Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт № 6**

***з дисципліни:*** «Мови та парадигми програмування»

***з розділу***: «Динамічні структури (Черга, Стек, Списки, Дерево). Алгоритми обробки динамічних структур.»

до:

ВНС Лабораторної Роботи № 10

Алготестер Лабораторної Роботи № 5

Алготестер Лабораторної Роботи № 7-8

ПрактичнихРобіт № 6

***Виконав:***

студент групи ШІ-11

Ільницький Олександр Ігорович

# **Тема роботи:**

Знайомство з динамічними структурами (Списки, Дерево) і алгоритмами обробки динамічних структур.

# **Мета роботи:**

Навчитись створювати динамічні структури, алгоритми для їх обробки.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Класи та структури в C++
* Тема №2: Лінійні списки та бінарне дерево в С++

1. Індивідуальний план опрацювання теорії:

* Тема №1: Тема №1: Класи та структури в C++
  + Джерела Інформації
    - Лекції та практичні заняття
    - Власний досвід
  + Що опрацьовано:
    - Ознайомився з можливостями, які відкривають класи та структури. Навчився будувати свої класи та структури.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 07.12.2023
  + Звершення опрацювання теми: 13.12.2023
* Тема №2: Лінійні списки та бінарне дерево в С++
  + Джерела Інформації
    - <https://www.youtube.com/watch?v=-StYr9wILqo&t=589s&ab_channel=CodeBeauty>
    - <https://www.youtube.com/watch?v=UHxtjVsOTHc&t=734s&ab_channel=CodeBeauty>
    - Лекції та практичні заняття
  + Що опрацьовано:
    - Ознайомився з структурою Node, та навчився будувати зв’язні списки і дерева з її допомогою.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 07.12.2023
  + Звершення опрацювання теми: 13.12.2023

# **Виконання роботи:**

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

Завдання №1 “**Programming: VNS Lab 10 Task 1**”

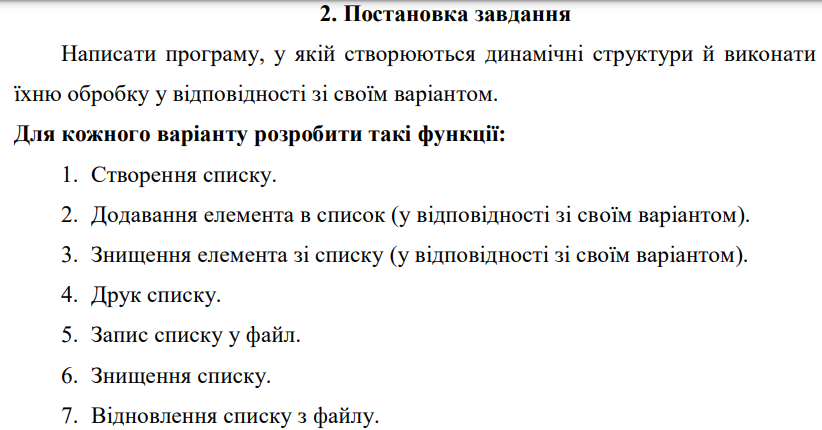
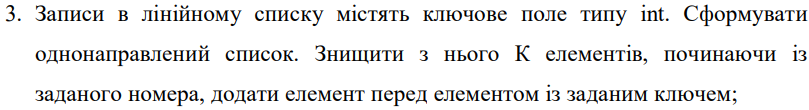
* Варіант: 3
* 
* 

Рисунок : VNS Lab 10 Task

Завдання №2 **“Programming: Algotester Lab 5 Task 1**”

