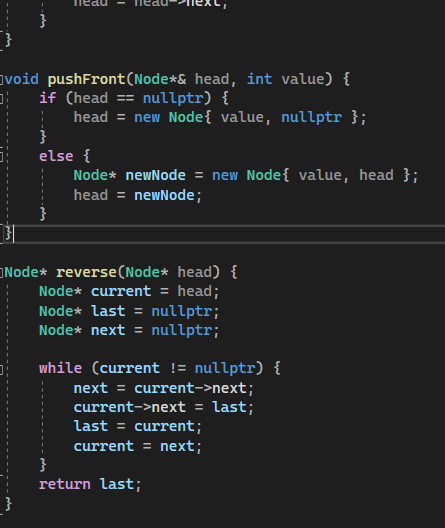


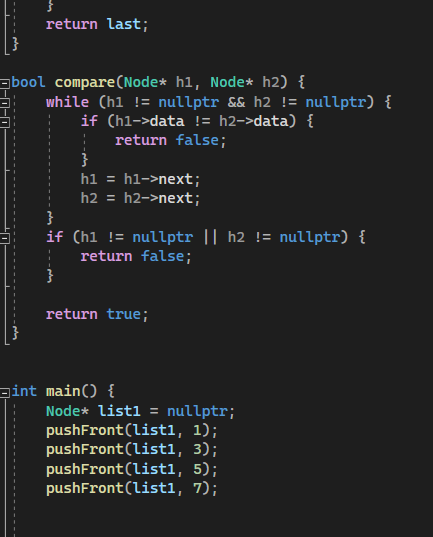


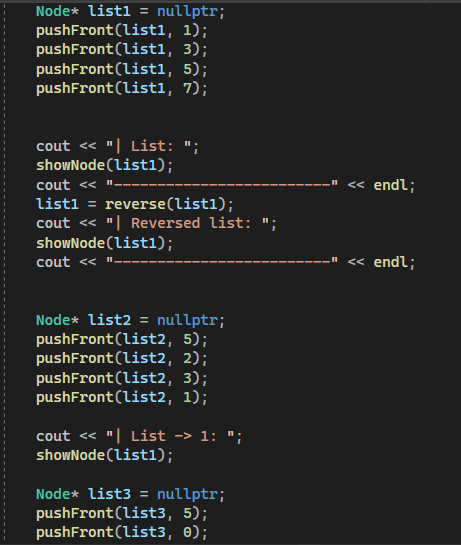
[ai\_programming\_playground/ai\_11/yuliia\_salishcheva/epic\_6/algotester\_lab\_7-8\_variant\_2\_yuliia\_salishcheva.cpp at epic\_6\_practice\_and\_labs\_yuliia\_salishcheva · artificial-intelligence-department/ai\_programming\_playground (github.com)](https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_6_practice_and_labs_yuliia_salishcheva/ai_11/yuliia_salishcheva/epic_6/algotester_lab_7-8_variant_2_yuliia_salishcheva.cpp)

