Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту

A blue and white logo

Description automatically generated

**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

**Виконав:**

Студент групи ШІ-13

Штурин Захарій Михайлович

Львів 2024

**Тема:**

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

**Мета:**

Навчитись створювати свої STL елементи використовуючи класи та структури, правильно використати вказівники, аби досягти максимальної ефективності та функціональності.

**Теоретичні відомості:**

1. Вивчив/знав:
   1. Дерева
   2. Стек
   3. Черга
   4. Зв’язаний список
2. Джерела:

Всю інформацію до теоретичних відомостей я отримав на лекційних/практичних парах. Додатково використовував сайт <https://acode.com.ua/> та <https://www.w3schools.com/>

**Виконання роботи:**

1. *Опрацювання завдання та вимог до програми та середовища*

**Завдання №1 Епік 6 - Практичне завдання**

Створити зв’язаний список та реалізувати на ньому операції приєднання, переведення та порівняння

Створити бінарне дерево та реалізувати на ньому операції підрахування суми нащадків і запис їх у батьківську вершину, а також операція віддзеркалення дерева

**Завдання №2 algotester 5v3**

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.

Клітинкі які мають суміжну сторону з вершиною мають висоту на один меншу, суміжні з ними і не розфарбовані мають ще на 1 меншу висоту і так далі.

**Завдання №3 algotester 78v1**

Ваше завдання - власноруч реалізувати структуру даних "Двозв’язний список".  
Ви отримаєте Q запитів, кожен запит буде починатися зі слова-ідентифікатора, після якого йдуть його аргументи.  
  
Вам будуть поступати запити такого типу:

* **Вставка**:  
  Ідентифікатор - insert  
  Ви отримуєте ціле число index елемента, на місце якого робити вставку.  
  Після цього в наступному рядку рядку написане число N - розмір списку, який треба вставити.  
  У третьому рядку N цілих чисел - список, який треба вставити на позицію index.
* **Видалення**:  
  Ідентифікатор - erase  
  Ви отримуєте 2 цілих числа - index, індекс елемента, з якого почати видалення та n - кількість елементів, яку треба видалити.
* **Визначення розміру**:  
  Ідентифікатор - size  
  Ви не отримуєте аргументів.  
  Ви виводите кількість елементів у списку.
* **Отримання значення** i-го елементу  
  Ідентифікатор - get  
  Ви отримуєте ціле число - indexindex, індекс елемента.  
  Ви виводите значення елемента за індексом.
* **Модифікація значення** i-го елементу  
  Ідентифікатор - set  
  Ви отримуєте 2 цілих числа - індекс елемента, який треба змінити, та його нове значення.
* **Вивід списку на екран**  
  Ідентифікатор - print  
  Ви не отримуєте аргументів.  
  Ви виводите усі елементи списку через пробіл.  
  Реалізувати використовуючи перегрузку оператора <<

**Завдання №4 vns 10v8**

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.

Записи в лінійному списку містять ключове поле типу int. Сформувати

двонаправлений список. Знищити з нього елемент після елемента із заданим

номером, додати К елементів у початок списку.

**Завдання №5 self-practice algotester 78v3**

Ваше завдання - власноруч реалізувати структуру даних "Двійкове дерево пошуку".  
Ви отримаєте Q запитів, кожен запит буде починатися зі слова-ідентифікатора, після якого йдуть його параметри.  
  
Вам будуть поступати запити такого типу:

* **Вставка**:  
  Ідентифікатор - insert  
  Ви отримуєте ціле число value - число, яке треба вставити в дерево.
* **Пошук**:  
  Ідентифікатор - contains  
  Ви отримуєте ціле число value - число, наявність якого у дереві необхідно перевірити.  
  Якщо value наявне в дереві - ви виводите Yes, у іншому випадку No.
* **Визначення розміру**:  
  Ідентифікатор - size  
  Ви не отримуєте аргументів.  
  Ви виводите кількість елементів у дереві.
* **Вивід дерева на екран**  
  Ідентифікатор - print  
  Ви не отримуєте аргументів.  
  Ви виводите усі елементи дерева через пробіл.  
  Реалізувати використовуючи перегрузку оператора <<