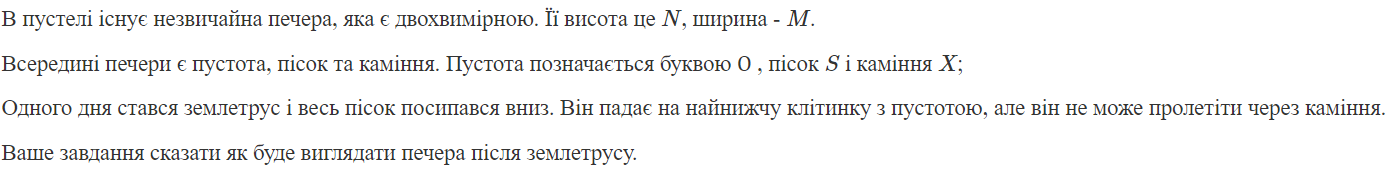
* Варіант: 2
* 

Рисунок : Algotester Lab 5 Task

Завдання №3 “**Programming: Algotester Lab 7-8 Task 1**”

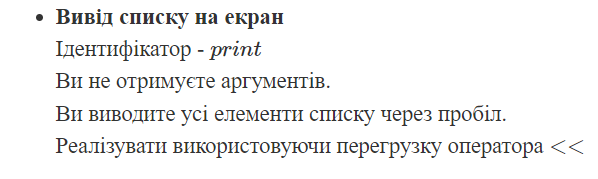
* Варіант: 1
* 
* 

Рисунок : Algotester Lab 7-8 Task 1

Завдання №4 “**Programming: Algotester Lab 7-8 Task 2**”

* Варіант: 1
* 

Рисунок : Algotester Lab 7-8 Task 2

Завдання №5 “**Programming: Class Practice Task 1 (1-3)**”

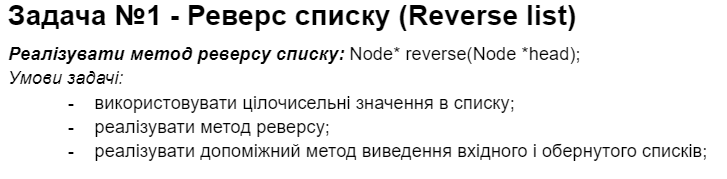
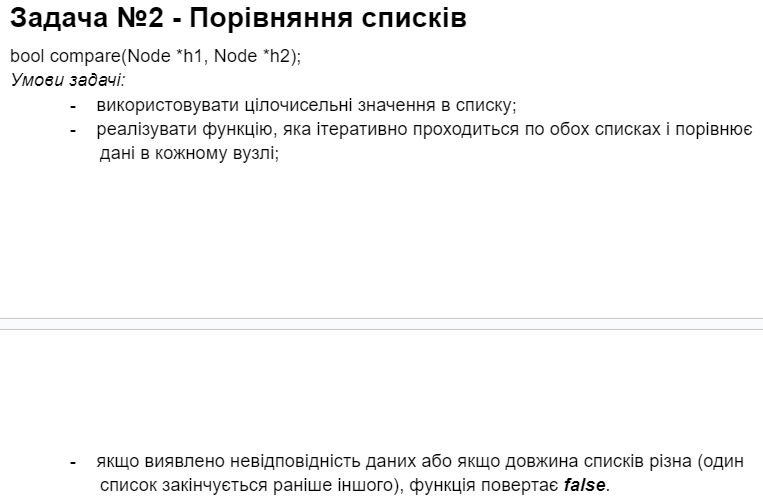
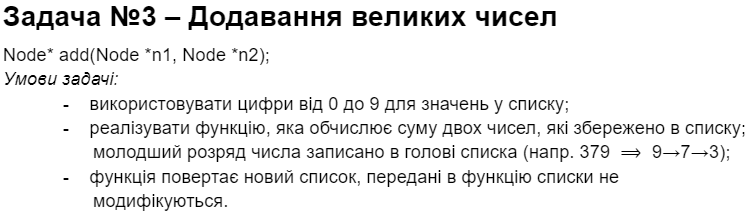
* 
* 
* 

Рисунок : Class Practice Task 1

Завдання №6 “**Programming: Class Practice Task 2 (4-5)**”

