

**Звіт**

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

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

***з дисципліни:*** «Основи програмування»

до:

ВНС Лабораторної Роботи № 10

Алготестер Лабораторної Роботи № 5

Алготестер Лабораторної Роботи № 7-8

Практичних Робіт до блоку № 6

**Виконала:**

Студентка групи ШІ-12

Лящук Соломія Володимирівна

Львів 2024

**ITERATION 6 / EPIC 6**

**“Динамічні структури (Черга, Стек, Списки, Дерево). Алгоритми обробки динамічних структур.”**

**Tasks - (Задачі 6 ітерації в Trello або Any Tasks Manager):**

* Epic 6 Task 1 - Theory Education Activities
* Epic 6 Task 2 - Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)
* Epic 6 Task 3 - Lab# programming: VNS Lab 10
* Epic 6 Task 4 - Lab# programming: Algotester Lab 5
* Epic 6 Task 5 - Lab# programming: Algotester Lab 7-8
* Epic 6 Task 6 - Practice# programming: Class Practice Task
* Epic 6 Task 7  - Practice# programming:  Self Practice Task
* Epic 6 Task 8  - Result Documentation Report and Outcomes Placement Activities (Docs and Programs on GitHub)
* Epic 6 Task 9 - Results Evaluation and Release

**Task 1**

**Theory Education Activities**

**“Динамічні структури (Черга, Стек, Списки, Дерево). Алгоритми обробки динамічних структур.”**

**Agenda (План Практичної):**

1. Основи Динамічних Структур Даних:
   * Вступ до динамічних структур даних: визначення та важливість
   * Виділення пам'яті для структур даних (stack і heap)
   * Приклади простих динамічних структур: динамічний масив
2. Стек:
   * Визначення та властивості стеку
   * Операції push, pop, top: реалізація та використання
   * Приклади використання стеку: обернений польський запис, перевірка балансу дужок
   * Переповнення стеку
3. Черга:
   * Визначення та властивості черги
   * Операції enqueue, dequeue, front: реалізація та застосування
   * Приклади використання черги: обробка подій, алгоритми планування
   * Розширення функціоналу черги: пріоритетні черги
4. Зв'язні Списки:
   * Визначення однозв'язного та двозв'язного списку
   * Принципи створення нових вузлів, вставка між існуючими, видалення, створення кільця(circular linked list)
   * Основні операції: обхід списку, пошук, доступ до елементів, об'єднання списків
   * Приклади використання списків: управління пам'яттю, FIFO та LIFO структури
5. Дерева:
   * Вступ до структури даних "дерево": визначення, типи
   * Бінарні дерева: вставка, пошук, видалення
   * Обхід дерева: в глибину (preorder, inorder, postorder), в ширину
   * Застосування дерев: дерева рішень, хеш-таблиці
   * Складніші приклади дерев: AVL, Червоно-чорне дерево
6. Алгоритми Обробки Динамічних Структур:
   * Основи алгоритмічних патернів: ітеративні, рекурсивні
   * Алгоритми пошуку, сортування даних, додавання та видалення елементів

Sourse:

<https://www.youtube.com/watch?v=-25REjF_atI&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=139>

<https://prometheus.org.ua/cs50/sections/section6.html>

<https://www.geeksforgeeks.org/list-reverse-function-in-c-stl/>

<https://www.bestprog.net/uk/2019/09/26/c-queue-general-concepts-ways-to-implement-the-queue-implementing-a-queue-as-a-dynamic-array-ua/#google_vignette>

<https://itproger.com/ua/spravka/cpp/queue>

<https://www.geeksforgeeks.org/circular-linked-list-in-cpp/>

<https://purecodecpp.com/uk/archives/2483#google_vignette>

<https://acode.com.ua/urok-141-perevantazhennya-operatoriv-vvodu-i-vyvodu/#toc-0>

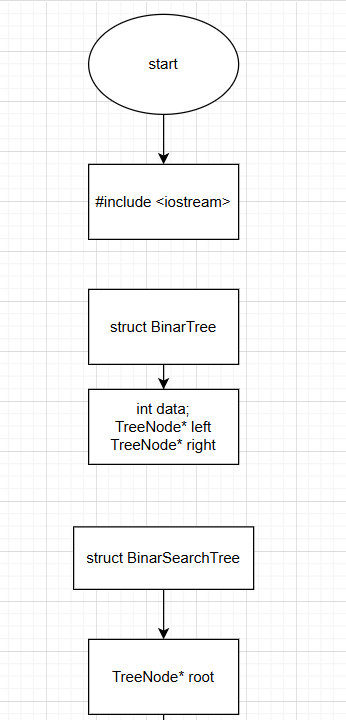
<https://www.bestprog.net/uk/2019/09/18/c-the-concept-of-stack-operations-on-the-stack-an-example-implementation-of-the-stack-as-a-dynamic-array-ua/#google_vignette>

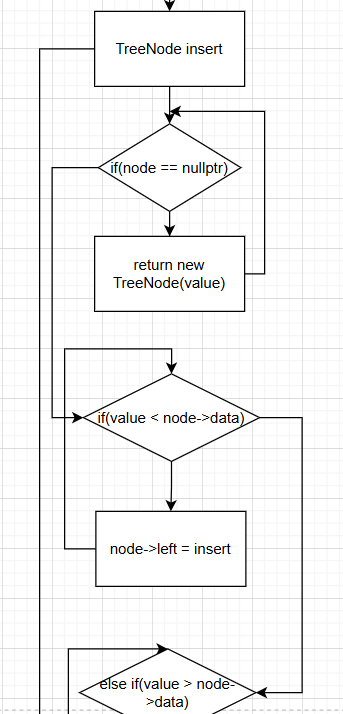
<https://www.geeksforgeeks.org/stdstringcompare-in-c/>

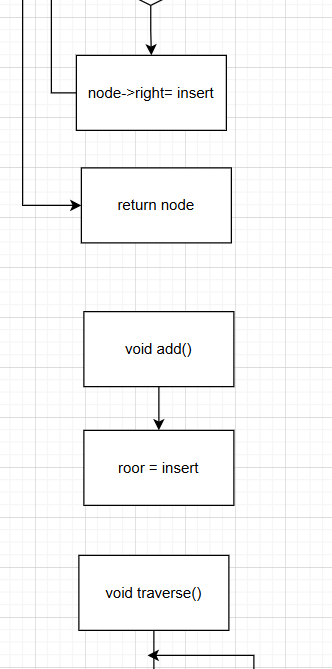
<https://www.geeksforgeeks.org/cpp-program-for-inserting-a-node-in-a-linked-list/>

