

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:** «Основи програмування»  
до:  
Практичних Робіт до блоку № 6

**Виконав:**  
Студент групи ШІ-13  
Мурашко Владислав Сергійович

Львів 2024

## Тема роботи:

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

## Мета роботи:

Навчитися працювати з динамічними структурами та писати алгоритми для їх обробки.

## Теоретичні відомості:

### 1. Теми, необхідні для виконання роботи:

- Черга.
- Стек.
- Однозв'язний та двозв'язний списки.
- Дерево.

### 2. Джерела використані для ознайомлення з вищезазначеними темами:

- Дивився деякі синтаксичні моменти в чата GPT та інтернеті.
- З усією іншою інформацією я був знайомий з власного досвіду.

## Виконання роботи:

*Опрацювання завдання та вимог до програми та середовища*

### Завдання №1 - Class Practice Work Task 1-3 – Зв'язаний список

#### Задача

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

Умови задачі:

- використовувати цілочисельні значення в списку;
- реалізувати метод реверсу;
- реалізувати допоміжний метод виведення вхідного і обернутого списків;

#### 2. Реалізувати метод порівняння списків `bool compare(Node *h1, Node *h2);`

Умови задачі:

- використовувати цілочисельні значення в списку;
- реалізувати функцію, яка ітеративно проходиться по обох списках і порівнює дані в кожному вузлі;
- якщо виявлено невідповідність даних або якщо довжина списків різна (один список закінчується раніше іншого), функція повертає **false**.

#### 3. Реалізувати метод додавання великих чисел `Node* add(Node *n1, Node *n2);`

Умови задачі:

- використовувати цифри від 0 до 9 для значень у списку;
- реалізувати функцію, яка обчислює суму двох чисел, які збережено в списку; молодший розряд числа записано в голові списка (напр.  $379 \Rightarrow 9 \rightarrow 7 \rightarrow 3$ );
- функція повертає новий список, передані в функцію списки не модифікуються.

## Завдання №2 - Class Practice Work Task 4-5 – Бінарні дерева

### Задача

**4. Реалізувати метод відзеркалення дерева:** `TreeNode *create_mirror_flip(TreeNode *root);`

Умови задачі:

- 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

**5. Реалізувати метод запису кожному батьківському вузлу суму підвузлів:**

`void tree_sum(TreeNode *root);`

Умови задачі:

- 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 №3 – VNS lab 10 варіант 10

### Задача

*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. Додати в нього елемент із заданим номером, знищити K елементів з кінця списку.*

## Завдання №4 – Algotester lab 5 variant 3

### Задача

У вас є карта гори розміром  $N \times M$ .

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.

Клітинки 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 №5 – Algotester lab 7-8 variant 3

### Задача

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 valuevalue - число, яке треба вставити в дерево.
- **Пошук:**  
Ідентифікатор - contains  
Ви отримуєте ціле число valuevalue - 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 <<

## Завдання №6 – Self practice Algotester lab 5 variant 2

### Задача

В пустелі існує незвичайна печера, яка є двохвимірною. Її висота це N, ширина - M.

Всередині печери є пустота, пісок та каміння. Пустота позначається буквою O , пісок S і каміння X;

Одного дня стався землетрус і весь пісок посипався вниз. Він падає на найнижчу клітинку з пустотою, але він не може пролетіти через каміння.

Ваше завдання сказати як буде виглядати печера після землетрусу.

*Дизайн та планована оцінка часу виконання завдань*

**Завдання №1 - Class Practice Work Task 1-3 – Зв’язаний список**

Попередньо розраховував, що це завдання займе 1 годину

**Завдання №1 - Class Practice Work Task 4-5 – Бінарне дерево**

Попередньо розраховував, що це завдання займе 1 годину

**Завдання №3 – VNS lab 10 варіант 10**

Попередньо розраховував, що це завдання займе 1 годину

**Завдання №4 – Algotester lab 5 variant 3**

Дизайн алгоритму завдання знаходиться в папці “Diagrams” під назвою  
algotester\_lab\_5\_variant\_3\_vladyslav\_murashko..drawio