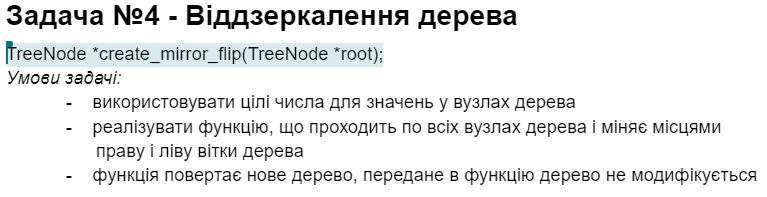
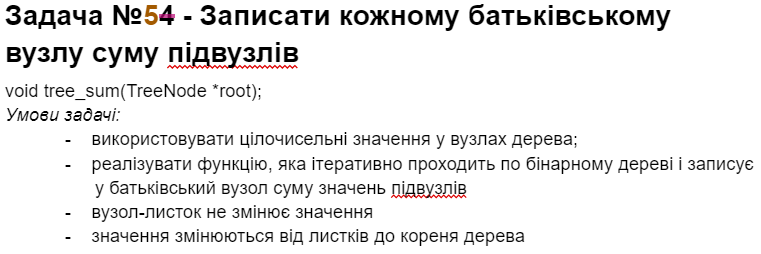
* 
* 

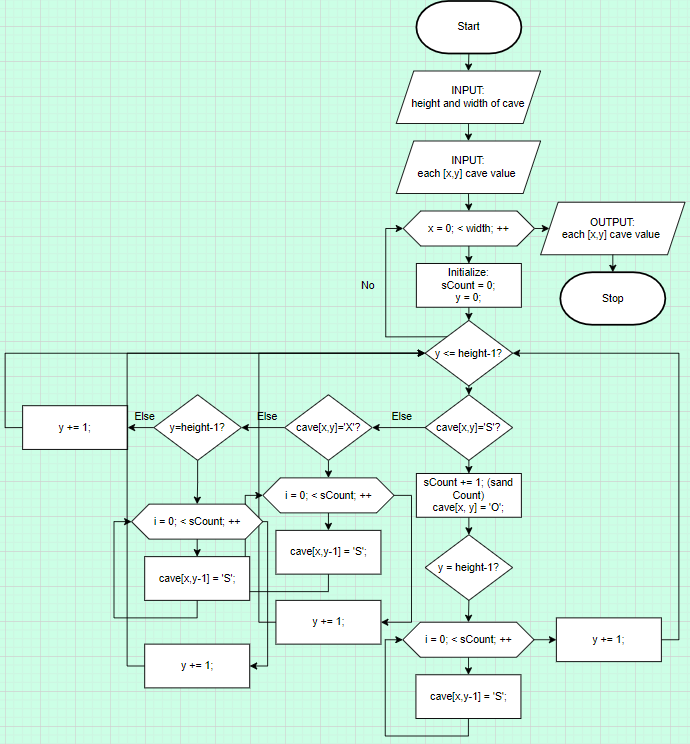
Рисунок : Class Practice Task 2

## **2. Дизайн та планована оцінка часу виконання завдань:**

Завдання №1 “**Programming: VNS Lab 10 Task 1**”

* Теоретично необхідний час для виконання: 60 хв

Завдання №2 “**Programming: Algotester Lab 5 Task 1**”

* Теоретично необхідний час для виконання: 50 хв
* 

Завдання №3 “**Programming: Algotester Lab 7-8 Task 1**”

* Теоретично необхідний час для виконання: 75 хв

Завдання №4 “**Programming: Algotester Lab 7-8 Task 2**”

* Теоретично необхідний час для виконання: 10 хв

Завдання №5 “**Programming:  Class Practice Task 1 (1-3)**”

* Теоретично необхідний час для виконання: 60 хв

Завдання №6 “**Programming:  Class Practice Task 2 (4-5)**”

* Теоретично необхідний час для виконання: 60 хв

**Теоретичний час на виконання всіх задач: 5 годин 35 хв**

## **3. Код програм з посиланням на зовнішні ресурси:**

Завдання №1 “**Programming: VNS Lab 10 Task 1**”