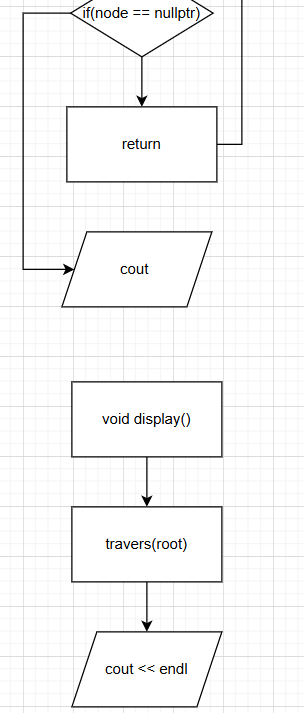
**Task 2**

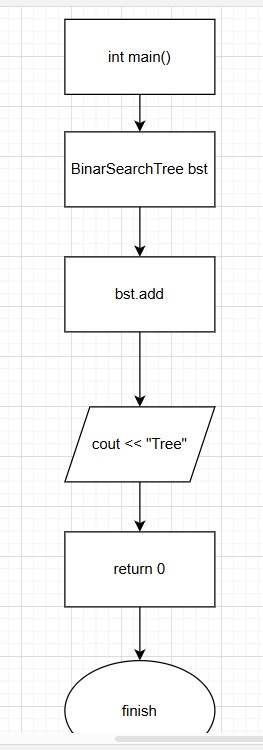
**Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)**

****

****

****

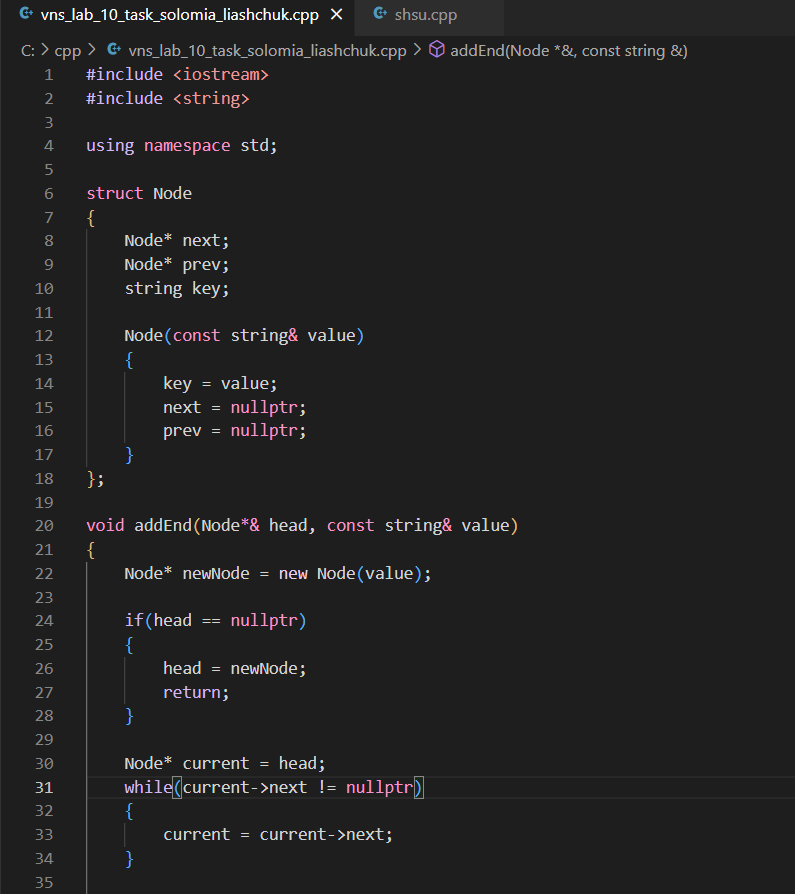
****

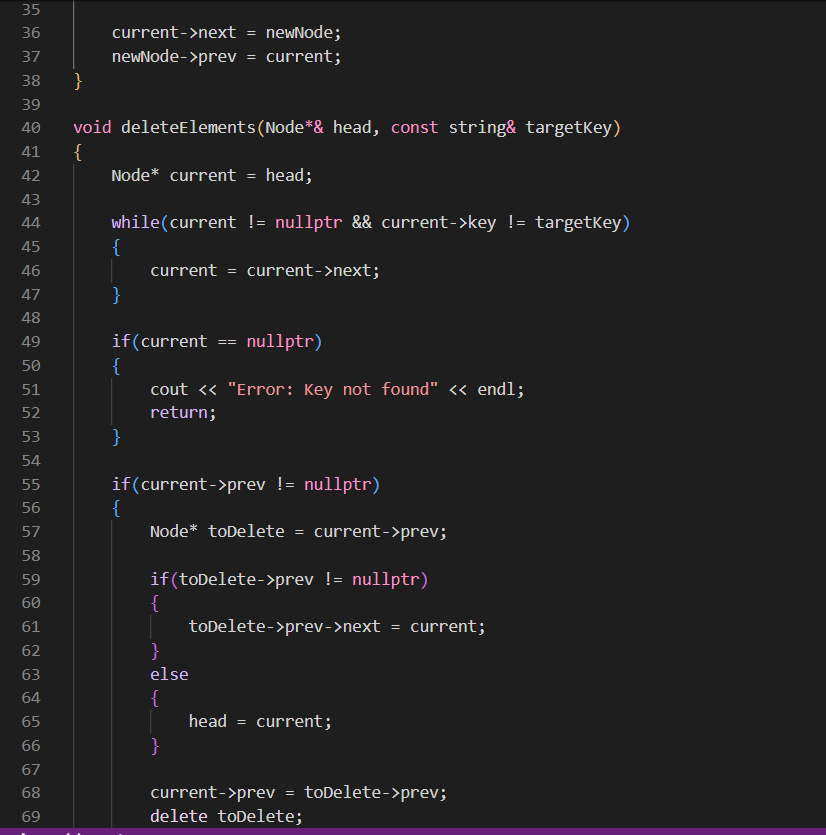
****

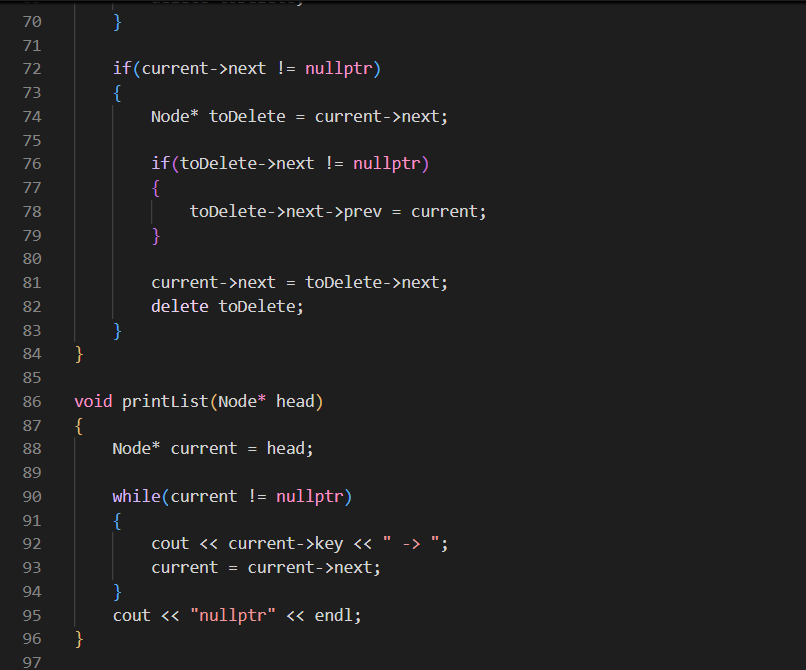
**Task 3**

**VNS Lab 10**

21.Записи в лінійному списку містять ключове поле типу \*char (рядок символів). Сформувати двонаправлений список. Знищити елементи перед і після елемента із заданим ключем. Додати по К елементів у початок й у кінець списку.