*3) Код програми з посиланням на зовнішні ресурси*

Всі коди розташовані у папці Code

**Для практичного** practice\_work\_team\_tasks\_binary\_tree\_zakharii\_shturyn.cpp та practice\_work\_team\_tasks\_linked\_list\_zakharii\_shturyn.cpp

Планував витратити по 2-3 год на кожне

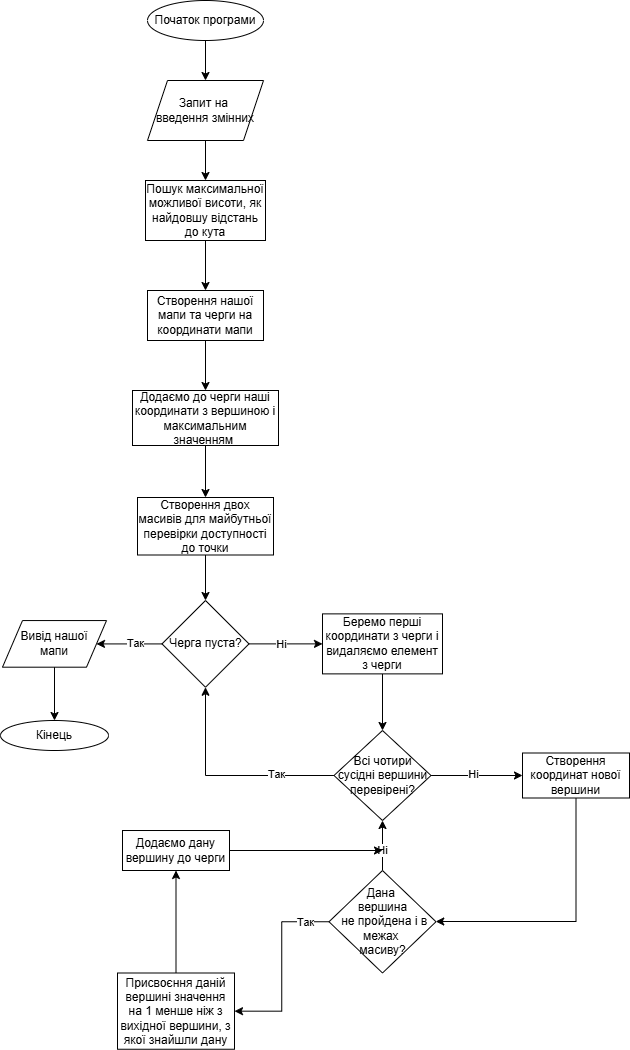
**Для лабораторних** відповідноvns\_lab\_(номер)\_task\_(варіант)\_ zakharii\_shturyn.cpp

Планував 3 год

**До алготестеру** відповідно algotester\_lab\_(номер)\_task\_(варіант)\_zakharii\_shturyn.cpp

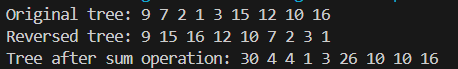
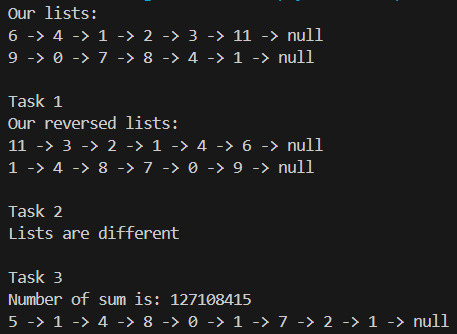
До 78 планував 5 год, до 5 планував 3 год

**Algotester lab 5v3**



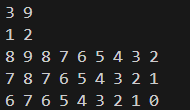
*4) Результати виконаних завдань, тестування та фактично затрачений час*