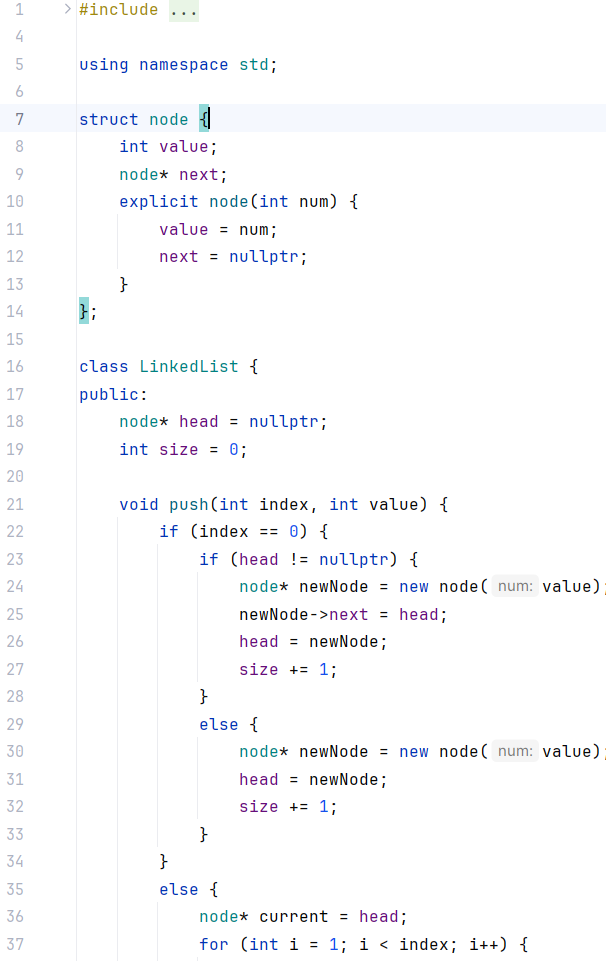
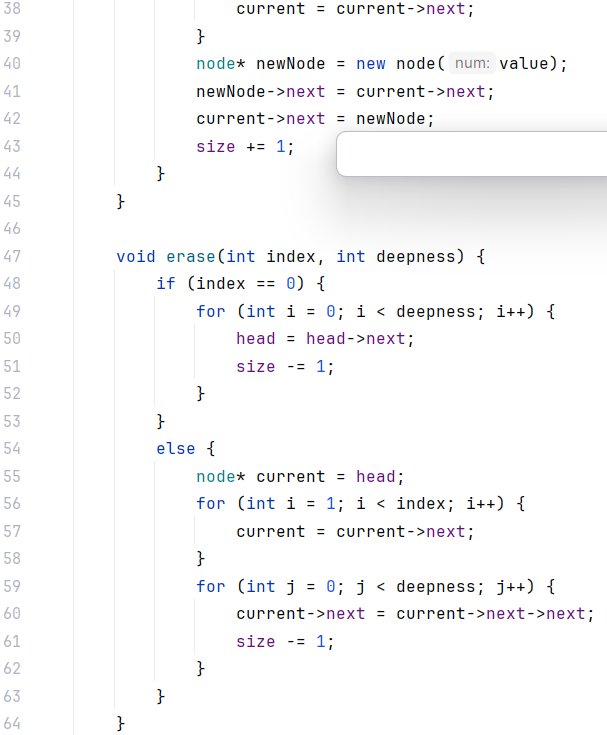