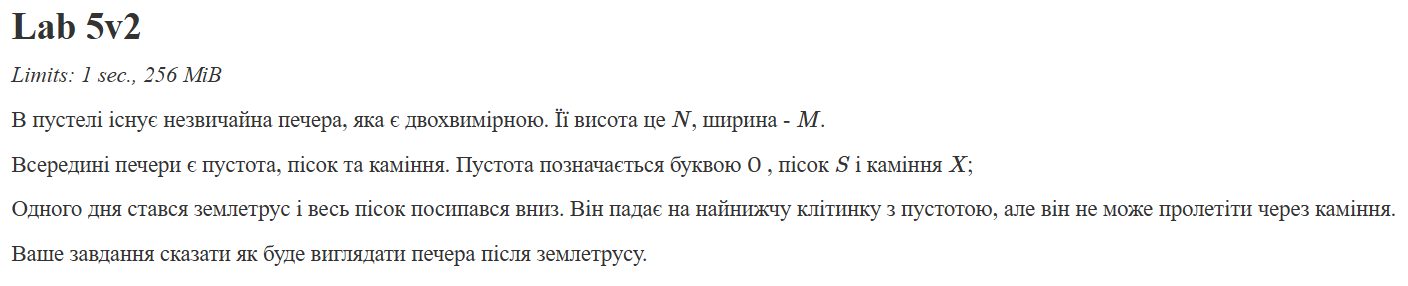


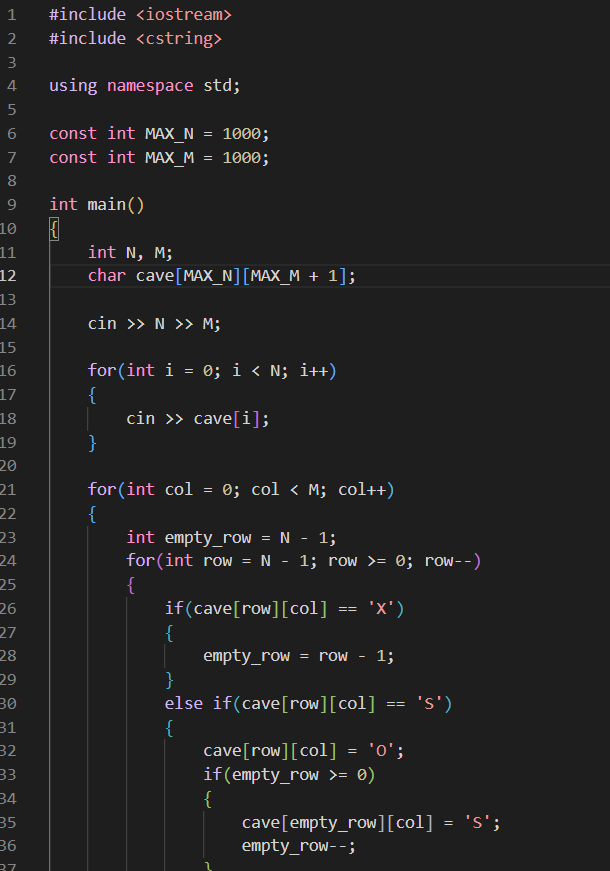


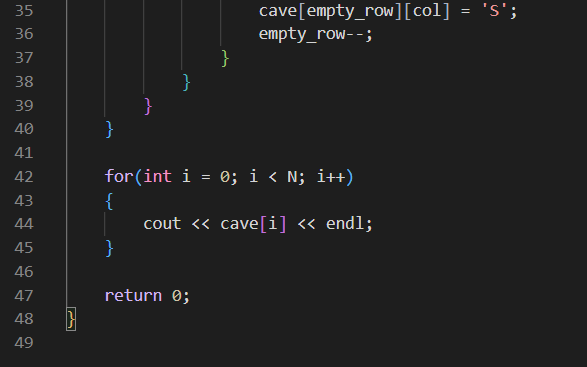


**Task 4**

**Algotester Lab 5**

****



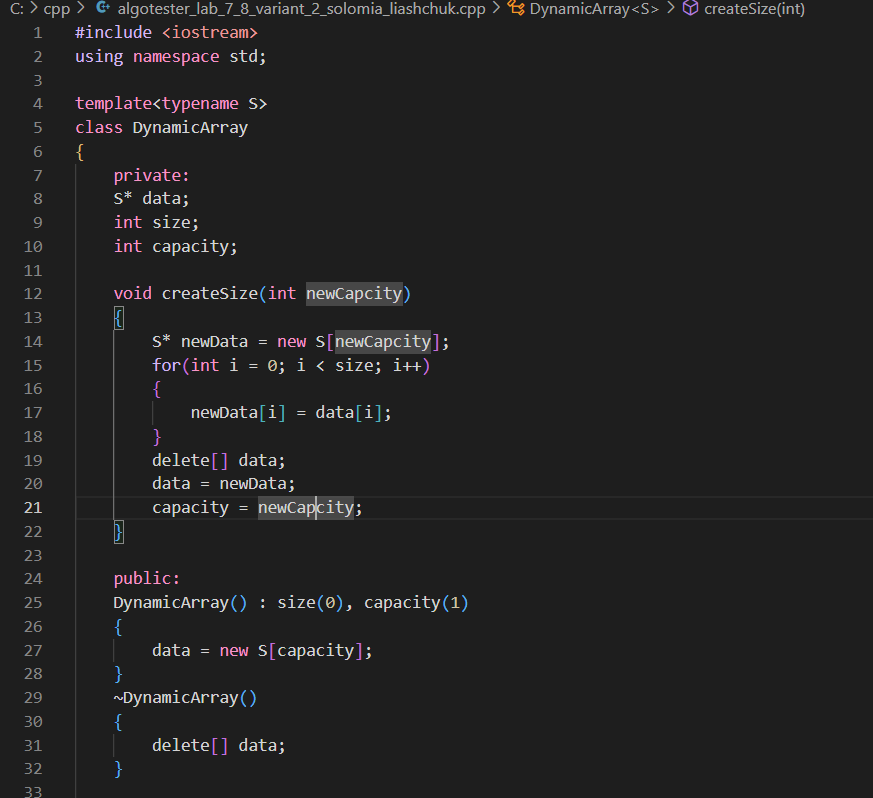


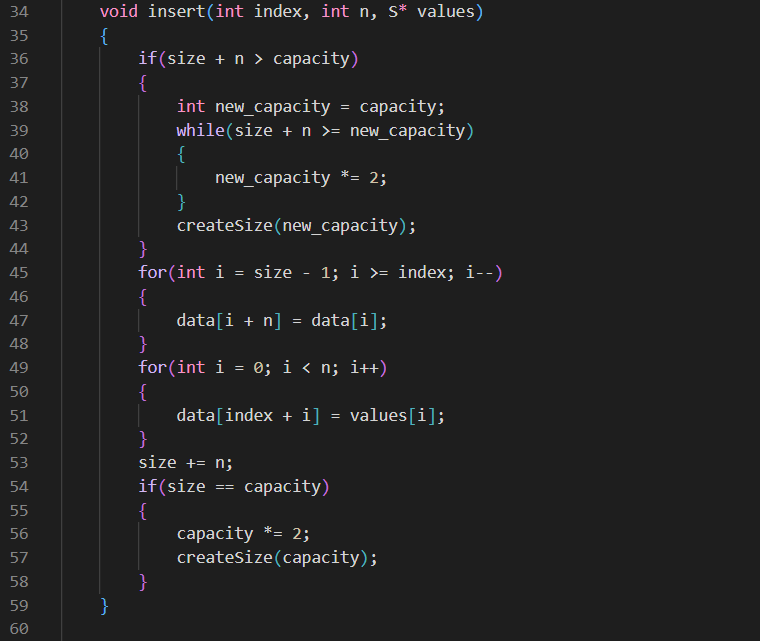
**Task 5**

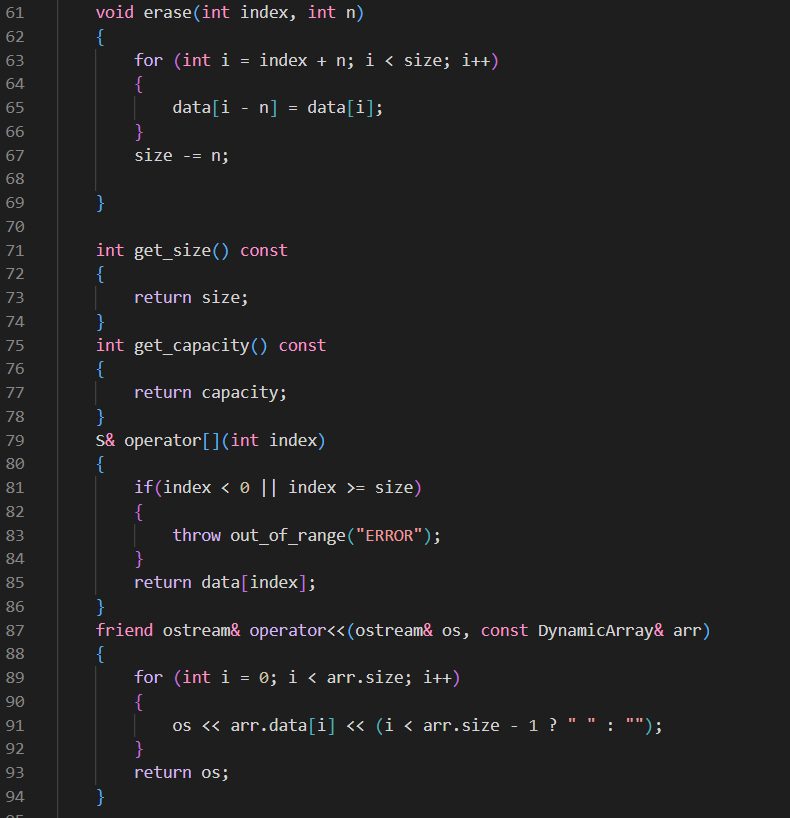