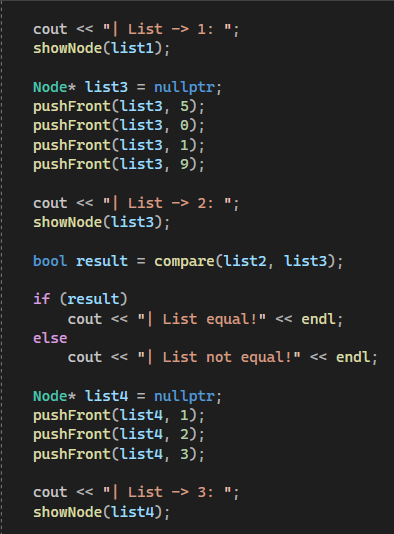
Завдання №4: Class Practice Task

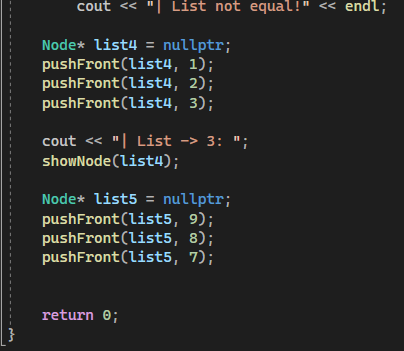


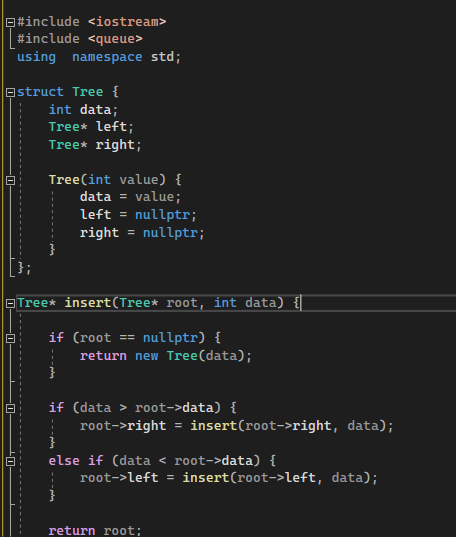


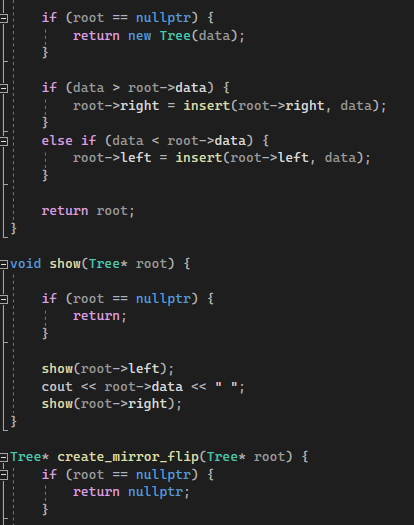


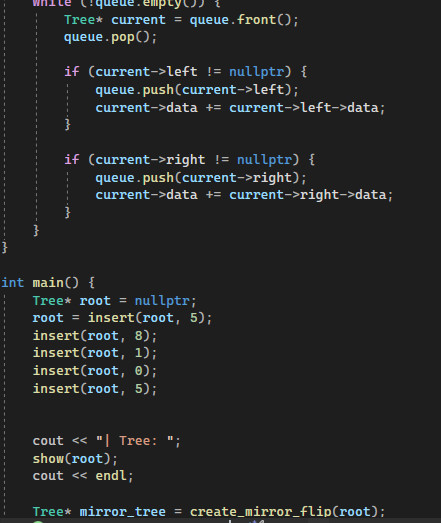


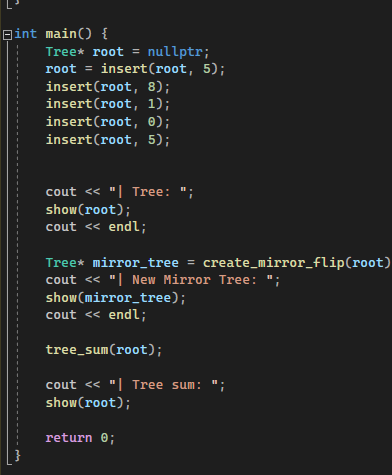






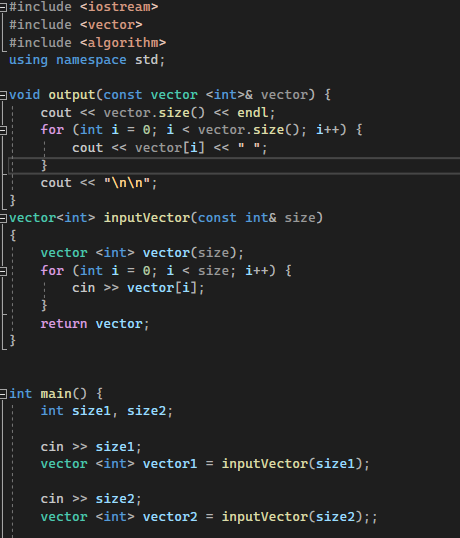


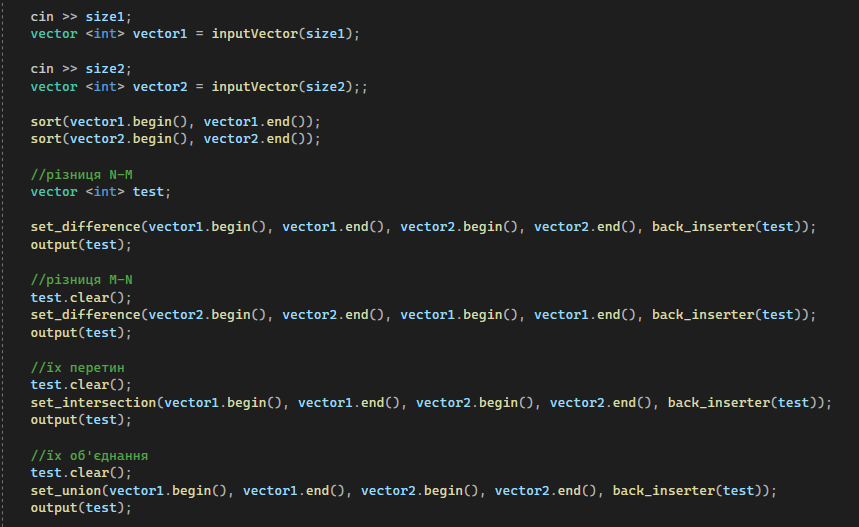


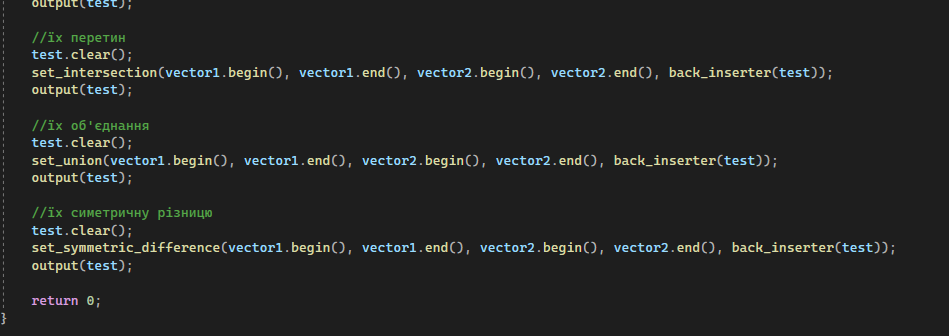


[ai\_programming\_playground/ai\_11/yuliia\_salishcheva/epic\_6/practice\_work\_task\_1\_yuliia\_salishcheva.cpp at epic\_6\_practice\_and\_labs\_yuliia\_salishcheva · artificial-intelligence-department/ai\_programming\_playground (github.com)](https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_6_practice_and_labs_yuliia_salishcheva/ai_11/yuliia_salishcheva/epic_6/practice_work_task_1_yuliia_salishcheva.cpp)

Завдання №5: Self Practice Task

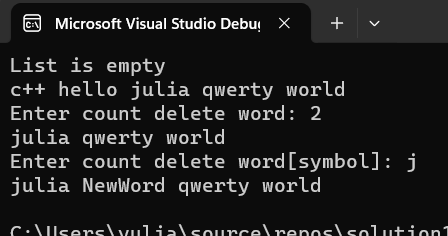






[ai\_programming\_playground/ai\_11/yuliia\_salishcheva/epic\_6/self\_practice\_work\_algotester\_task\_1\_yuliia\_salishcheva.cpp at epic\_6\_practice\_and\_labs\_yuliia\_salishcheva · artificial-intelligence-department/ai\_programming\_playground (github.com)](https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_6_practice_and_labs_yuliia_salishcheva/ai_11/yuliia_salishcheva/epic_6/self_practice_work_algotester_task_1_yuliia_salishcheva.cpp)