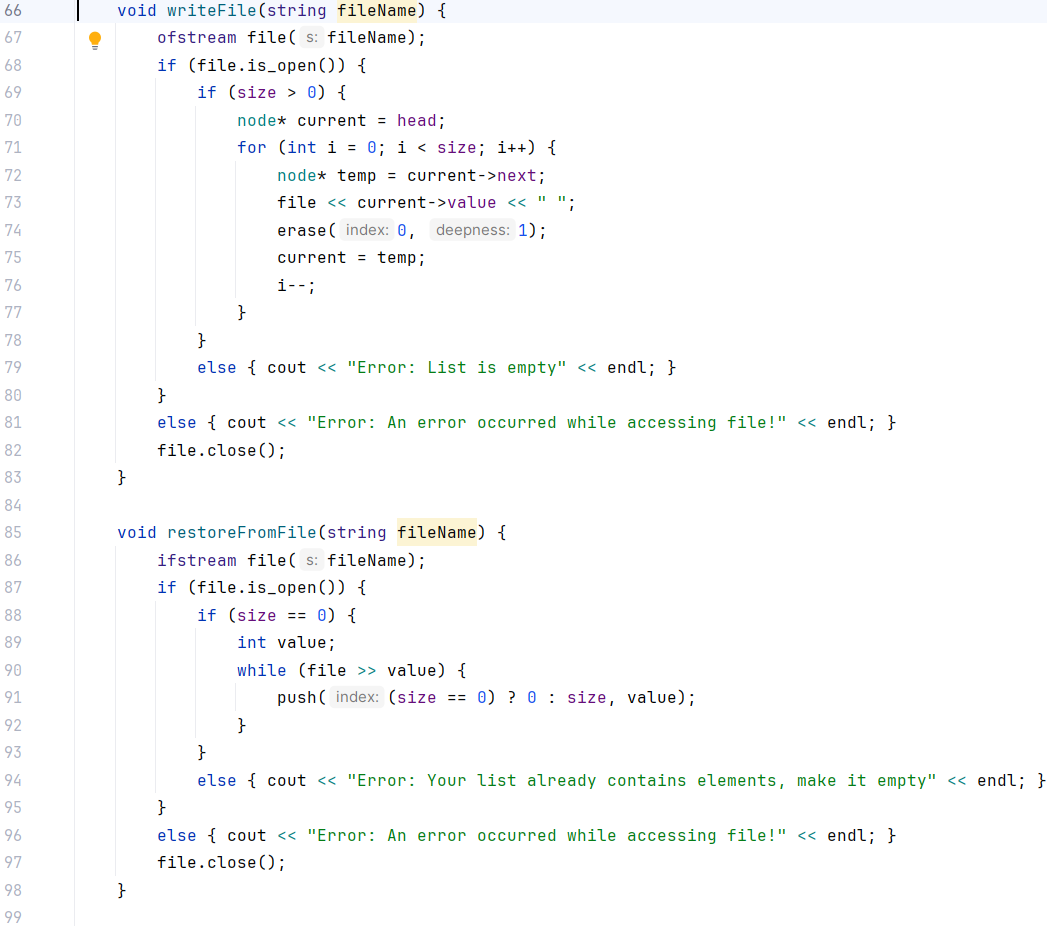
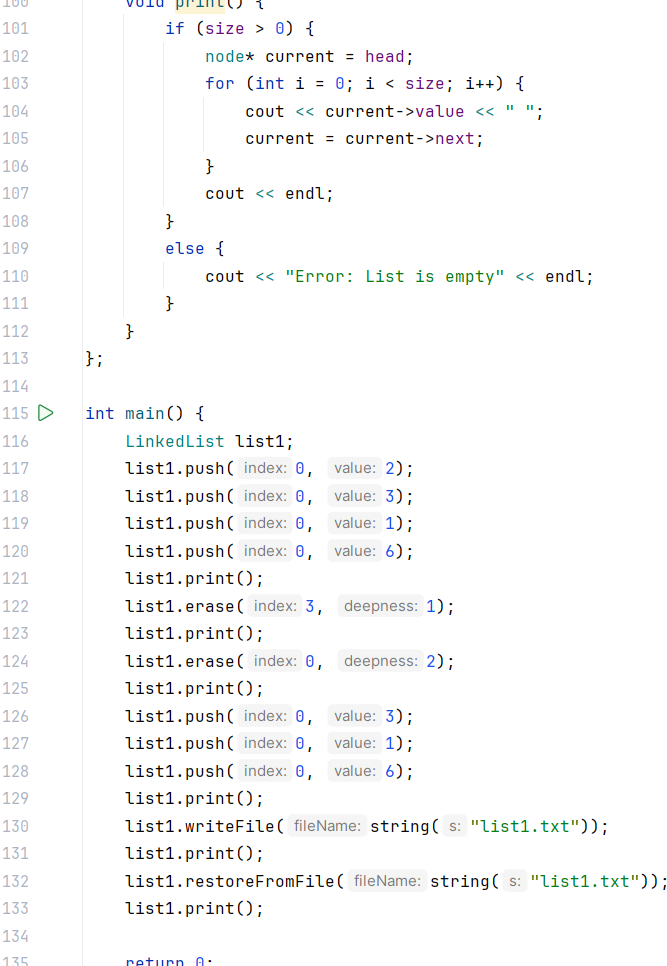