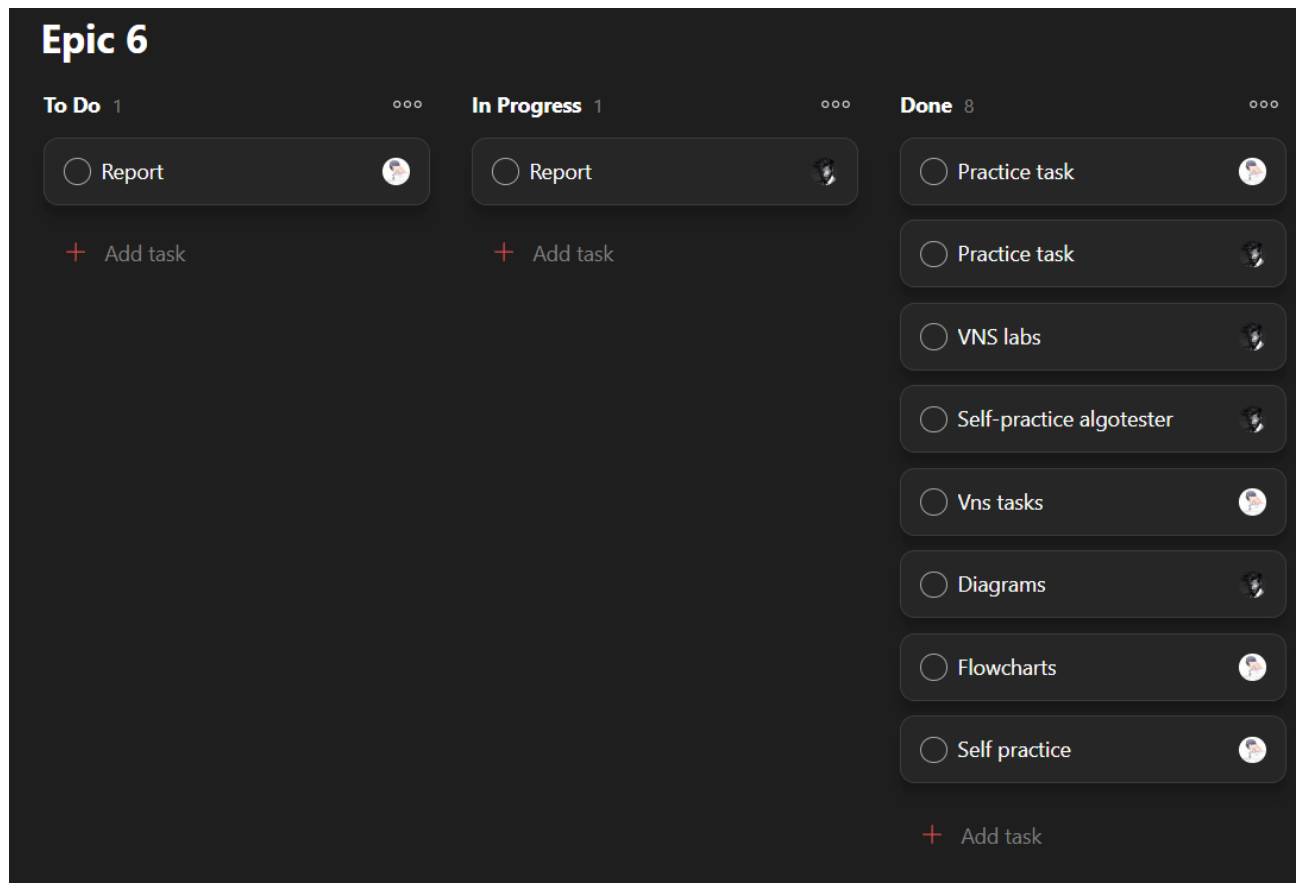
Попередньо розраховував, що це завдання займе 1 годину

