Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

**про виконання лабораторних та практичних робіт блоку № 4**

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

до:

Практичних Робіт до блоку № 1

**Виконала:**

Студентка групи ШІ-12

Костак Олеся Михайлівна

Львів – 2024 р.

**Тема роботи:**Динамічні структури (Черга, Стек, Списки, Дерево). Алгоритми обробки динамічних структур.

**Мета роботи:** Освоїти роботу з динамічними структурами даних, такими як черга, стек, списки та дерева. Навчитися реалізовувати базові операції над цими структурами, включаючи створення, додавання, видалення, пошук та обхід елементів. Розвинути навички алгоритмічної обробки динамічних структур для ефективного вирішення прикладних задач.

**Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №\*.1: Основи Динамічних Структур Даних.
* Тема №\*.2: Стек.
* Тема №\*.3: Черга.
* Тема №\*.4: Зв'язні Списки.
* Тема №\*.5: Дерева.
* Тема №\*.6: Алгоритми Обробки Динамічних Структур.

1. Індивідуальний план опрацювання теорії:

*Джерела Інформації*:

* 1. Сайт geeksforgeeks
  2. Сайт programiz.com
  3. Сайт w3schools.com
  4. Ютуб-канал Блоган
  5. Лекції Пшеничного
  6. Сайт cppreference.com

Тема №\*.1: **Основи Динамічних Структур Даних.**

* + *Що опрацьовано:*
    - Стаття [Dynamic Data Structures](https://www.dcs.bbk.ac.uk/~roger/cpp/week9.htm)

Тема №\*.2: **Стек**.

* + *Що опрацьовано:*
    - Відео [C++ ⦁ Теорія ⦁ Урок 141 ⦁ ADT ⦁ Стек](https://www.youtube.com/watch?v=ZYvYISxaNL0&t=909s&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD)
* Тема №\*.3: **Черга.**
  + *Що опрацьовано*
    - Стаття [Introduction to Queue Data Structure](https://www.geeksforgeeks.org/introduction-to-queue-data-structure-and-algorithm-tutorials/)
* Тема №\*.4: **Зв'язні Списки.**
  + *Що опрацьовано:*
    - Відео [C++ ⦁ Теорія ⦁ Урок 139 ⦁ ADT ⦁ Однозв'язний список](https://www.youtube.com/watch?v=-25REjF_atI&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD)
    - Стаття [Delete a Linked List node at a given position](https://www.geeksforgeeks.org/delete-a-linked-list-node-at-a-given-position/)

Тема №\*.5: **Дерева.**

* + *Що опрацьовано:*
    - Відео [C++ ⦁ Теорія ⦁ Урок 144 ⦁ ADT ⦁ Бінарне дерево](https://www.youtube.com/watch?v=qBFzNW0ALxQ&t=1144s&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD)
    - Стаття [Binary Tree](https://www.programiz.com/dsa/binary-tree)
* Тема №\*.6: **Алгоритми Обробки Динамічних Структур.**
  + *Що опрацьовано*:
    - [Searching Algorithms](https://www.geeksforgeeks.org/searching-algorithms/)

**Виконання роботи:**

Завдання **№1 VNS Lab 10 - Task 1**

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

* *Варіант завдання*: 5
* *Деталі завдання*:

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.

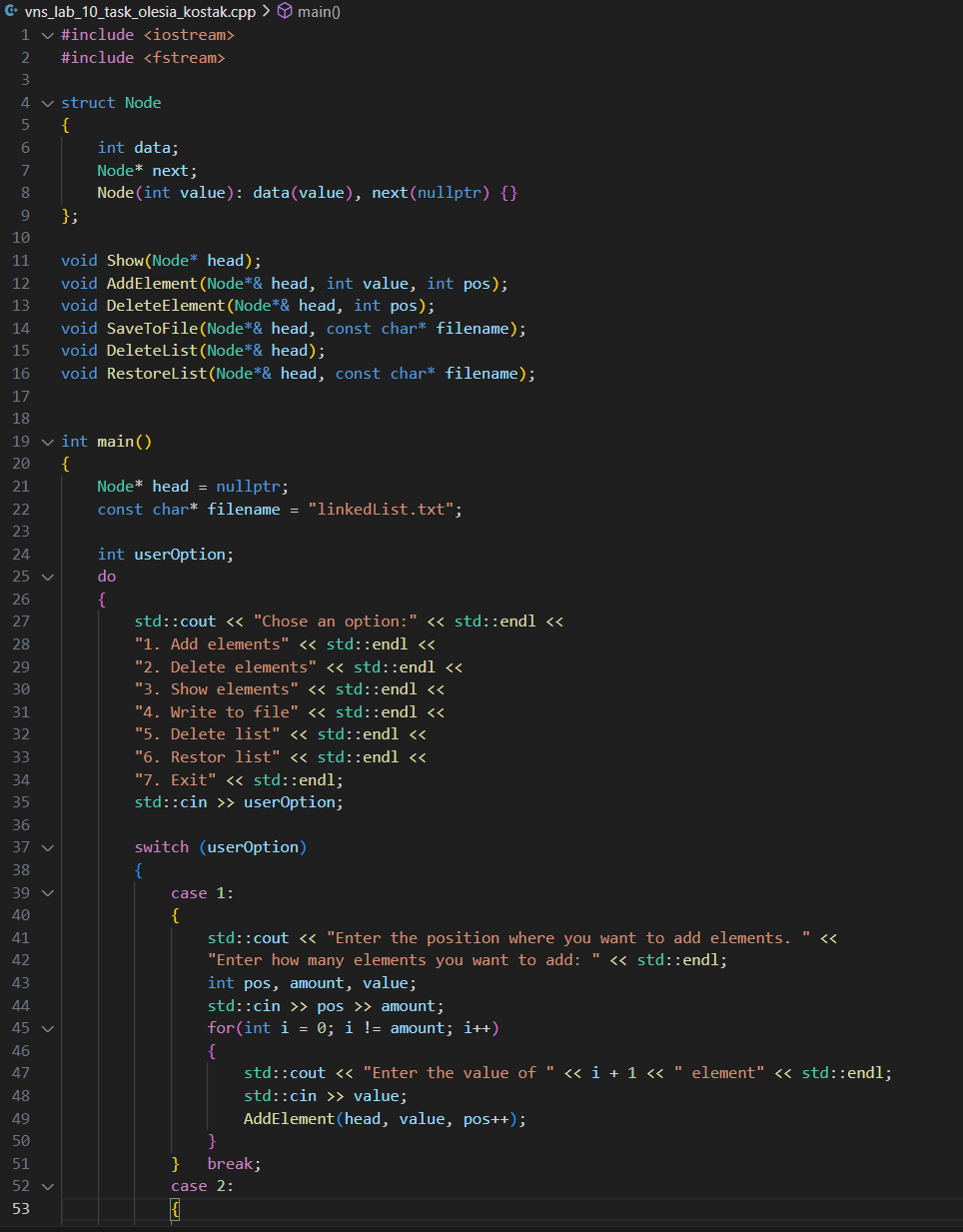
Відповідно до мого варіанту:

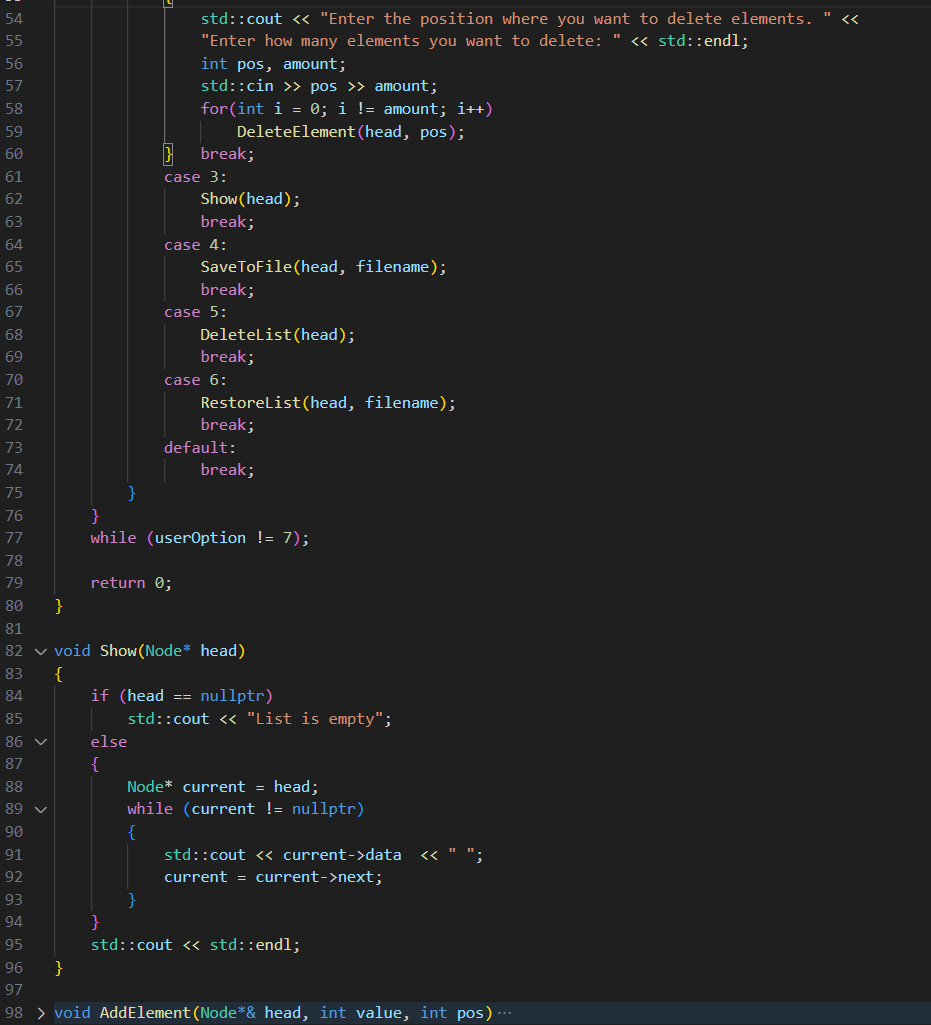
Записи в лінійному списку містять ключове поле типу int. Сформувати