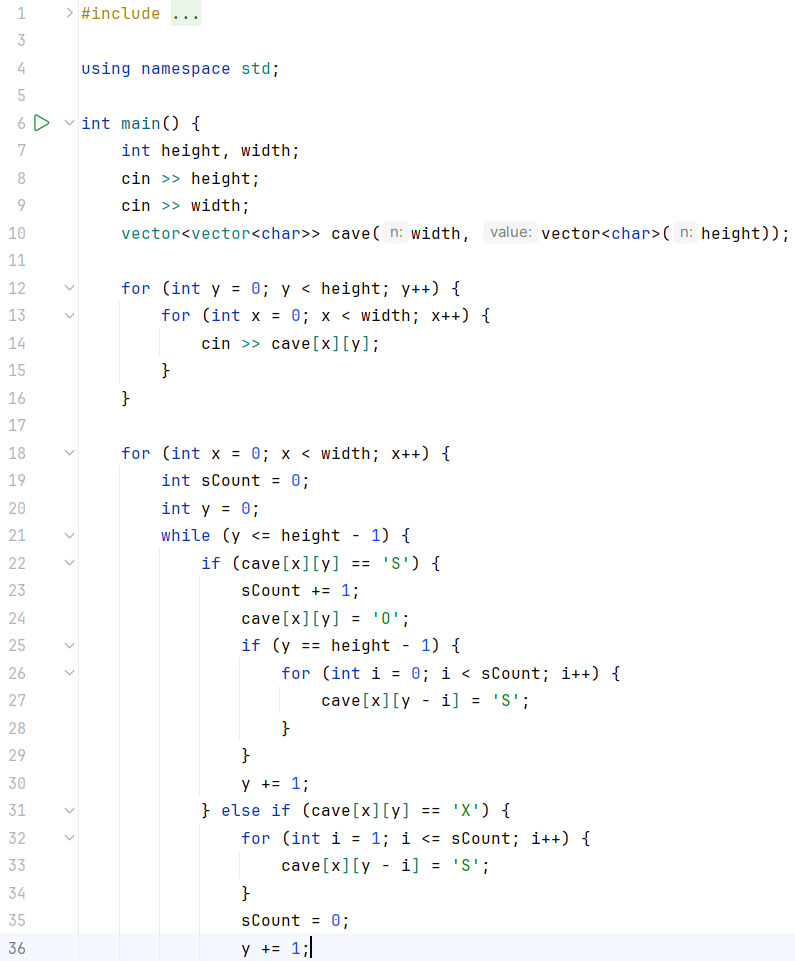
* Написаний код для виконання задачі:
* 
* 
* 
* 

