Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту

A blue and white logo

Description automatically generated

**Звіт**

**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

Бойко Роман Андрійович

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

### **Мета роботи:** Навчитися користуватися різними видами динамічних структур.

### **Теоретичні відомості:**

* Бінарне дерево пошуку
* Списки
* Черга

Джерела:

* [C++ ⦁ Теорія ⦁ Урок 144 ⦁ ADT ⦁ Бінарне дерево](https://www.youtube.com/watch?v=qBFzNW0ALxQ)
* [C++ ⦁ Теорія ⦁ Урок 139 ⦁ ADT ⦁ Однозв'язний список](https://www.youtube.com/watch?v=-25REjF_atI&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=139)
* [C++ ⦁ Теорія ⦁ Урок 140 ⦁ ADT ⦁ Двозв'язний список](https://www.youtube.com/watch?v=QLzu2-_QFoE&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=140)
* [C++ ⦁ Теорія ⦁ Урок 141 ⦁ ADT ⦁ Стек](https://www.youtube.com/watch?v=ZYvYISxaNL0&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=141)
* [C++ ⦁ Теорія ⦁ Урок 142 ⦁ ADT ⦁ Черга](https://www.youtube.com/watch?v=Yhw8NbjrSFA&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=142)

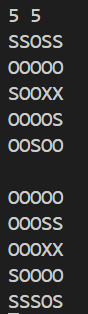
### **Виконання роботи**

**Особистий варіант - VNS Lab 10 - 1, Algotester Lab 7-8 - 2,3, Algotester Lab 5 - 2**

**Завдання 1: Algotester Lab 5**

В пустелi iснує незвичайна печера, яка є двохвимiрною. Її висота це N, ширина - M. Всерединi печери є пустота, пiсок та камiння. Пустота позначається буквою , пiсок S i камiння X; Одного дня стався землетрус i весь пiсок посипався вниз. Вiн падає на найнижчу клiтинку з пустотою, але вiн не може пролетiти через камiння. Ваше завдання сказати як буде виглядати печера пiсля землетрусу

**Вивід в терміналі:**

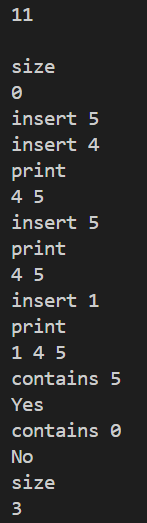
****

**Час виконання завдання ~ 30 хв**

**Завдання 2: Self Practice Task Algotester Lab 7-8 variant 3**

Ваше завдання - власноруч реалiзувати структуру даних "Двiйкове дерево пошуку". Ви отримаєте Q запитiв, кожен запит буде починатися зi слова-iдентифiкатора, пiсля якого йдуть його параметри. Вам будуть поступати запити такого типу: • Вставка: Iдентифiкатор - insert Ви отримуєте цiле число value - число, яке треба вставити в дерево. • Пошук: Iдентифiкатор - contains Ви отримуєте цiле число value - число, наявнiсть якого у деревi необхiдно перевiрити. Якщо value наявне в деревi - ви виводите Y es, у iншому випадку No. • Визначення розмiру: Iдентифiкатор - size Ви не отримуєте аргументiв. Ви виводите кiлькiсть елементiв у деревi. • Вивiд дерева на екран Iдентифiкатор - print Ви не отримуєте аргументiв. Ви виводите усi елементи дерева через пробiл. Реалiзувати використовуючи перегрузку оператора <<

**Вивід в терміналі:**

****

**Час виконання завдання ~ 2 години**

**Завдання 3: VNS Lab 10**

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

однонаправлений список. Знищити з нього елемент із заданим номером,

додати елемент із заданим номером;

**Вивід в терміналі:**

****

**Час виконання завдання ~ 1 година**

**Завдання 4: Practice Task 1-3**

## **Задача №1 - Реверс списку (Reverse list)**

***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;

## **Задача №2 - 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***.

## **Задача №3 – Додавання великих чисел**

Node\* add(Node \*n1, Node \*n2);

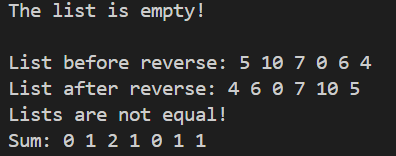
*Умови задачі:*

- 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 ⟹ 9→7→3);

- функція повертає новий список, передані в функцію списки не модифікуються.