**Algotester Lab 7-8**

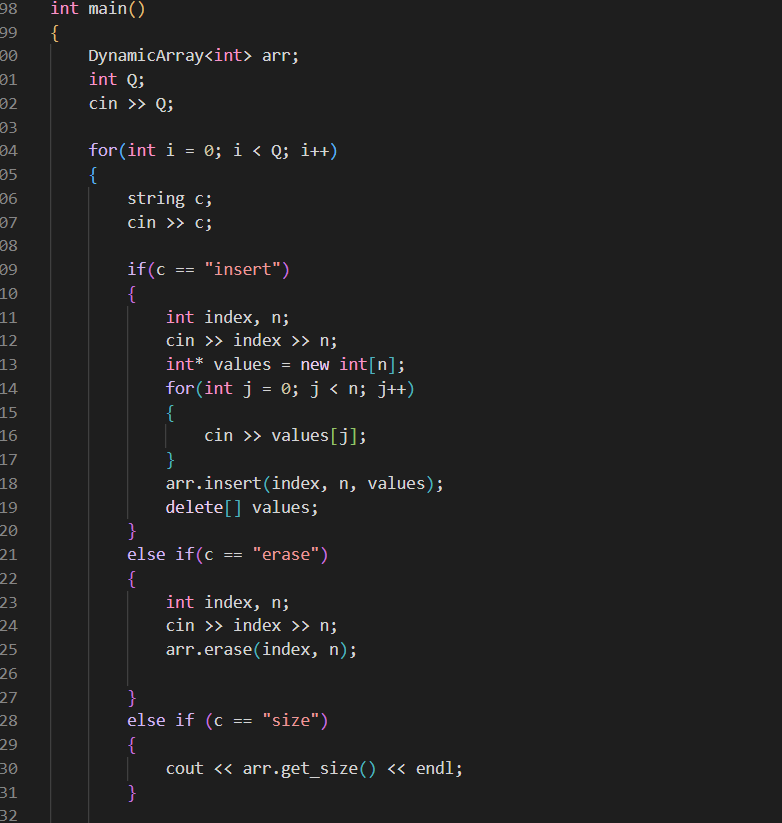
**V – 2**

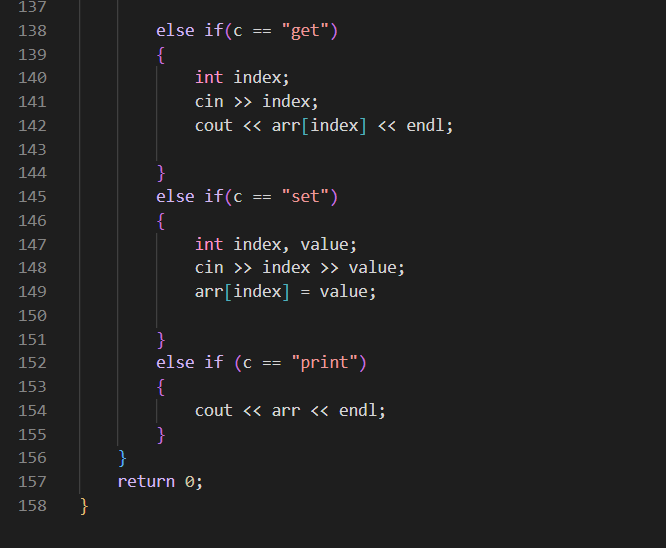
Ваше завдання - власноруч реалізувати структуру даних "Динамічний масив".  
Ви отримаєте QQ запитів, кожен запит буде починатися зі слова-ідентифікатора, після якого йдуть його аргументи.