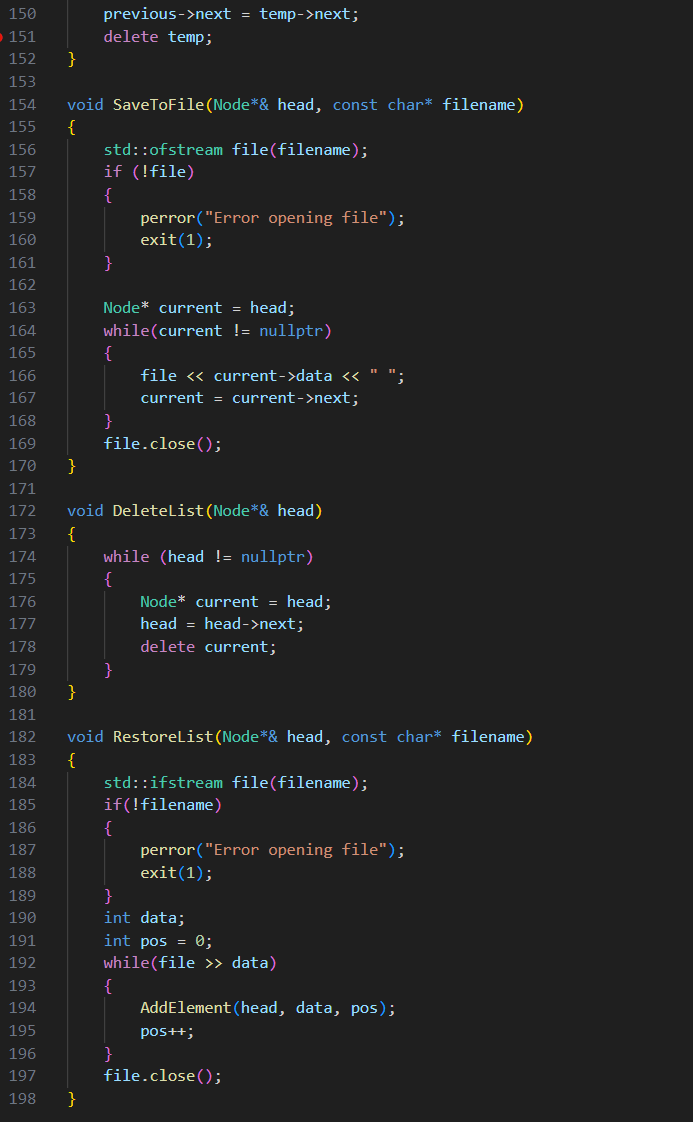
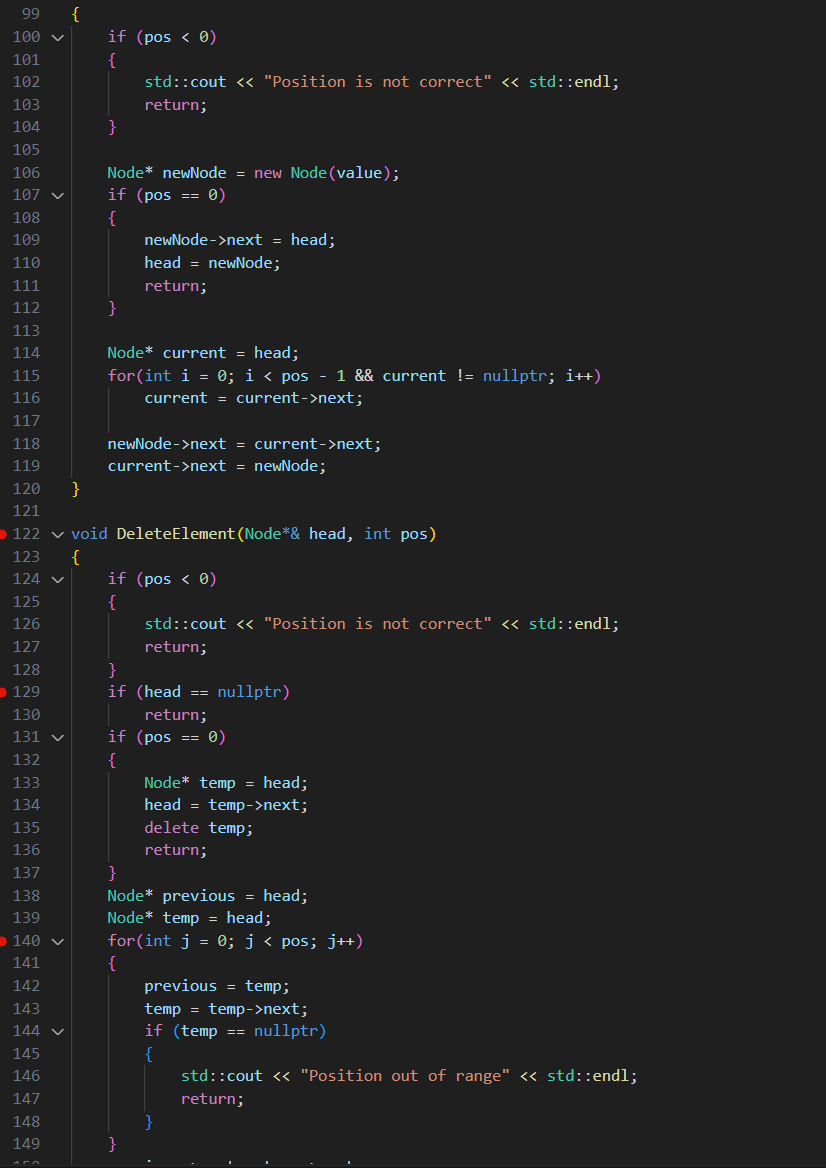
однонаправлений список. Знищити з нього К елементів, починаючи із

заданого номера, додати К елементів, починаючи із заданого номера;