**Результат виконання програми:**

****

**Час виконання завдання ~ 2.5 години**

**Завдання 5: Practice Task 4-5**

## **Задача №4 - Віддзеркалення 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

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

void tree\_sum(TreeNode \*root);

*Умови задачі:*

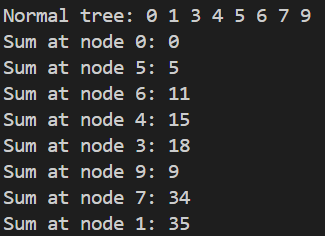
- використовувати цілочисельні значення у вузлах дерева;

- 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

**Результат виконання програми:**

****

**Час виконання завдання ~ 1.5 години**

**Завдання 6: Algotester Lab 7-8 Variant 2**

Ваше завдання - власноруч реалiзувати структуру даних "Динамiчний масив". Ви отримаєте Q запитiв, кожен запит буде починатися зi слова-iдентифiкатора, пiсля якого йдуть його аргументи. Вам будуть поступати запити такого типу:

• Вставка: Iдентифiкатор - insert Ви отримуєте цiле число index елемента, на мiсце якого робити вставку. Пiсля цього в наступному рядку рядку написане число N - розмiр масиву, який треба вставити. У третьому рядку N цiлих чисел - масив, який треба вставити на позицiю index.

• Видалення: Iдентифiкатор - erase Ви отримуєте 2 цiлих числа - index, iндекс елемента, з якого почати видалення та n - кiлькiсть елементiв, яку треба видалити.

• Визначення розмiру: Iдентифiкатор - size Ви не отримуєте аргументiв. Ви виводите кiлькiсть елементiв у динамiчному масивi.

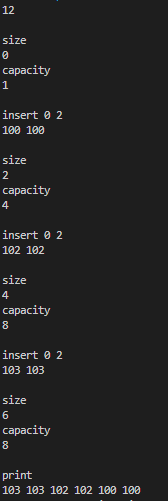
• Визначення кiлькостi зарезервованої пам’ятi: Iдентифiкатор - capacity Ви не отримуєте аргументiв. Ви виводите кiлькiсть зарезервованої пам’ятi у динамiчному масивi. Ваша реалiзацiя динамiчного масиву має мати фактор росту (Growth factor) рiвний 2.

• Отримання значення i-го елементу Iдентифiкатор - get Ви отримуєте цiле число - index, iндекс елемента. Ви виводите значення елемента за iндексом. Реалiзувати використовуючи перегрузку оператора []

• Модифiкацiя значення i-го елементу Iдентифiкатор - set Ви отримуєте 2 цiлих числа - iндекс елемента, який треба змiнити, та його нове значення. Реалiзувати використовуючи перегрузку оператора []

• Вивiд динамiчного масиву на екран Iдентифiкатор - print Ви не отримуєте аргументiв. Ви виводите усi елементи динамiчного масиву через пробiл. Реалiзувати використовуючи перегрузку оператора <<