****

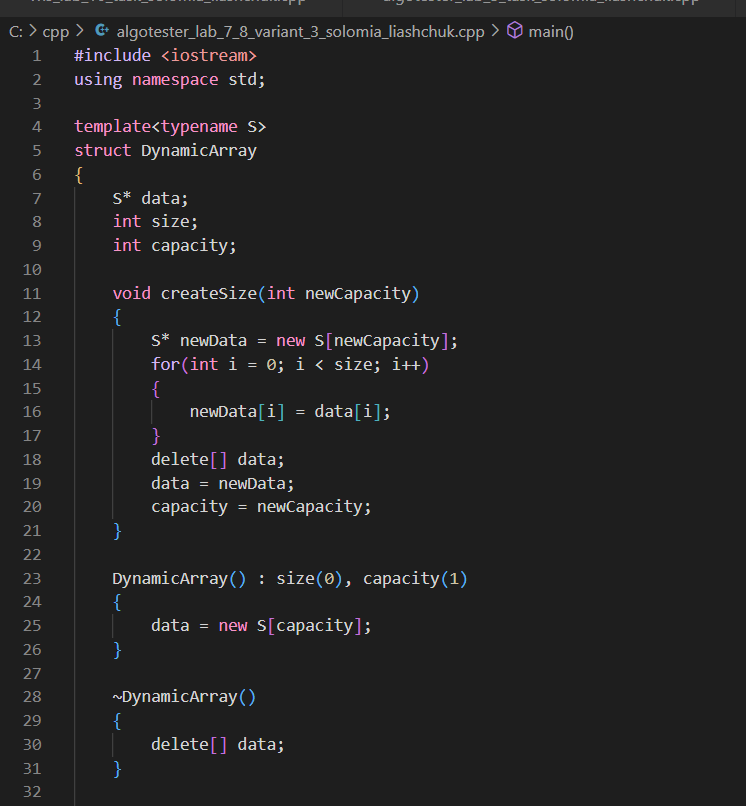
****

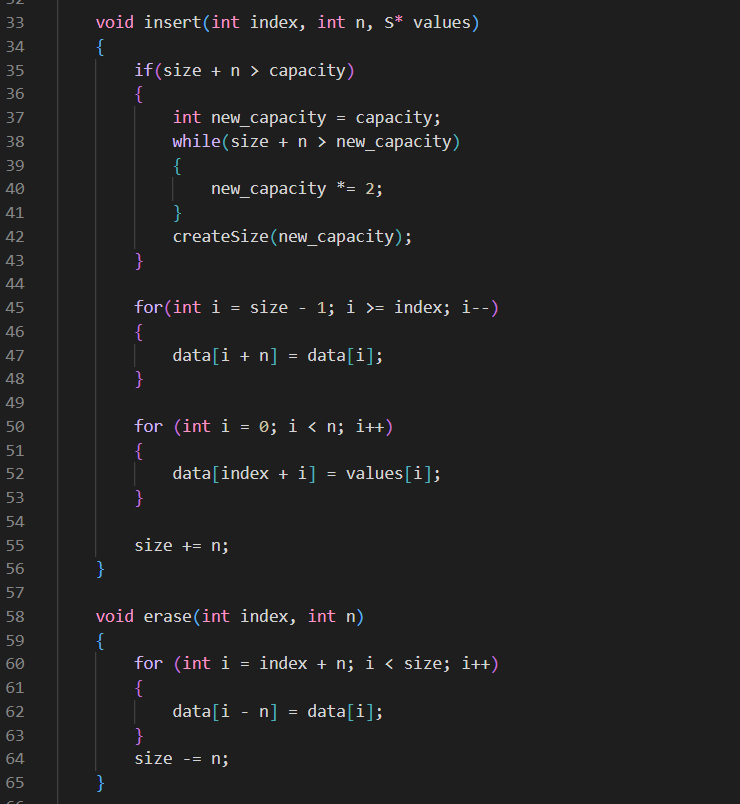
****

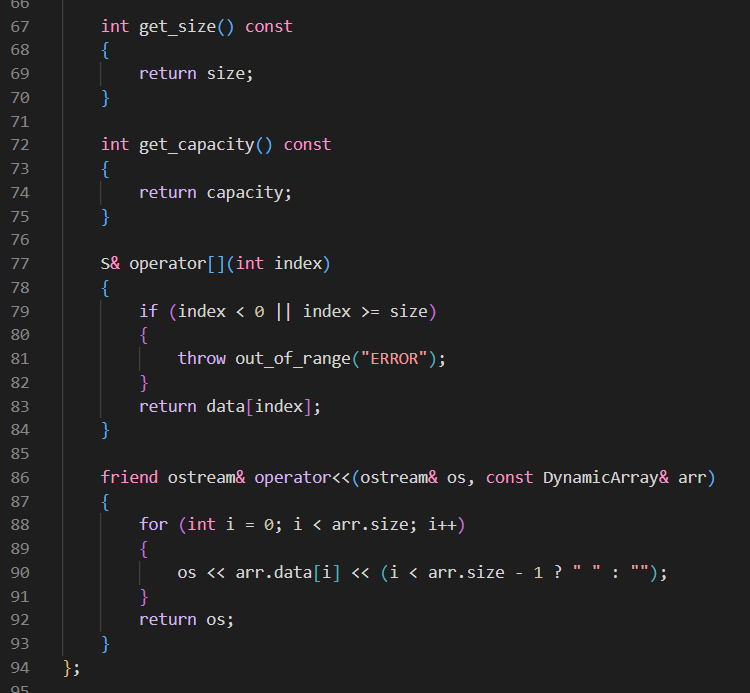
****

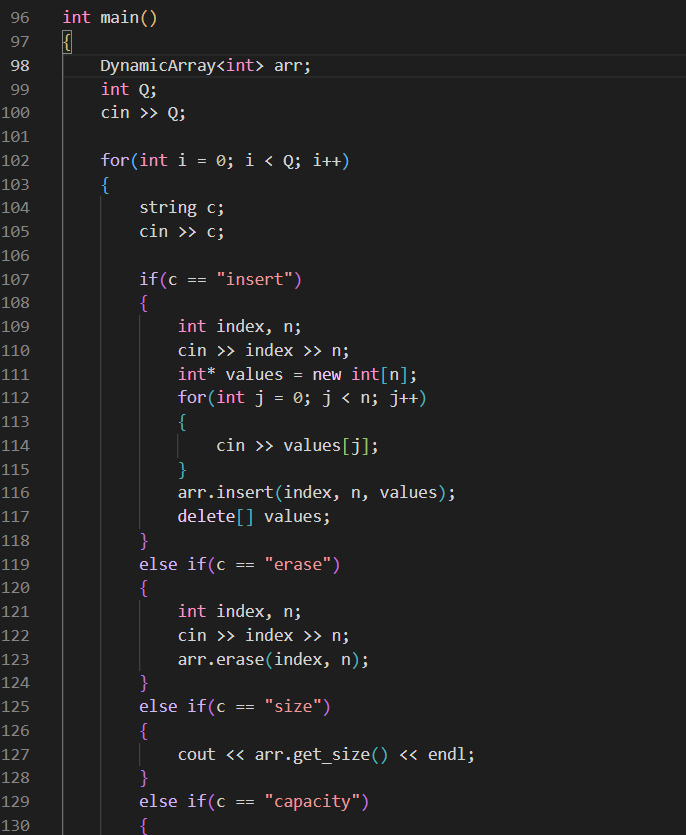
****

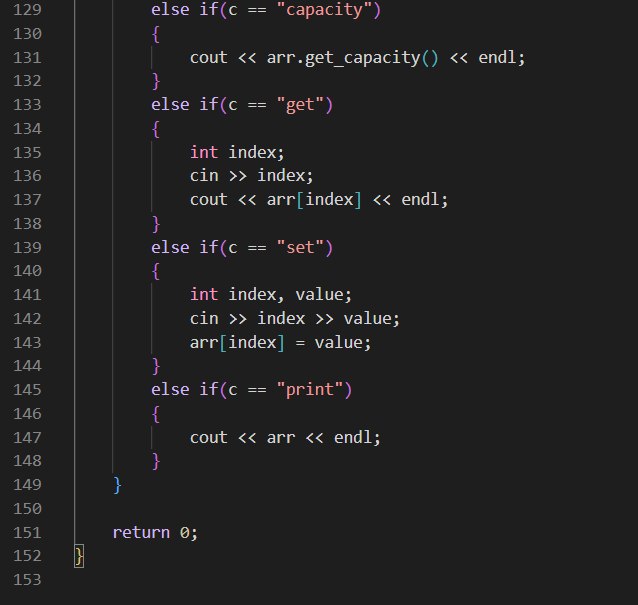
**Реалізація через структуру:**

****

****

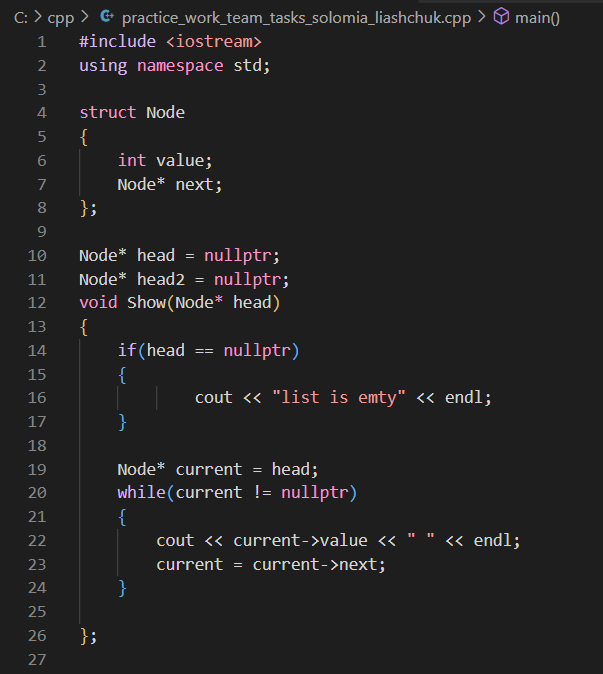