Рисунок : VNS Lab 10 Code

Завдання №2 “**Programming: Algotester Lab 5 Task 1**”

* Написаний код для виконання задачі:



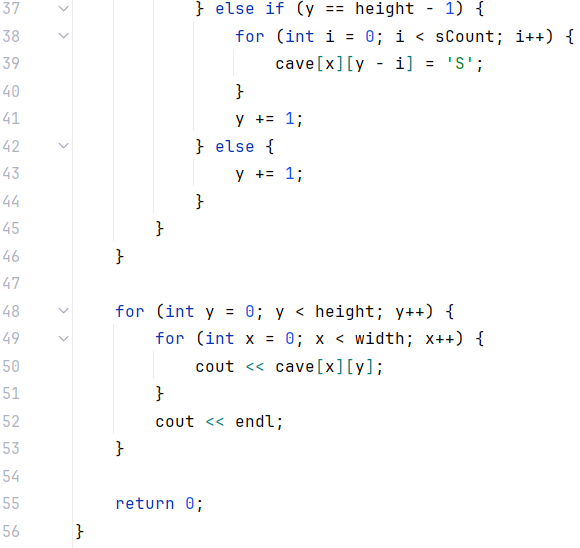
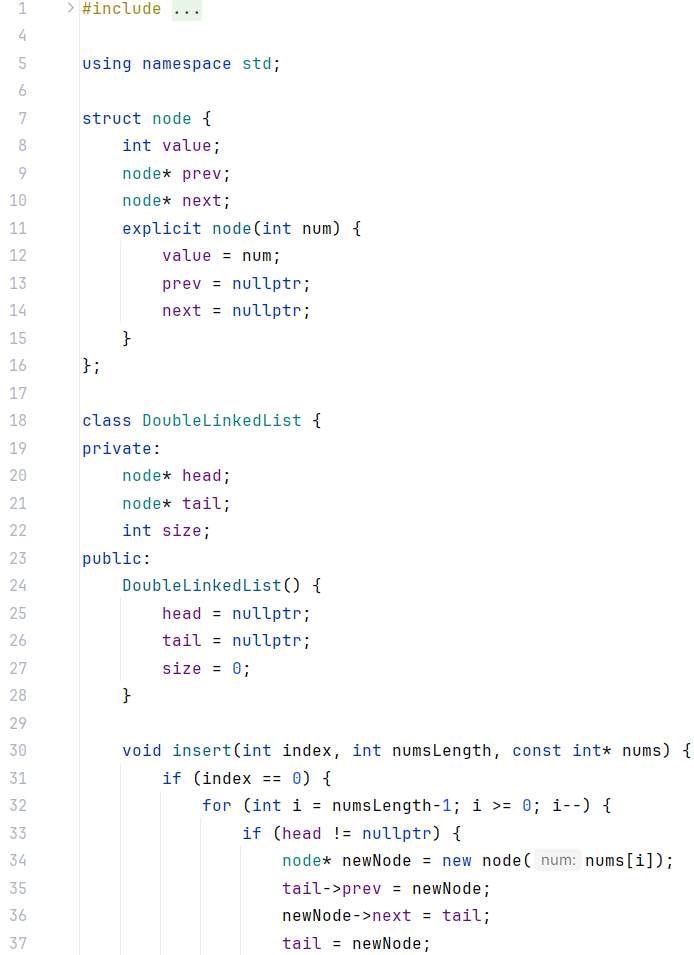
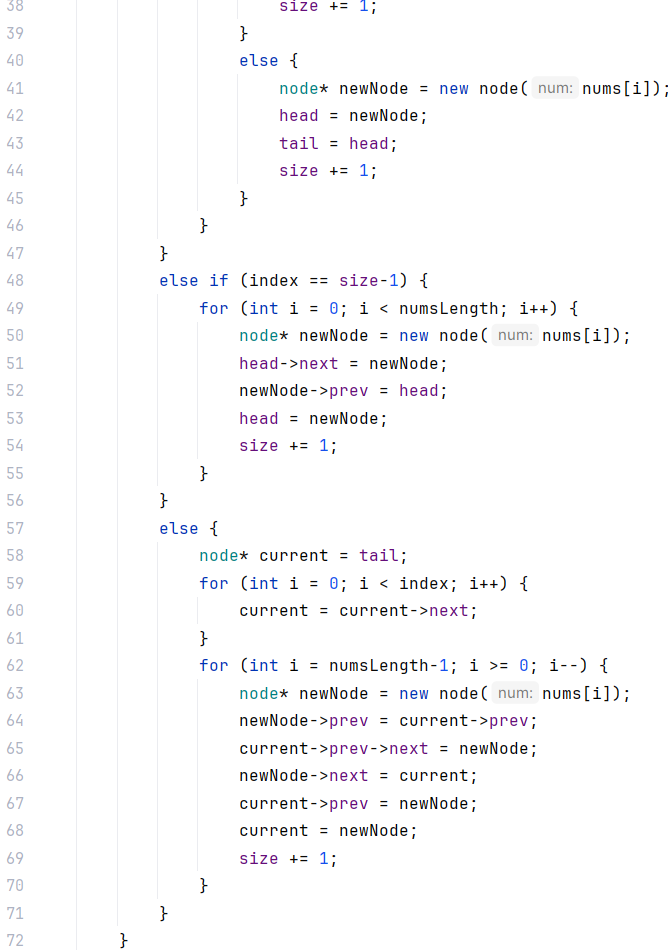