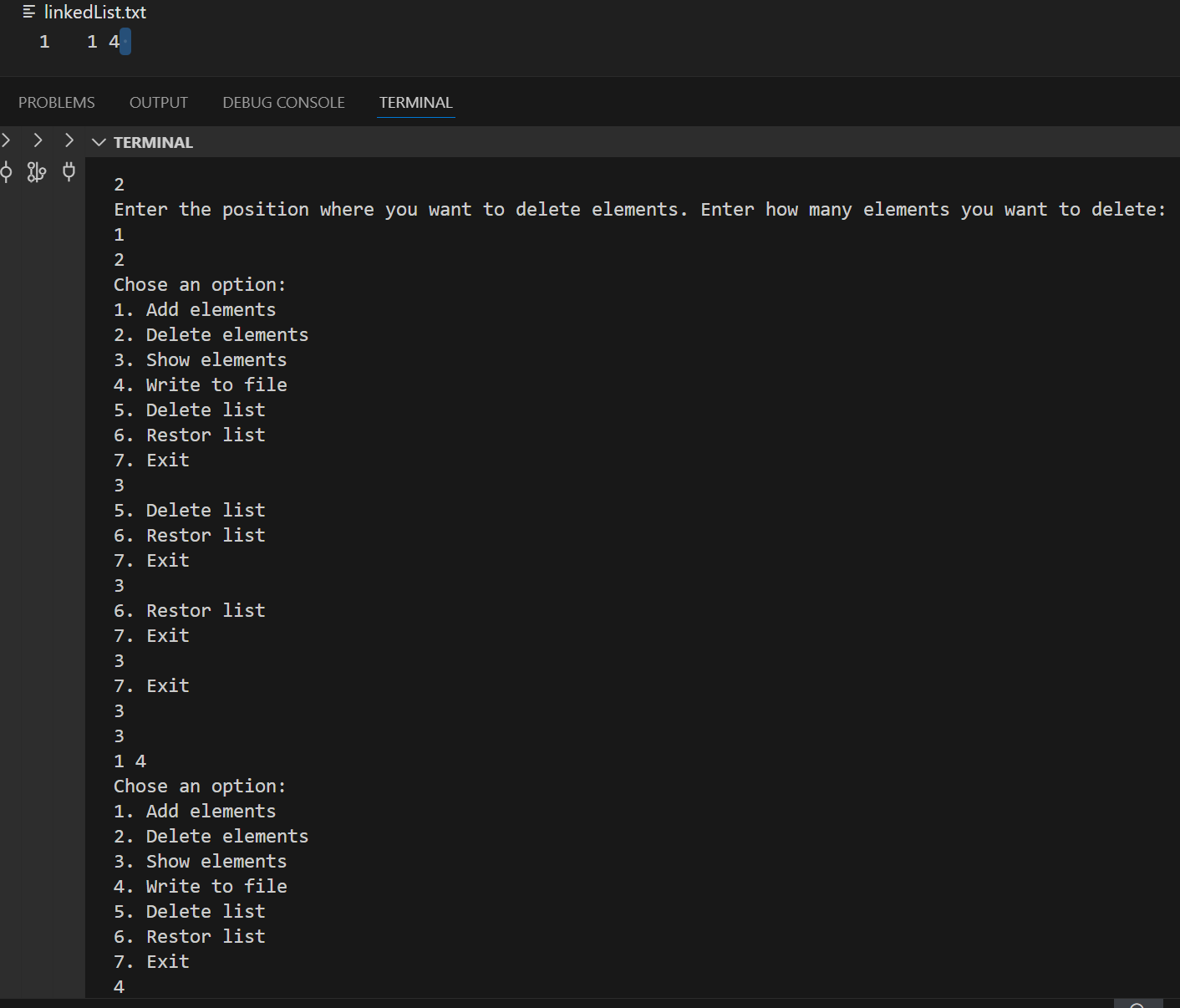
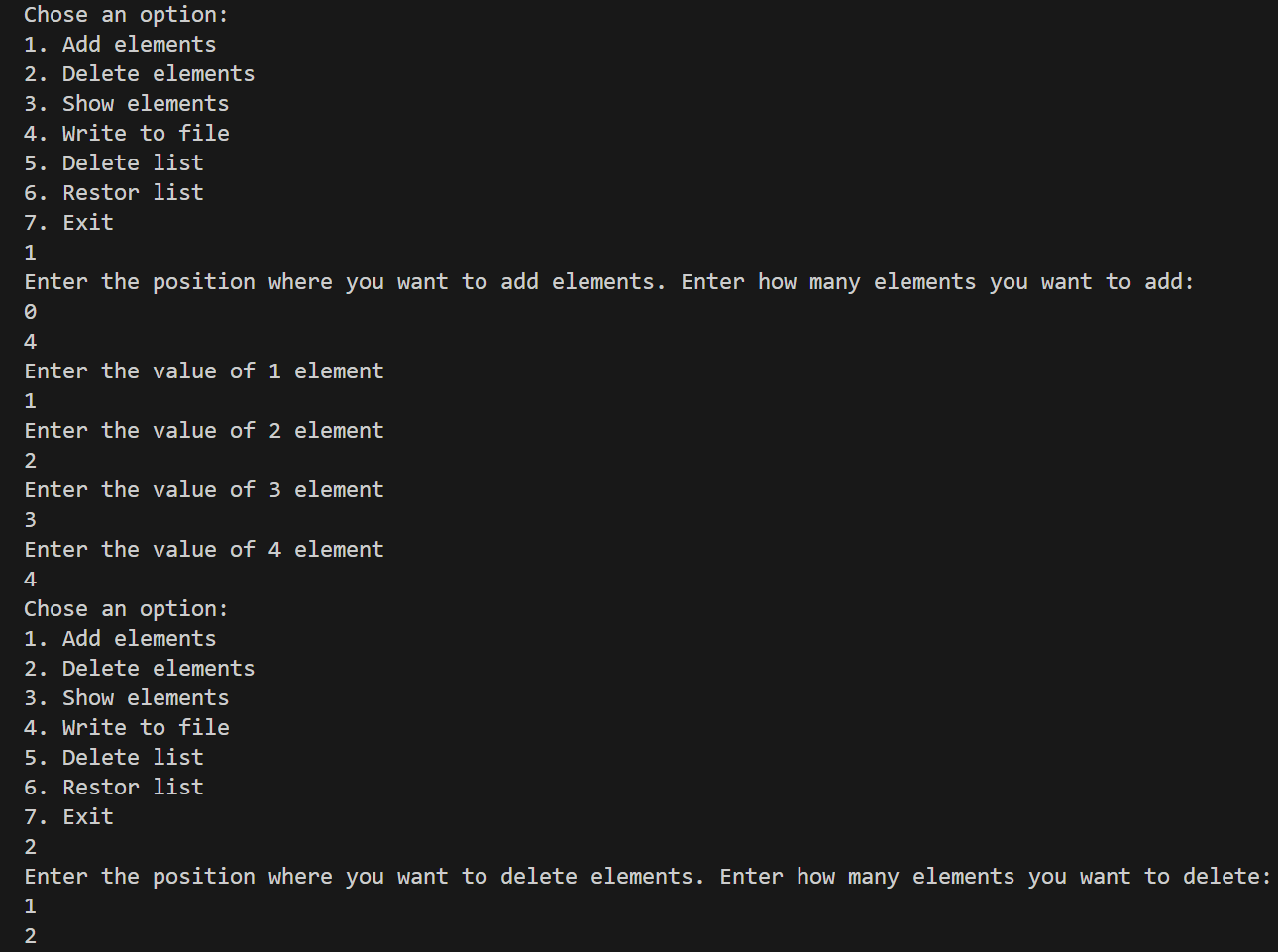
**2. Код програми**

****



****

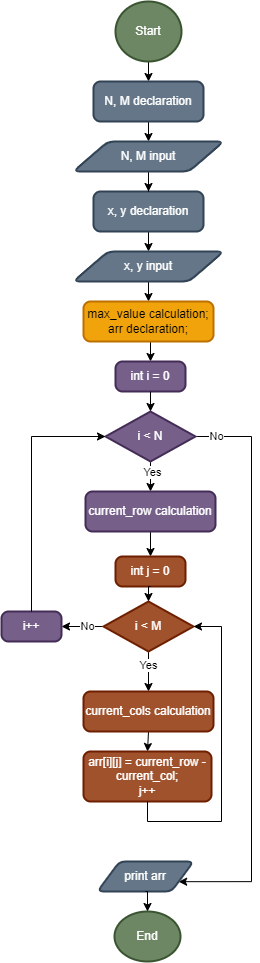
**Результати виконання завдань, тестування та фактично затрачений час:**