**Завдання №5 – Algotester lab 7-8 variant 3**

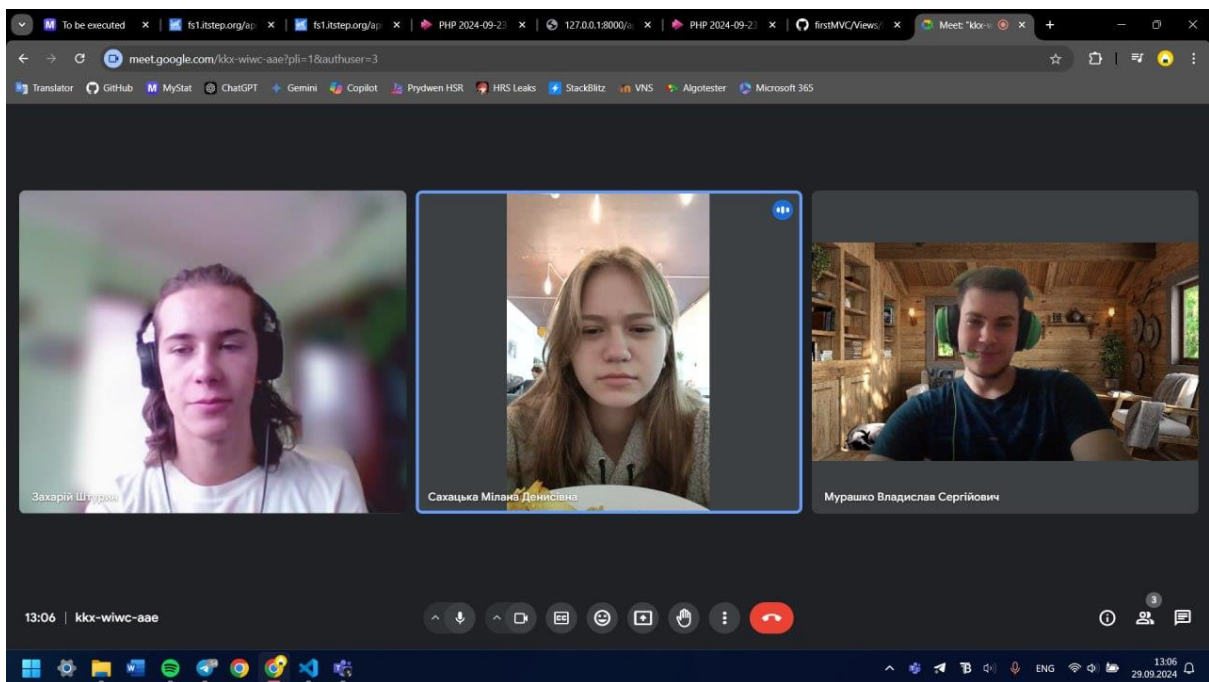
Попередньо розраховував, що це завдання займе 1 годину

**Завдання №6 – Self practice Algotester lab 5 variant 2**

Попередньо розраховував, що це завдання займе 30 хвилин



Командна дошка



Скрін зустрічі нашої команди

*Результати виконаних завдань, тестування та фактично затрачений час*

### **Завдання №1 - Class Practice Work Task 1-3 – Зв'язаний список**

```
list - 5 6 7
!list - 7 6 5
list == list - True
list == !list - False
list + !list - 2 3 3 1
```

Зайняло часу – 1 година

### **Завдання №2 - Class Practice Work Task 4-5 – Бінарне дерево**

```
Root 4
Insert 1
Insert 2
Insert 3
Insert 4
Insert 5
Insert 6
Insert 0
Tree - 0 1 2 3 4 5 6
Mirror - 6 5 4 3 2 1 0
Tree converted to sum - 0 3 3 3 9 6 6
```

Зайняло часу – 1 година

### **Завдання №3 – VNS lab 10 варіант 10**

```
5
10 5
10 18 17 16 15 5
10 18 17
List is empty
10 18 17
```

```
list.txt
1 10 18 17
```

Зайняло часу – 30 хвилин

### Завдання №4 – Algotester lab 5 variant 3

3 4	3 9
1 2	3 2
3 4 3 2	6 7 6 5 4 3 2 1 0
2 3 2 1	7 8 7 6 5 4 3 2 1
1 2 1 0	8 9 8 7 6 5 4 3 2

Зайняло часу – 30 хвилин

### Завдання №5 – Algotester lab 7-8 variant 3

3	6
insert 1	print
insert 2	
print	insert 2
1 2	insert 2
	insert 3
	remove 2
	print
	3

Зайняло часу – 30 хвилин

### Завдання №6 – Self practice Algotester lab 5 variant 2

4 4	4 3
S00S	S00
X000	00X
S000	000
0000	00S
-----	-----
S000	000
X000	00X
0000	000
S00S	S0S

Зайняло часу – 30 хвилин



**Висновок:** Я навчився працювати з динамічними структурами та писати алгоритми для їх обробки.