**Результат виконання програми:**

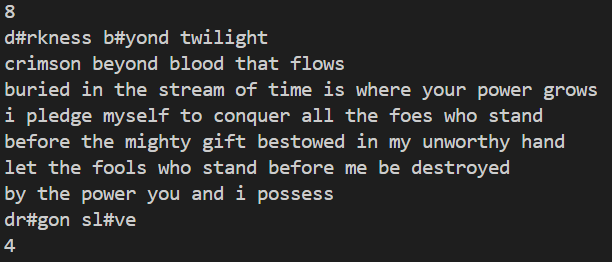
****

**Час виконання завдання ~ 3 години**

**Завдання 7: Self Practice Task Algotester Lab 3 variant 1**

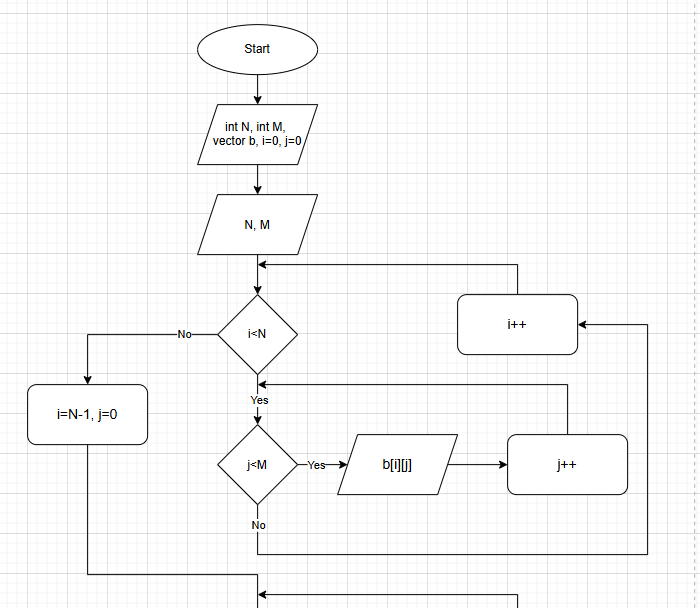
Ви з’явились у свiтi пiд назвою Атод посеред Пустелi Безправ’я. Так сталось, що Ви попали саме в той час i мiсце, де ведеться битва мiж чаклункою Лiною i темними силами, якi хочуть знищити цей свiт. На жаль, трапилась халепа, бо деякi слова iз книги чар були пошкодженi пiд час битви. Одне таке слово можна вiдновити виконавши ритуал зцiлення над пошкодженими буквами. Ритуал зцiлення можна виконати на всiх пiдряд розташованих пошкоджених буквах. Вам не залишається нiчого iншого як допомогти Лiнi вiдновити цi слова i сказати скiльки мiнiмально треба провести таких ритуалiв, щоб прочитати одне з наймогутнiших у цьому свiтi заклять - Поневолення Дракона!

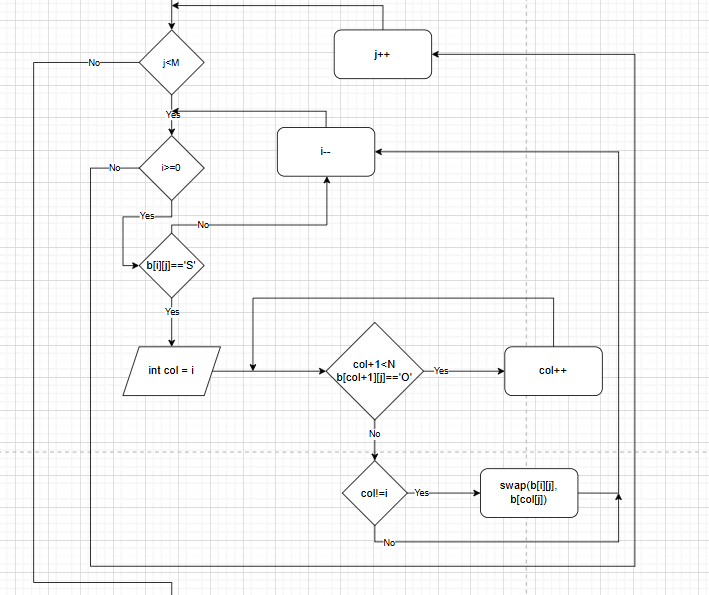
**Результат виконання програми:**

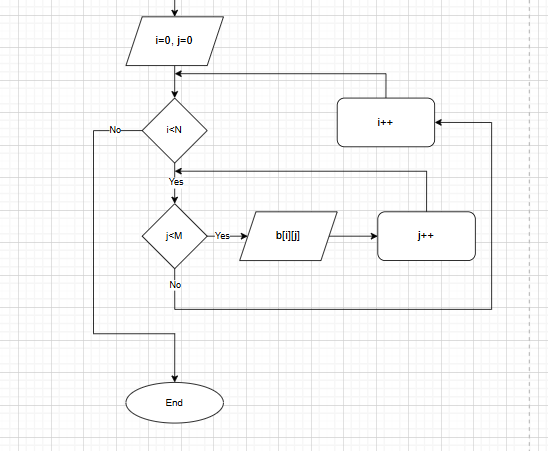


**Час виконання завдання ~ 15 хвилин**

**Блок-схема до задачі Algotester Lab 5**

****





**Висновок:**

**У цьому епіку я навчився працювати з різними видами динамічних структур**