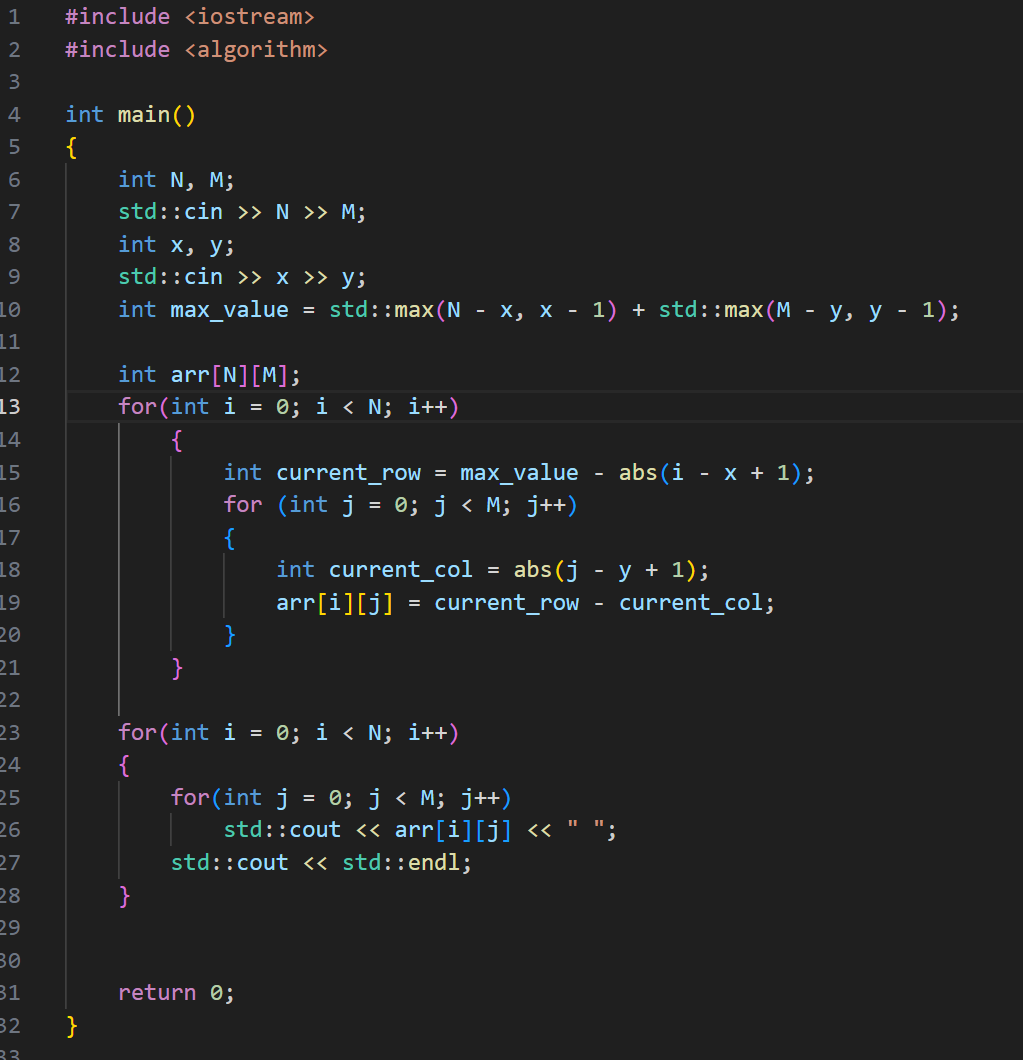
Час затрачений на виконання завдання: 4 год

Завдання **№2 Algotester Lab 5V3**

* *Варіант завдання*: 3
* [*Деталі завдання*](https://algotester.com/en/ContestProblem/DisplayWithEditor/135603)

**1. Діаграма:**

**2. Код програми:**

****

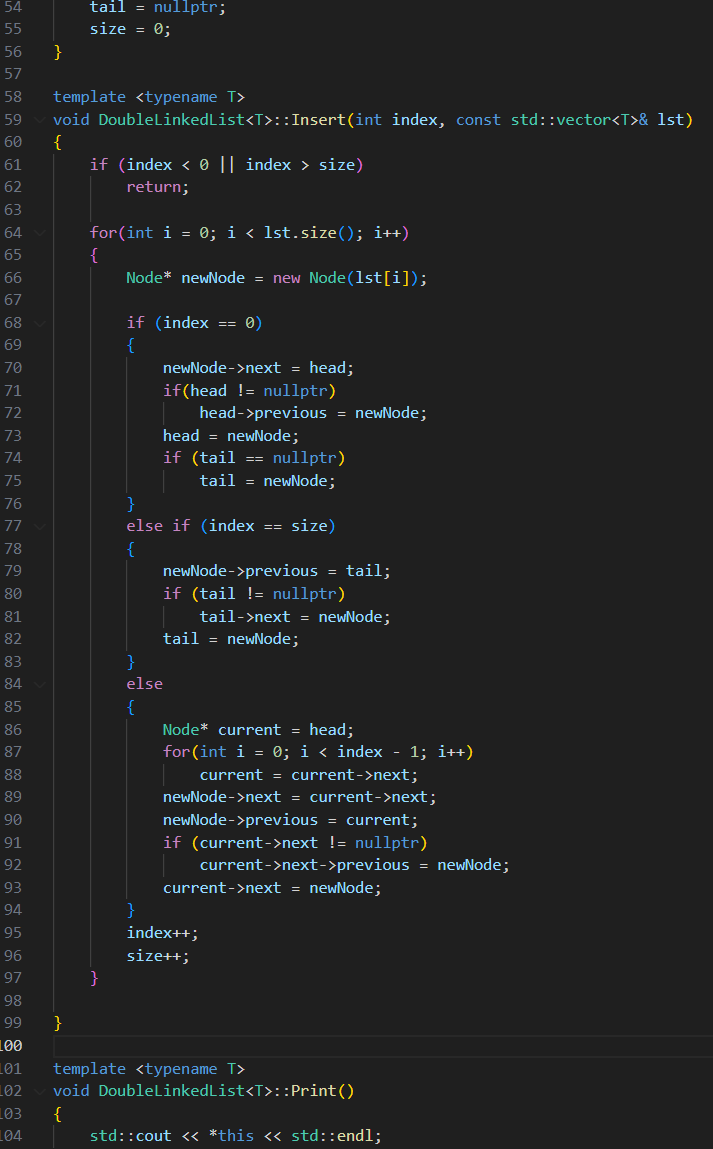
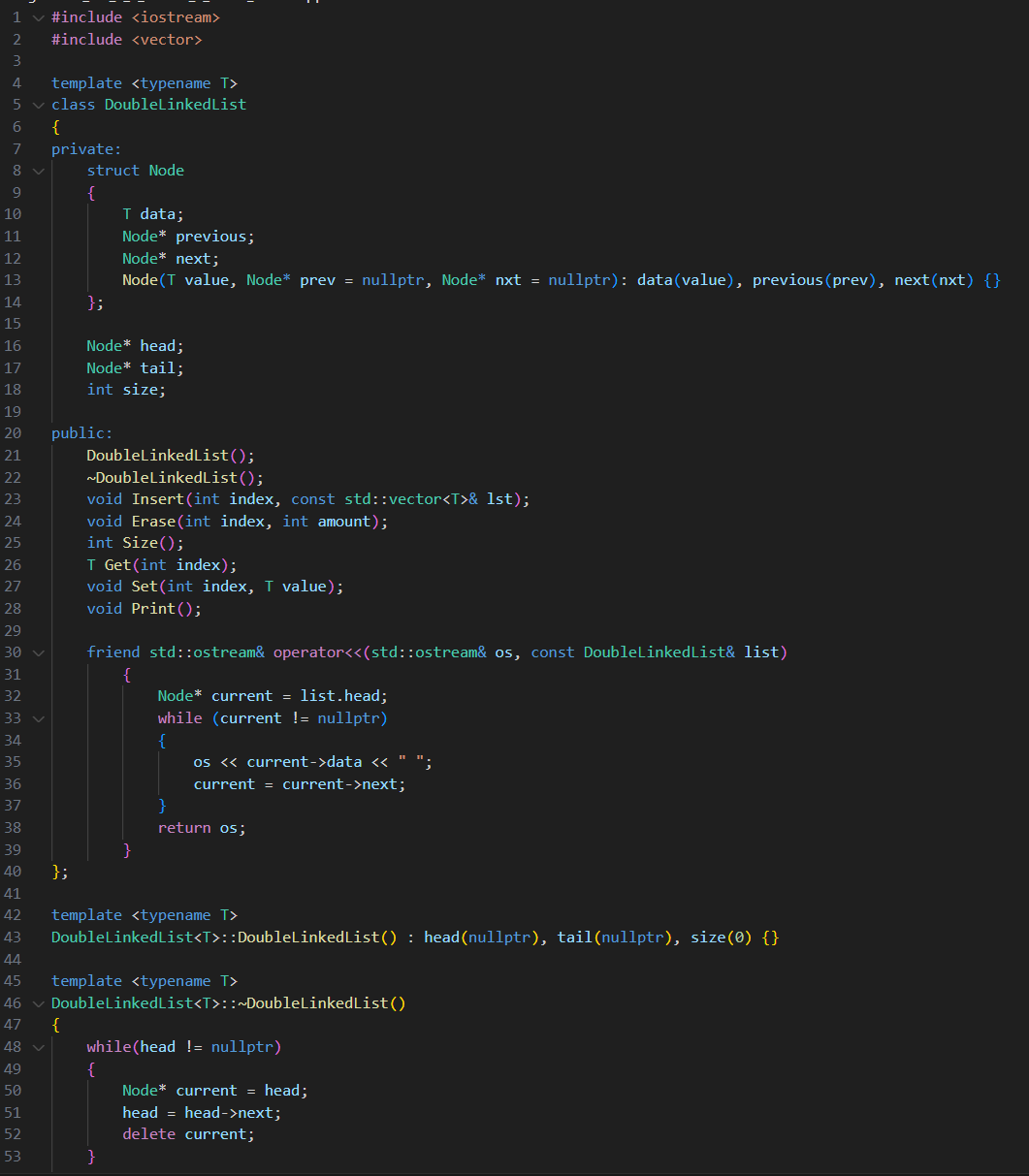
**3. Результат:**