****

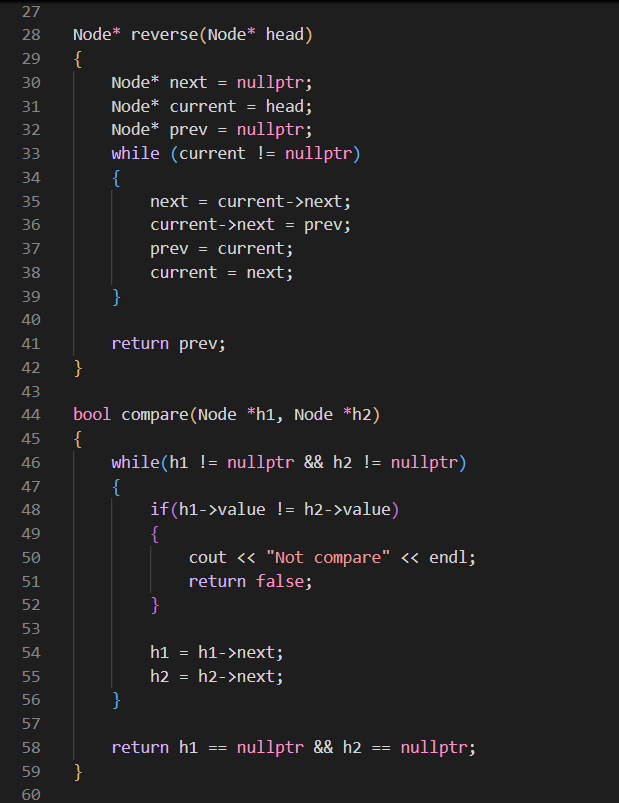
****

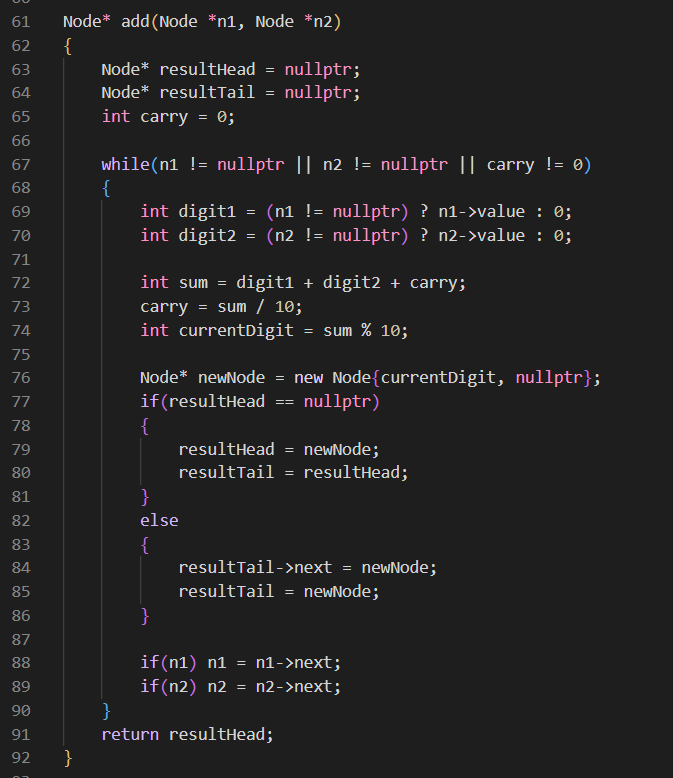
****

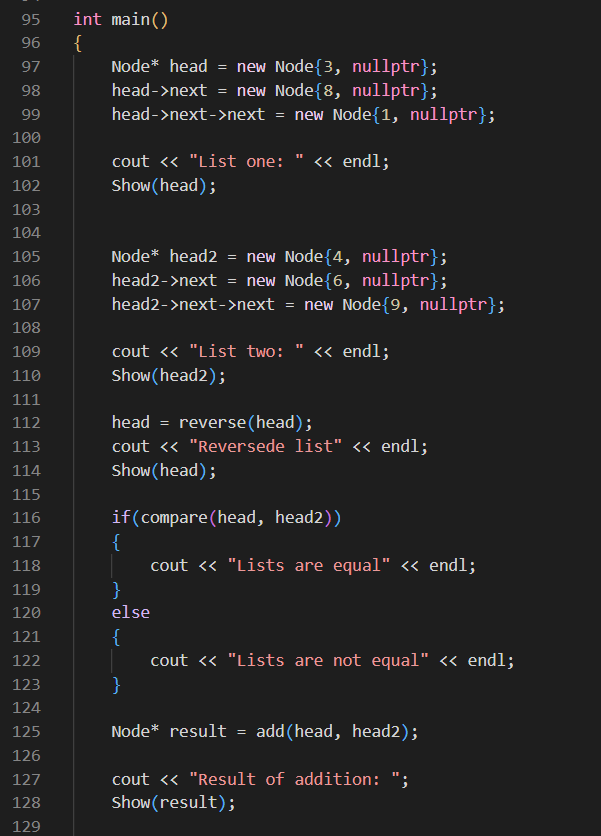
**Task 6**

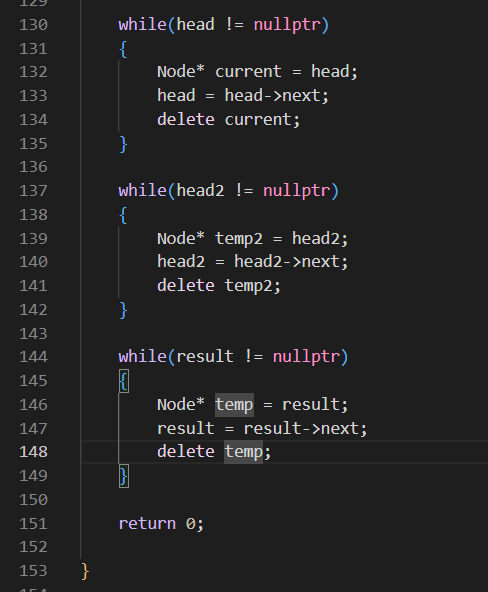
**Class Practice Task(робота з списками1-3)**