## **4. Результати виконання завдань, тестування та фактично затрачений час:**

Завдання №1:VNS Lab 10

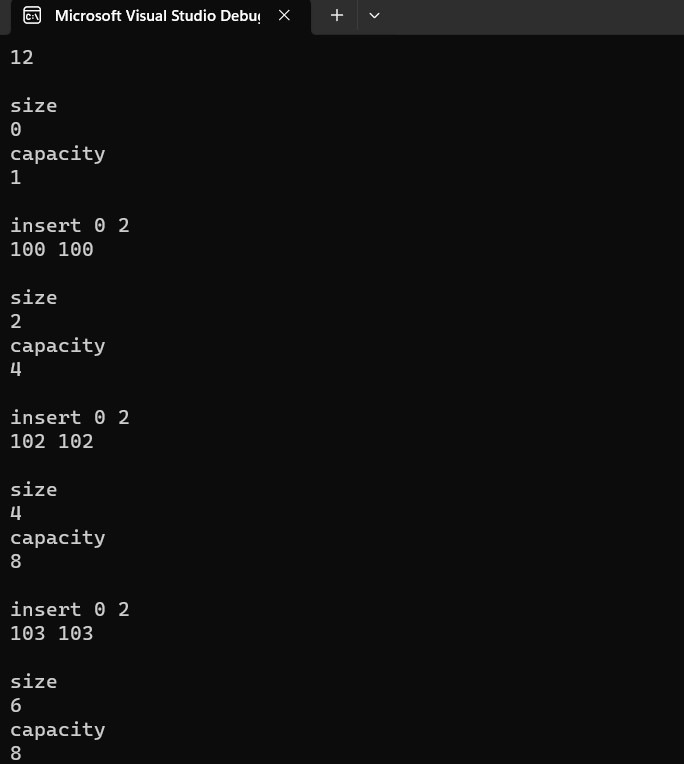
Час затрачений на виконання завдання: 1 год.

Завдання №2: Algotester Lab 5



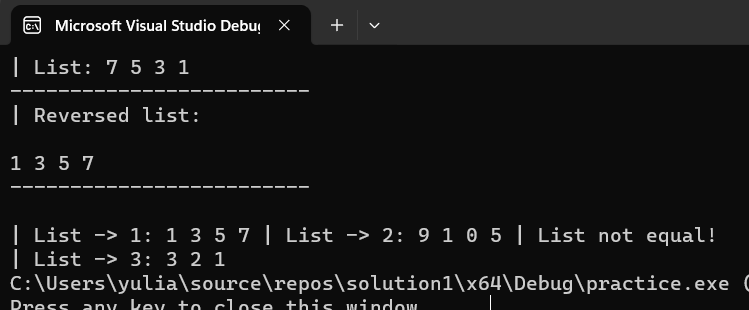
Час затрачений на виконання завдання: 1 год

Завдання №3: Algotester Lab 78



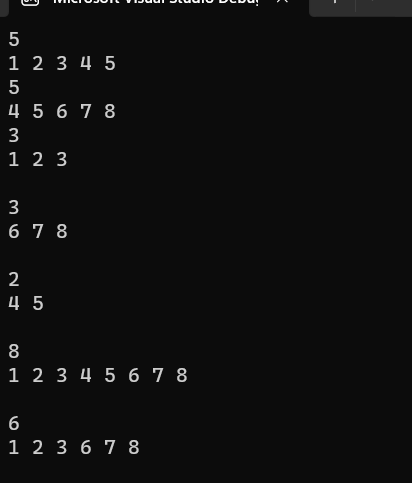
Час затрачений на виконання завдання: 2 год