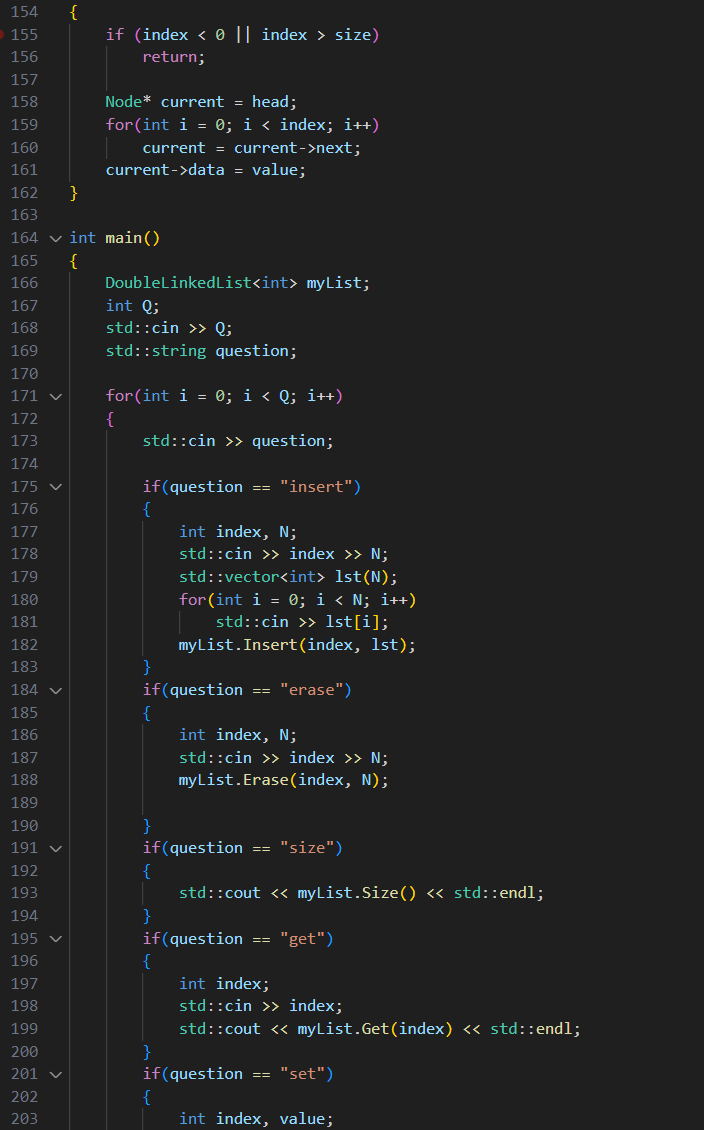
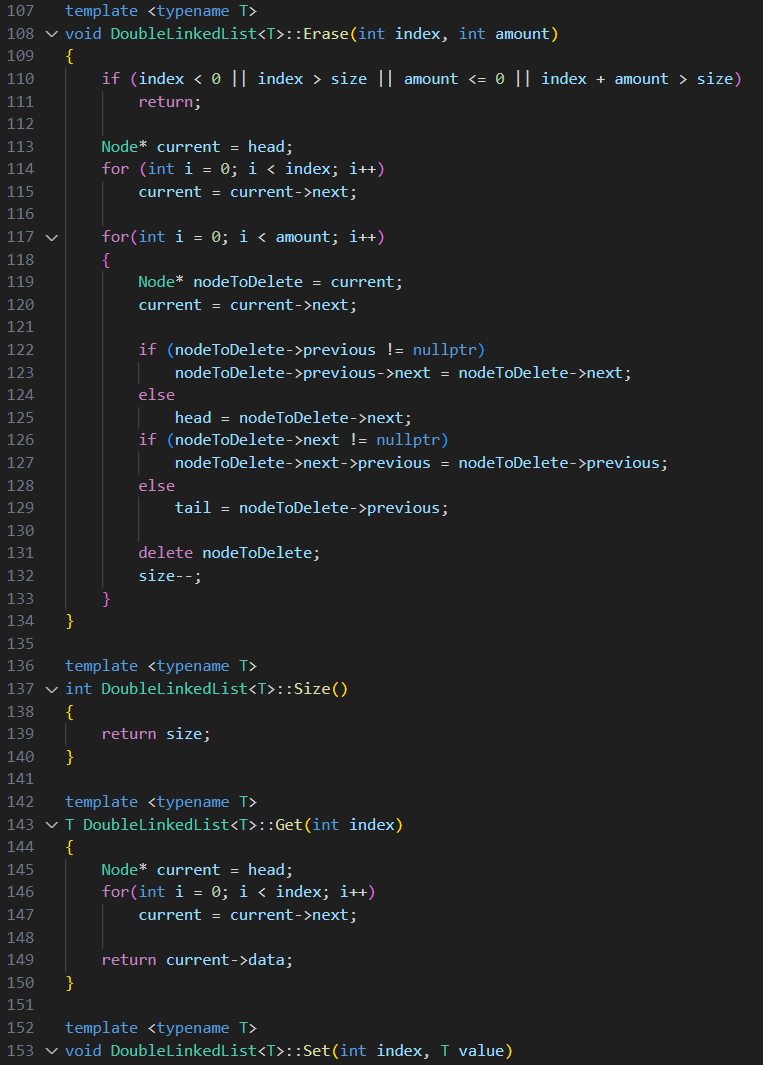


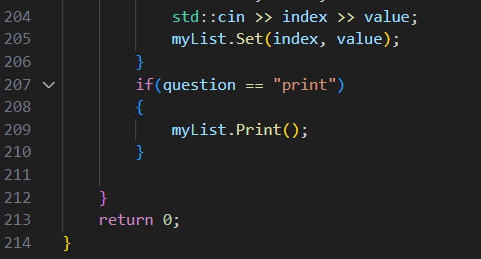
Завдання **№3 Algotester Lab 7-8**

* *Варіант завдання*: 1
* [*Деталі завдання*](https://algotester.com/en/ContestProblem/DisplayWithEditor/135607)

**1. Код програми:**

****



****

**2. Результат:**



Завдання **№5 Class Practice Work (1-3)**

* *Варіант завдання*: -
* *Деталі завдання*:

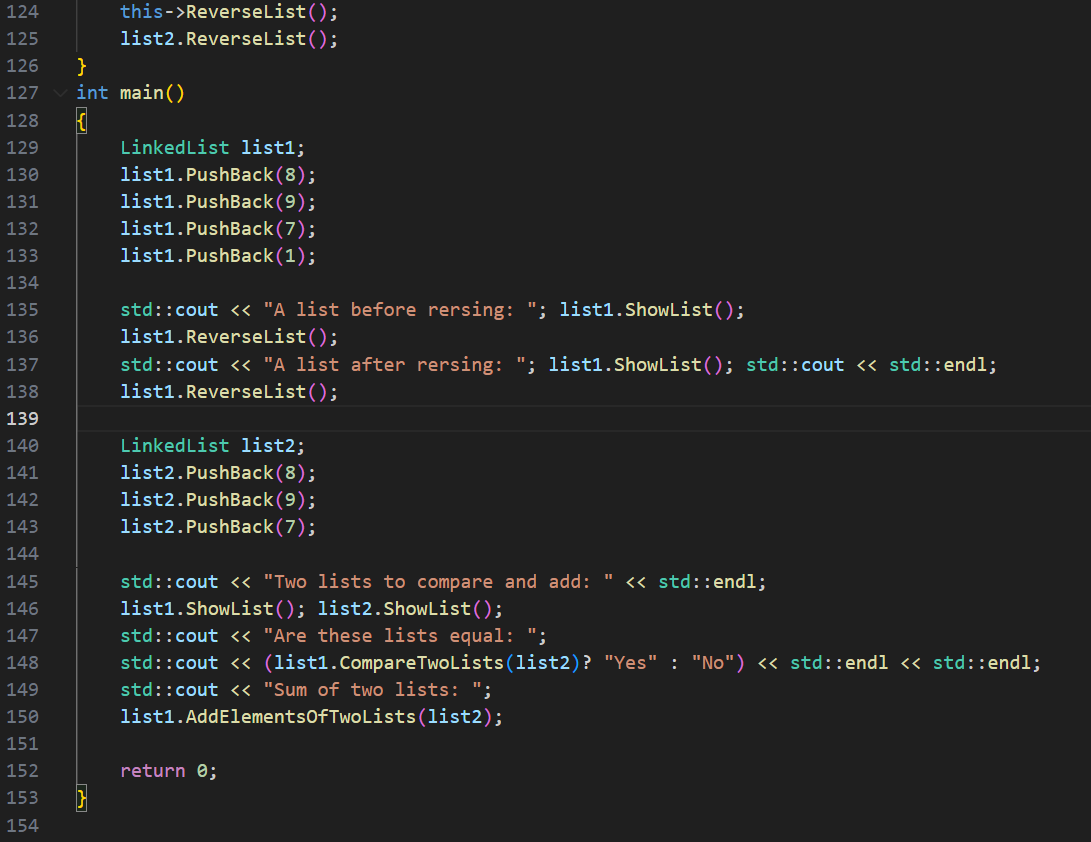
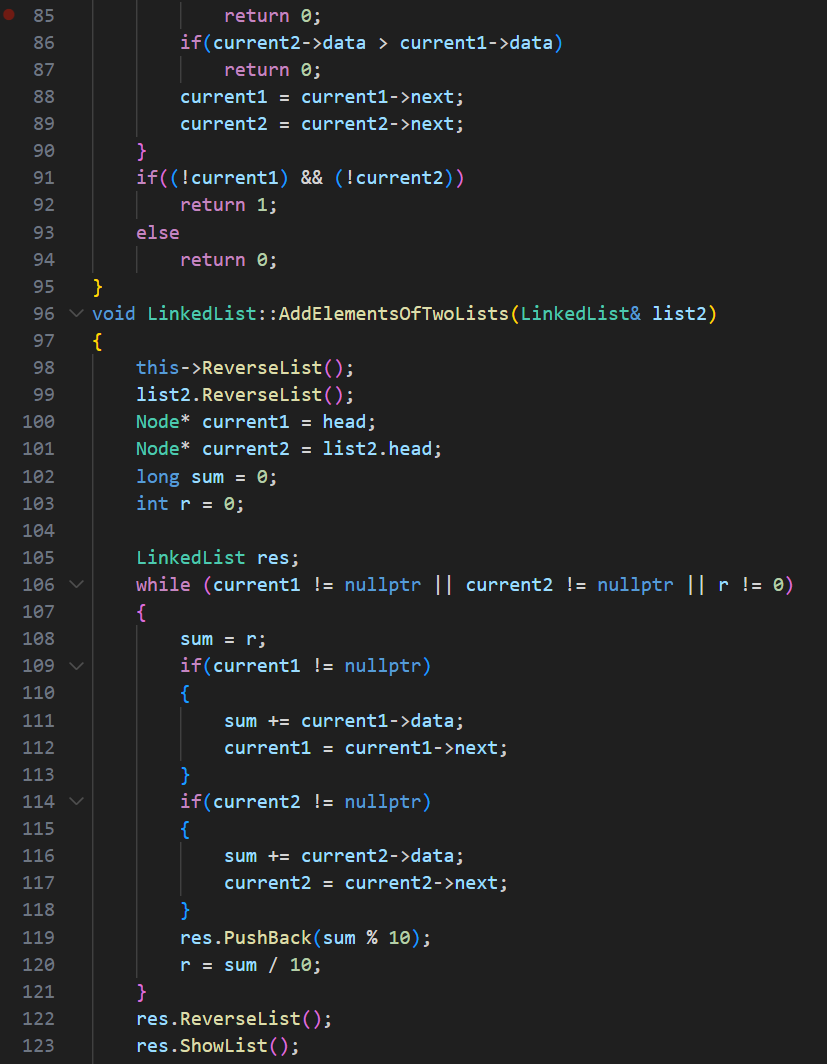
1. Реалізувати метод реверсу списку: Node\* reverse(Node \*head);

2. Порівняти два списки

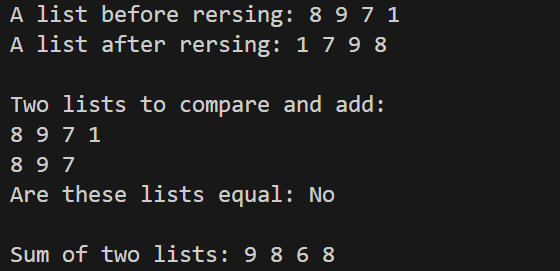
3. Додавання великих чисел (додати два списки)

**1. Код програми:**

****

****

**2. Результат:**

****

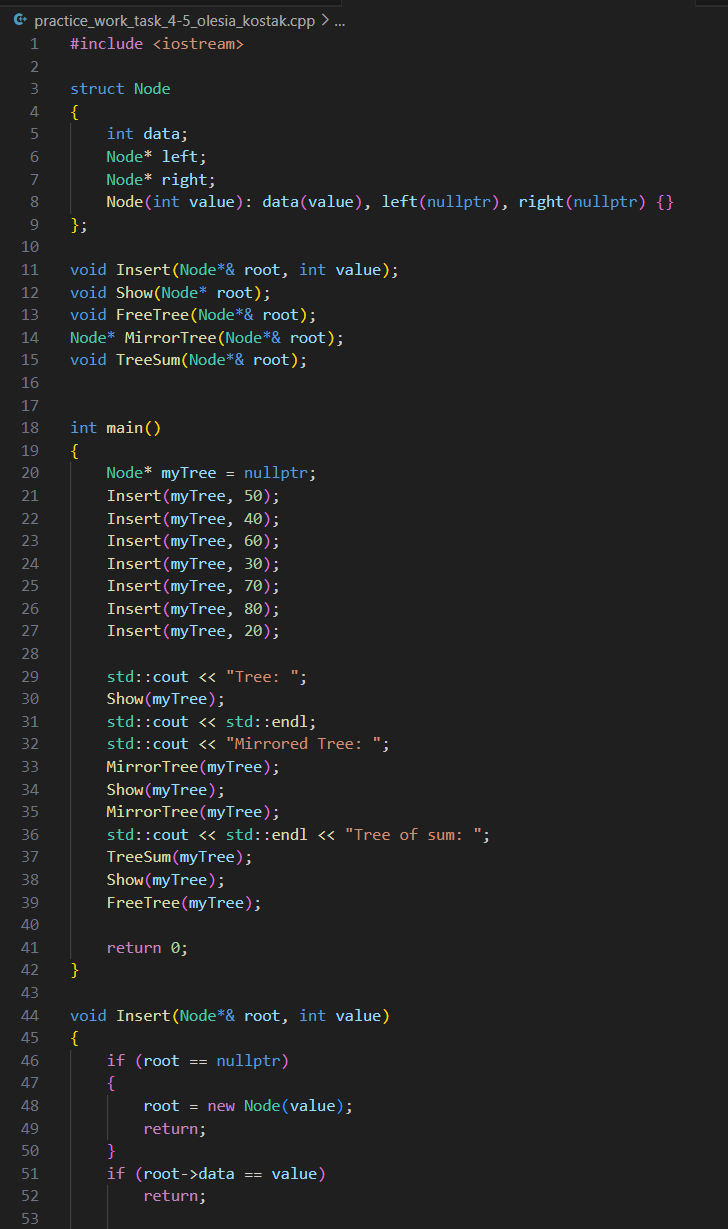
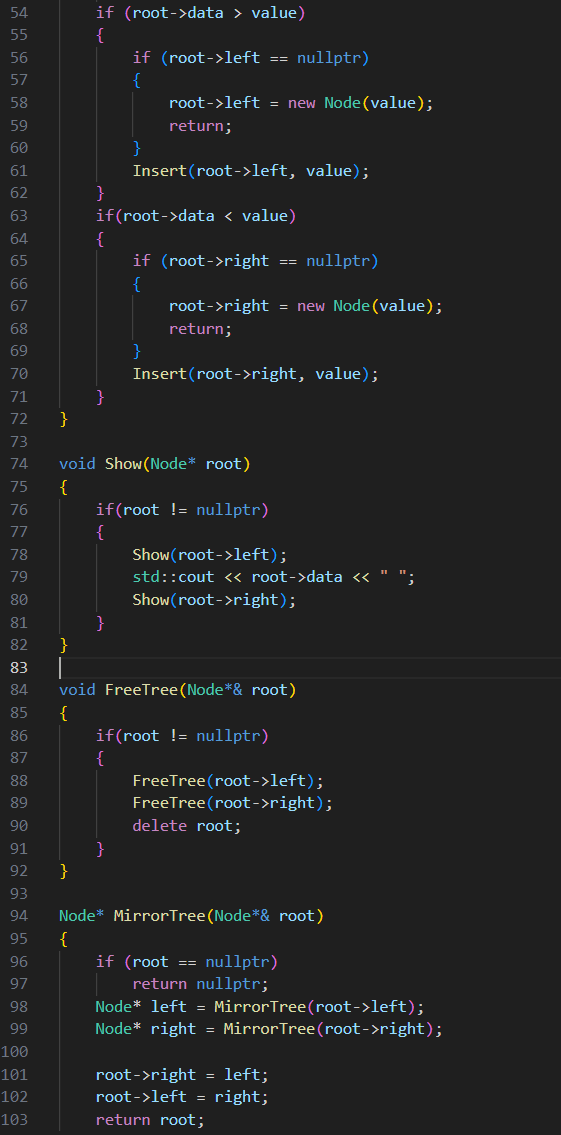
Завдання **№5 Class Practice Work (4-5)**

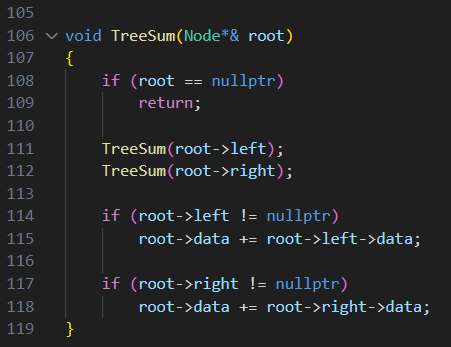
* *Варіант завдання*: -
* *Деталі завдання*:

1. Віддзеркалити дерево.