**Завдання №1 Епік 6 - Практичне завдання**



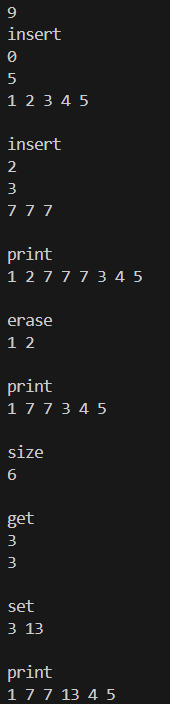
Обидва зайняли годин 5

**Завдання №2 algotester 5v3**



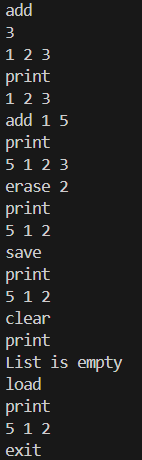
Зайняло 2 год

**Завдання №3 algotester 78v1**



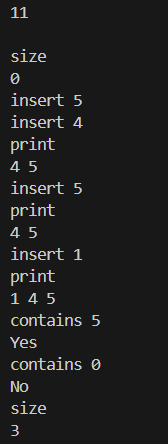
Зайняло 7 год

**Завдання №4 vns 10v8**



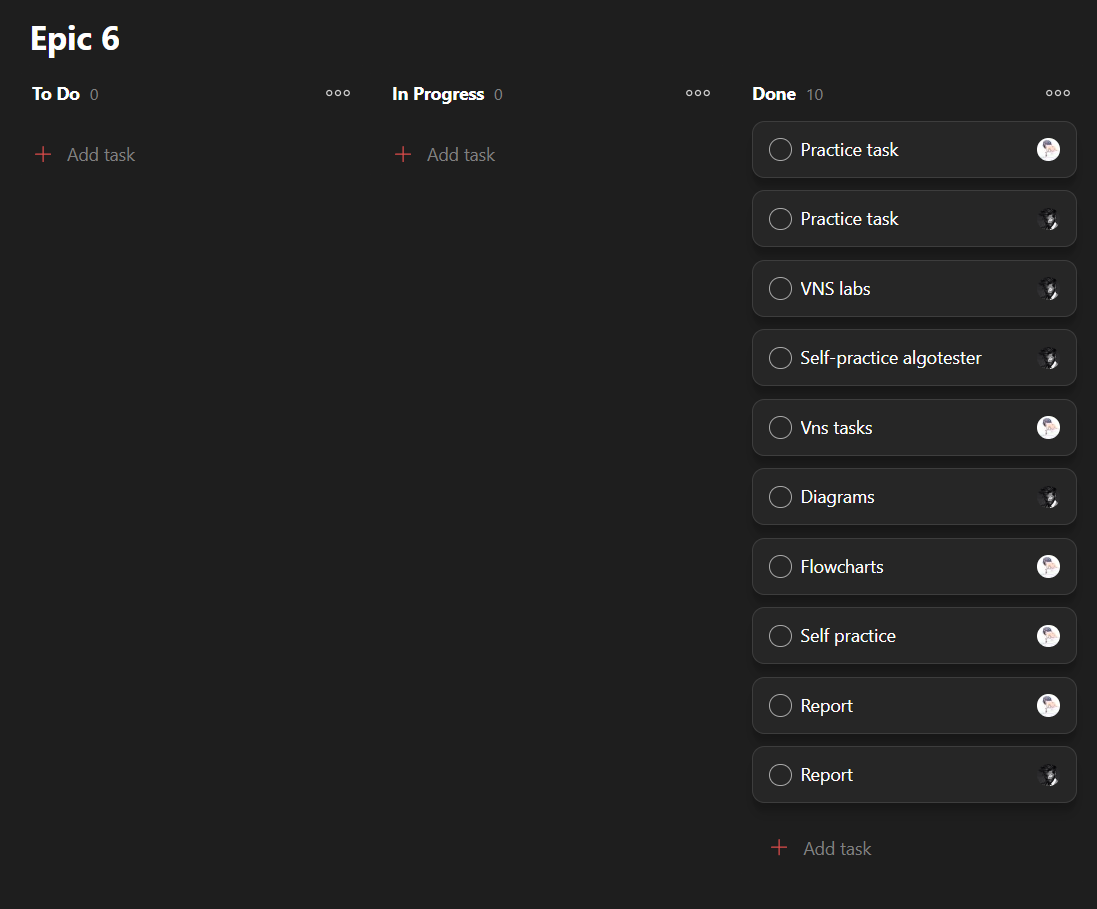
Зайняло 2 год

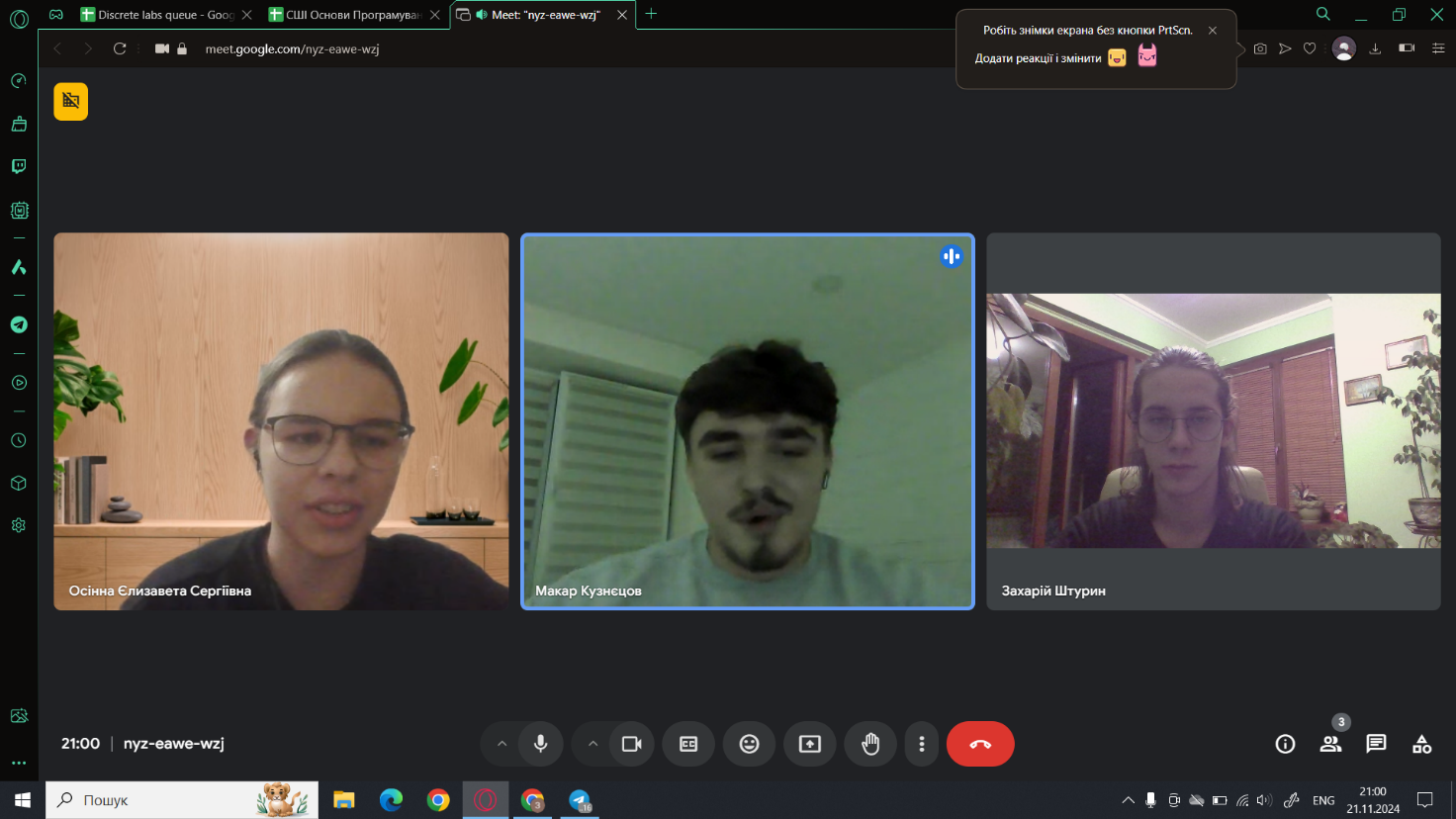
**Завдання №5 self-practice algotester 78v3**



Зайняло 15 хв

*5) Додатково*





**Висновки:** впродовж цього епіку я познайомився з тонною нових структур, навчився їх самостійно реалізовувати та додавати функціонал.