Завдання №4: Class Practice Task



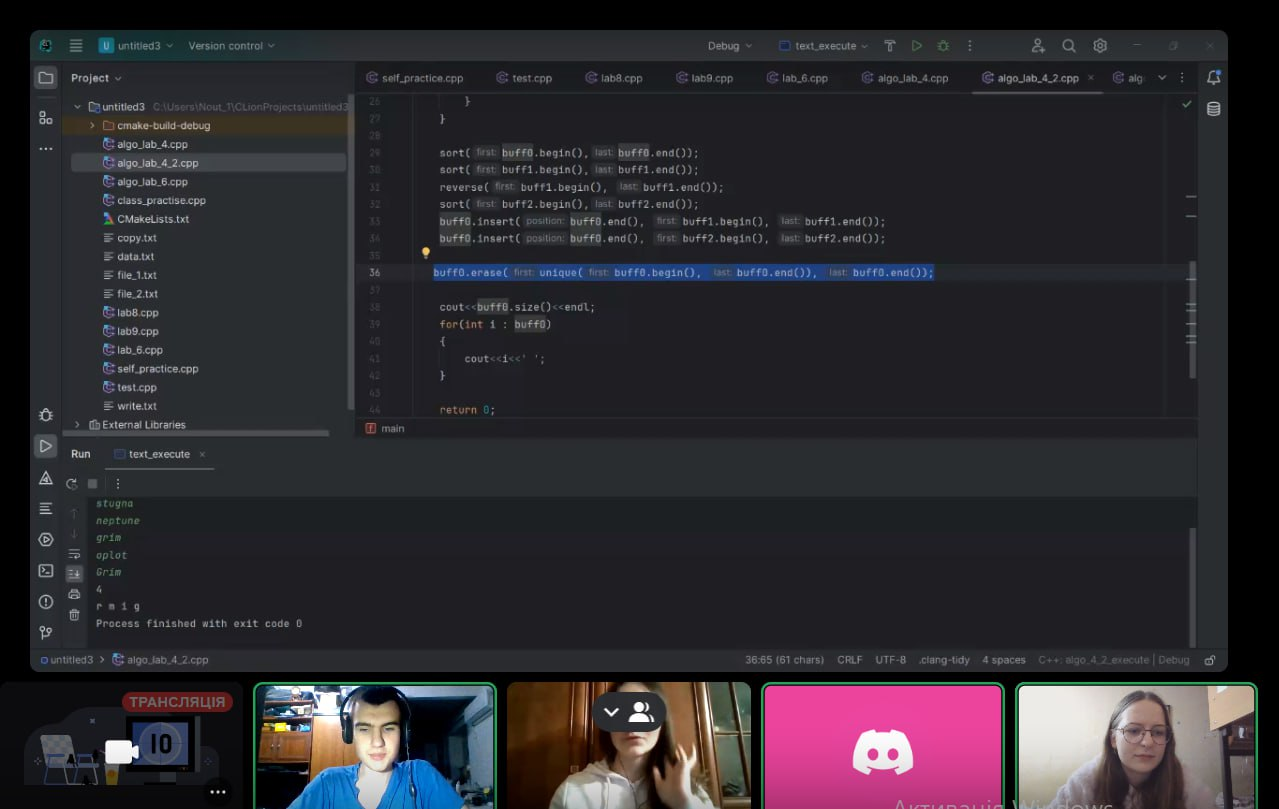
Час затрачений на виконання завдання: 1.5 год

Завдання №5: Self Practice Task



Час затрачений на виконання завдання: 1 год

## **5.** **Командна робота та комунікація:**



*Зустріч у діскорді.*

# **Висновки:**

# Протягом опрацювання цього епіку я вивчила новий матеріал, який стосується динамічних структур: списки, дерева, черга та стек.Також дізналась про їхні відмінності, ситуації їхнього використання та алгоритми їхньої обробки. Ці навички я застосувала у виконанні лабортаорних робіт. Всі файли включно зі звітом запушила на гітхаб.