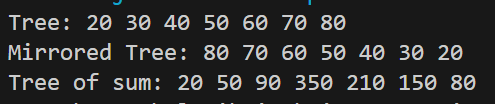
2. Записати кожному батьківському вузлу суму підвузлів.

**1. Код програми:**

**** ****

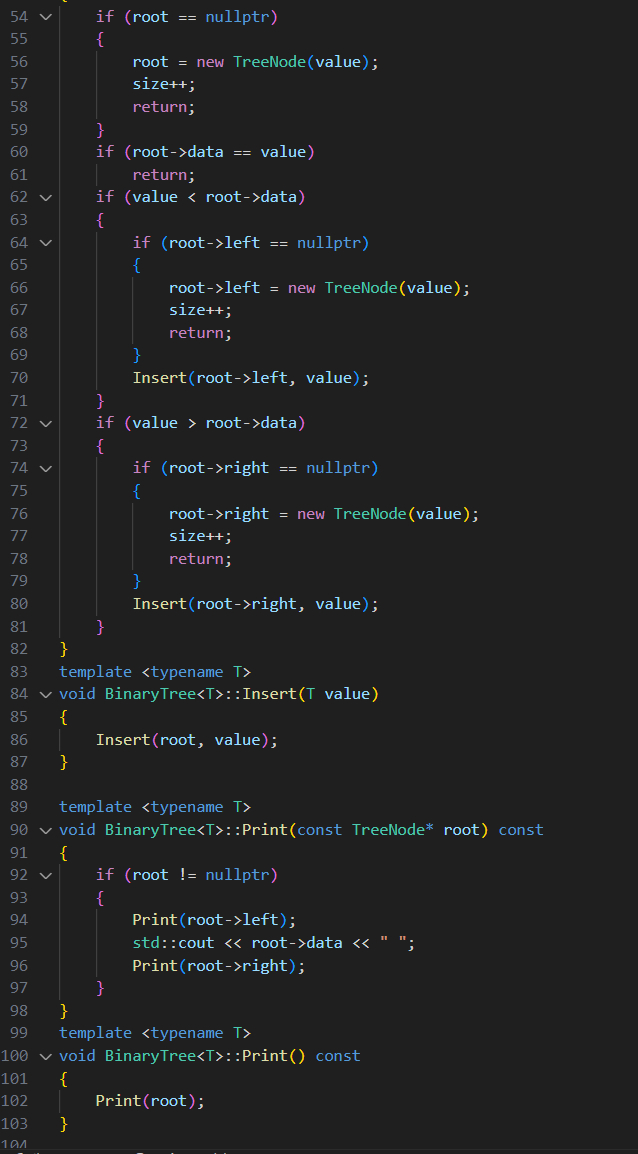
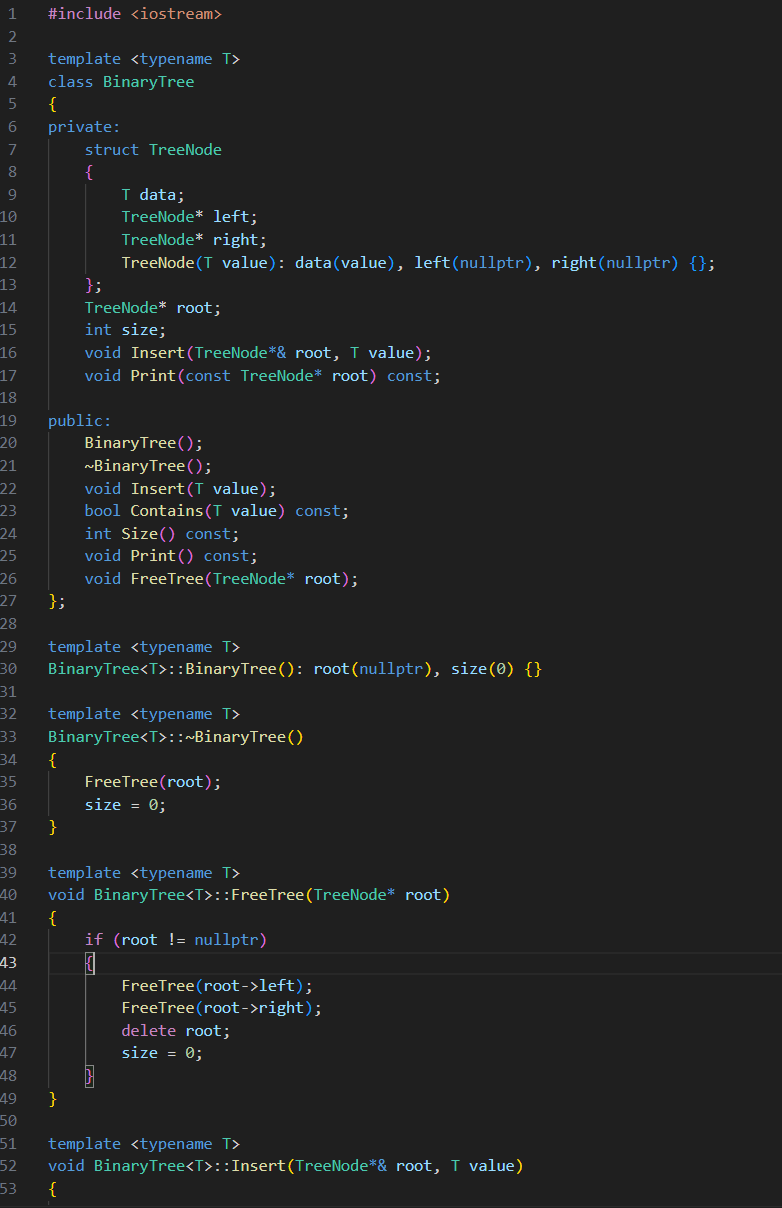
****

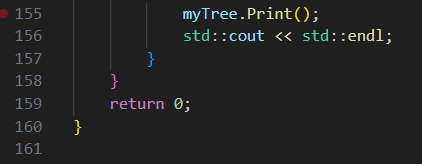
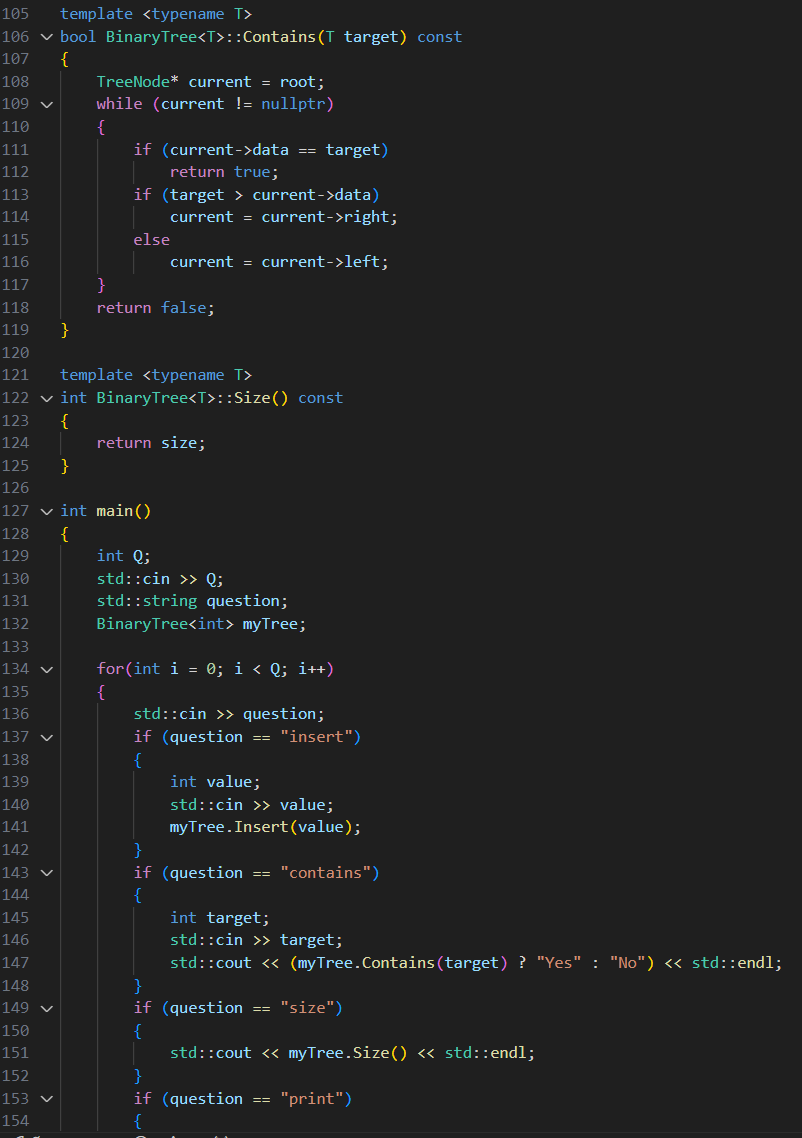
**2. Результат:**