****

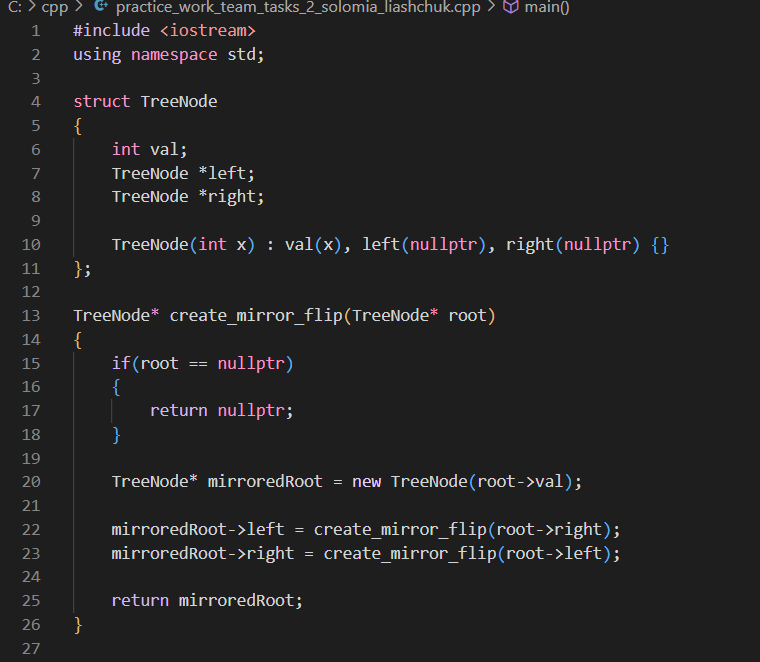
****

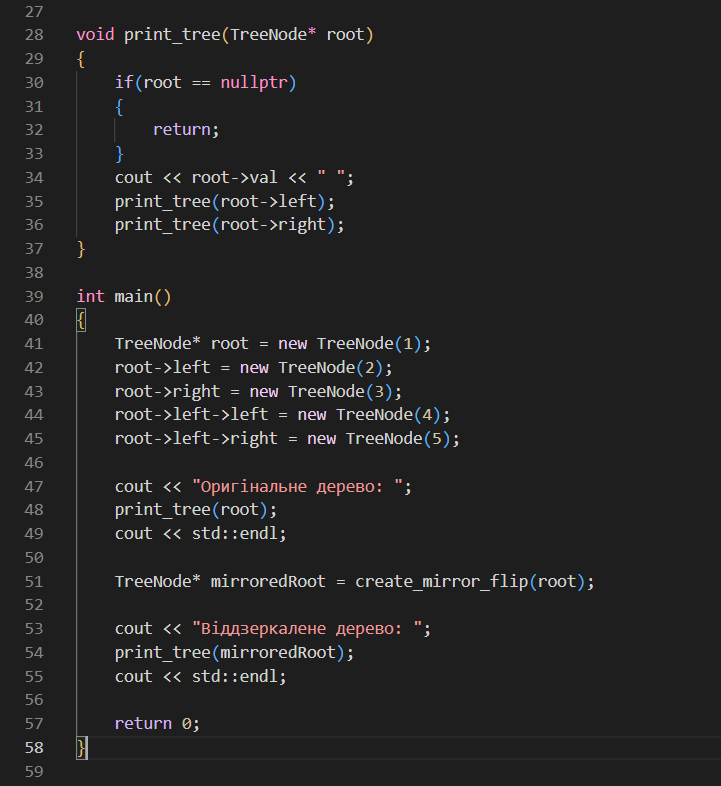
****

****

****

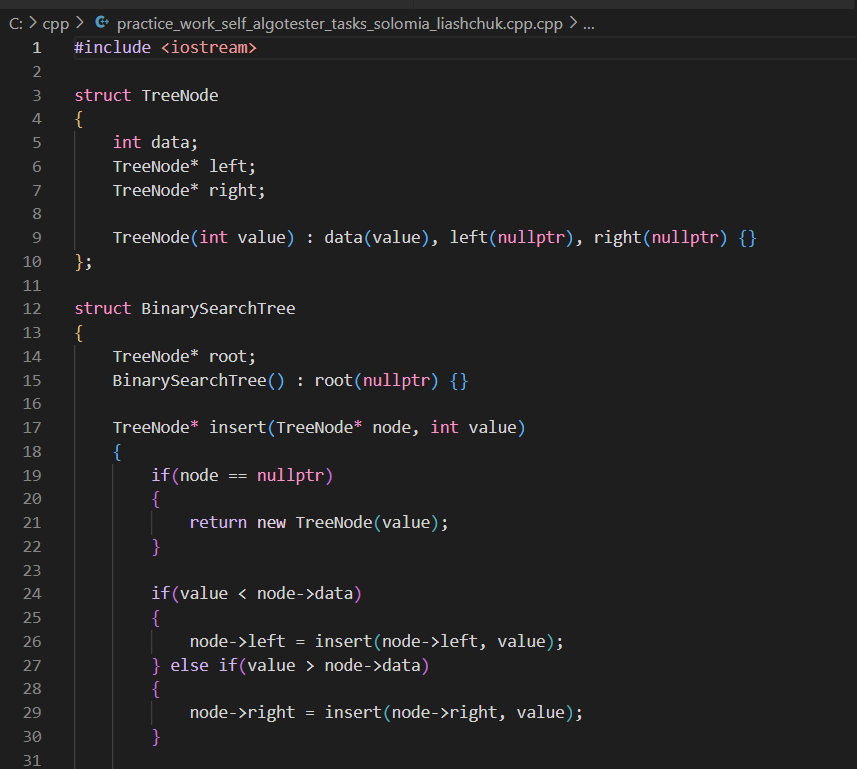
**Class Practice Task(робота з деревами4-5)**

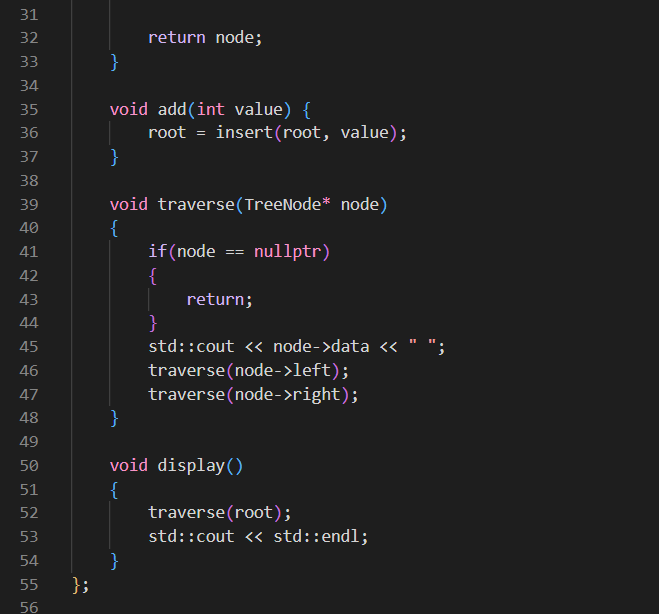
****

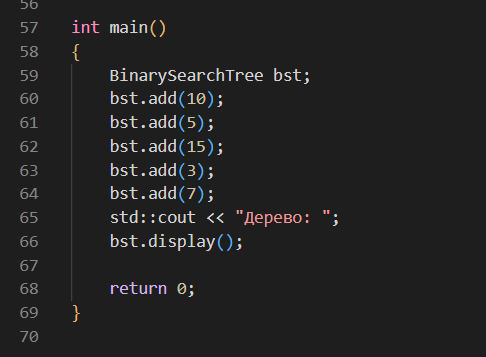
****

**Task 7**

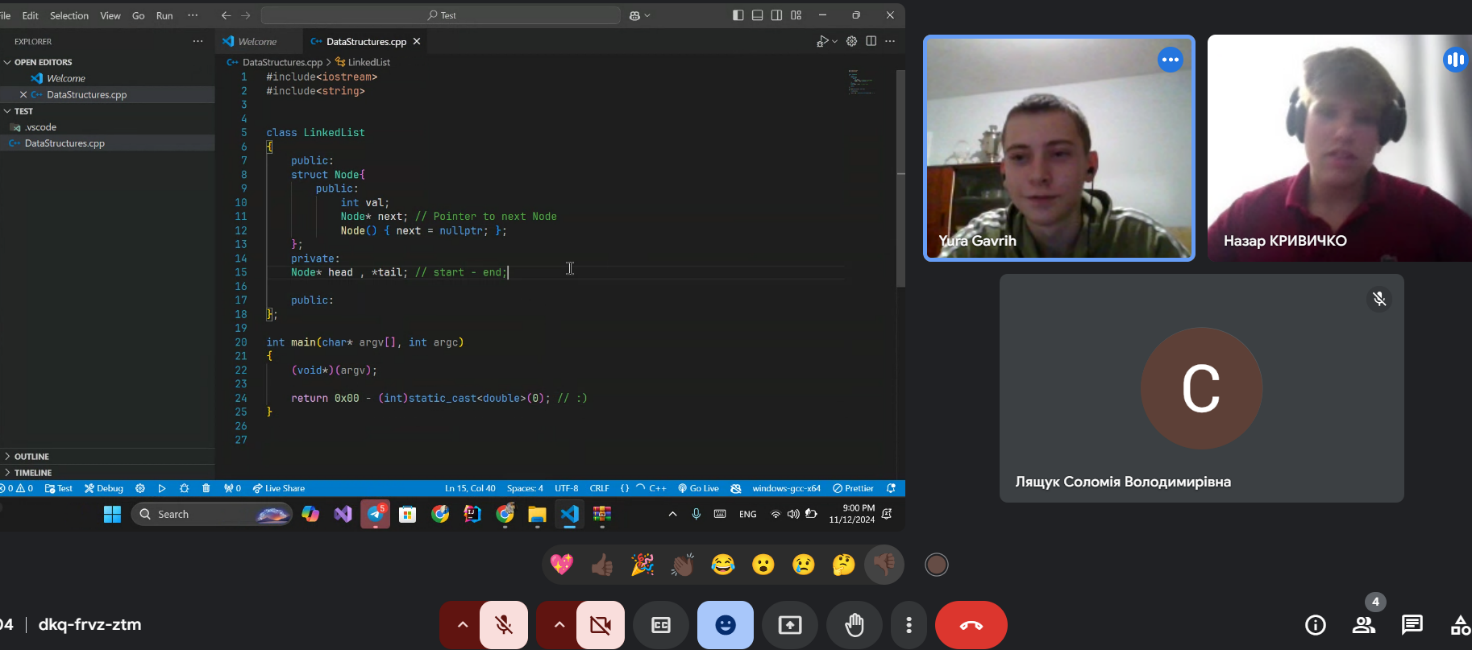
**Self Practice Task**

****

****

****

**Робота в команді:**



**Висновок:** цей епік був найскладніший, тому що великий об’єм нового матеріалу. Я дізналася багато нового й одразу попрактикувалася в цих темах, відкрила для себе нову нішу в програмуванні. Дуже цікаво було працювати з списками та деревами.