****

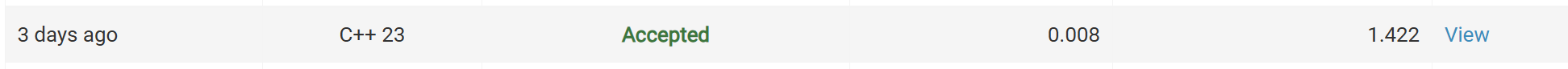
Завдання **№6 Self Practice Work (**[Lab 78v3](https://algotester.com/en/ContestProblem/DisplayWithEditor/135609)**)**

**1. Код програми:**



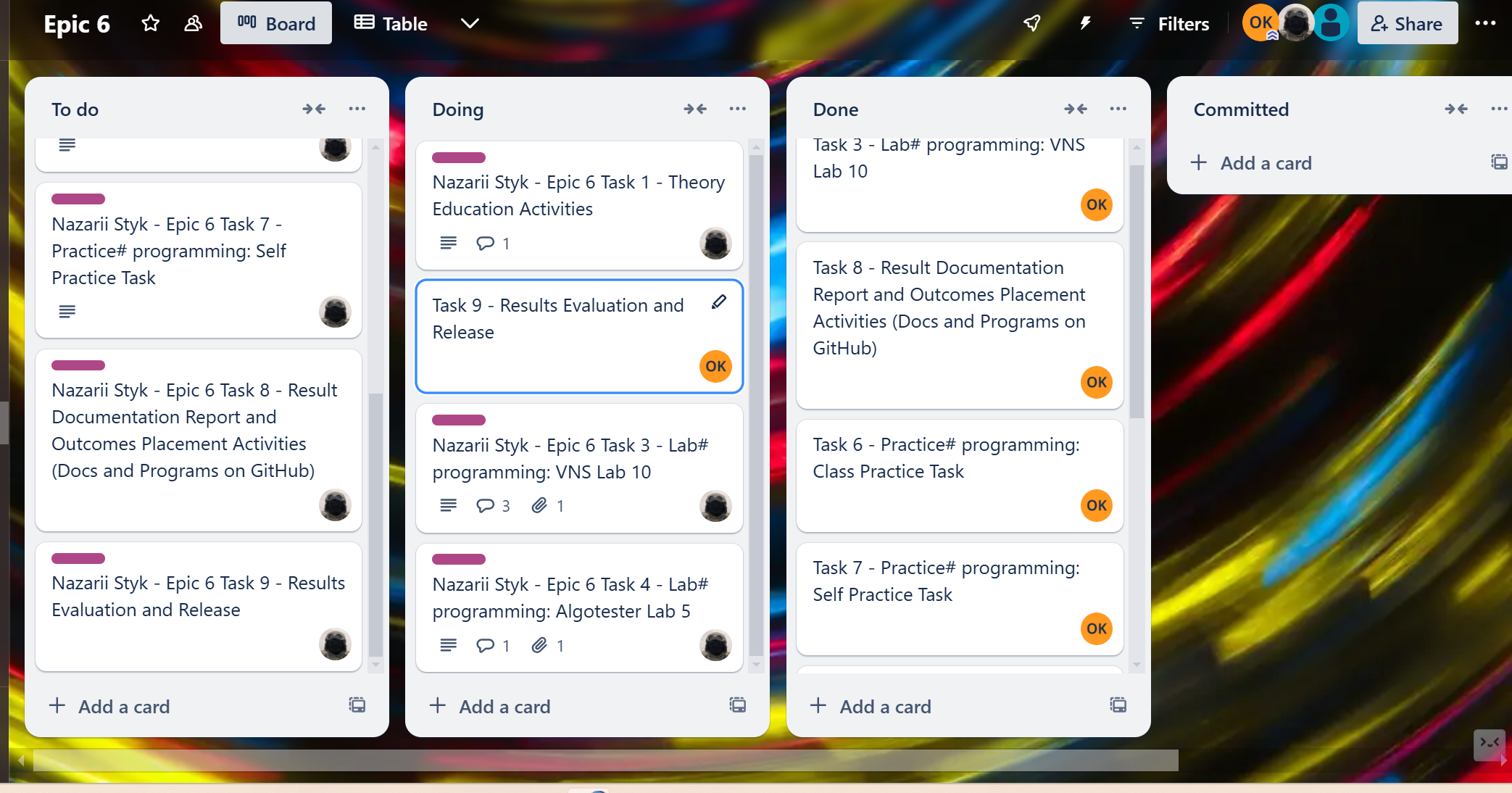
****

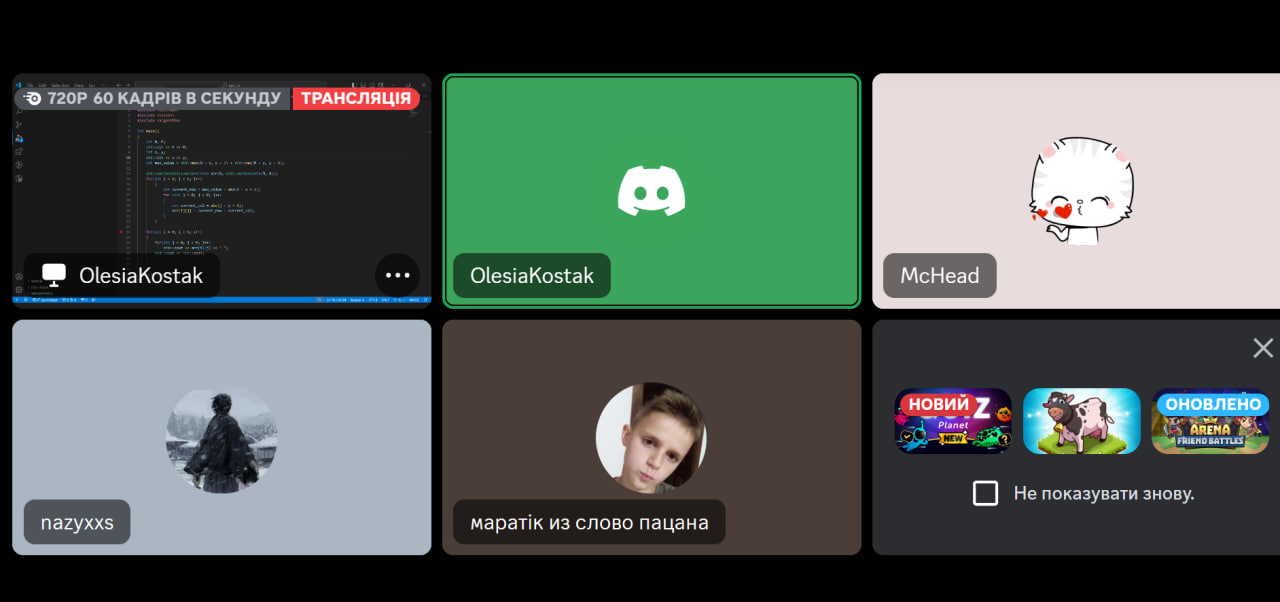
**2. Результат:**

****

**Кооперація з командою:**

Спільна борда в трелло по епіку:



Балакаємо про епіки:

**Висновки:**

По завершенню завдань Епіка №6, я освоїла роботу з динамічними структурами даних, такими як черга, стек, списки та дерева. Навчилася реалізовувати базові операції над цими структурами, включаючи створення, додавання, видалення, пошук та обхід елементів. Розвинула навички алгоритмічної обробки динамічних структур для ефективного вирішення прикладних задач.