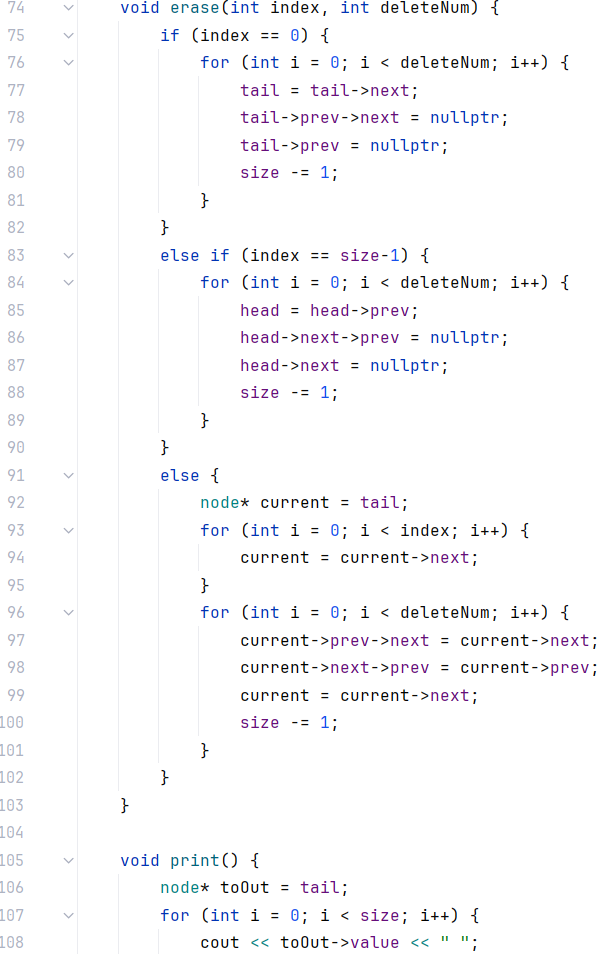
Рисунок : Algotester Lab 5 Code

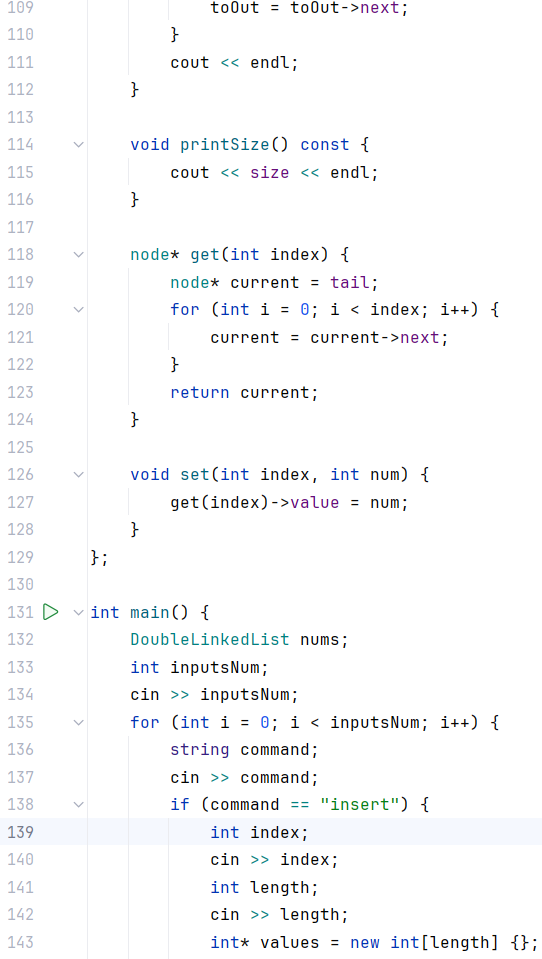
Завдання №3 “**Programming: Algotester Lab 7-8 Task 1**”

* Написаний код для виконання задачі:









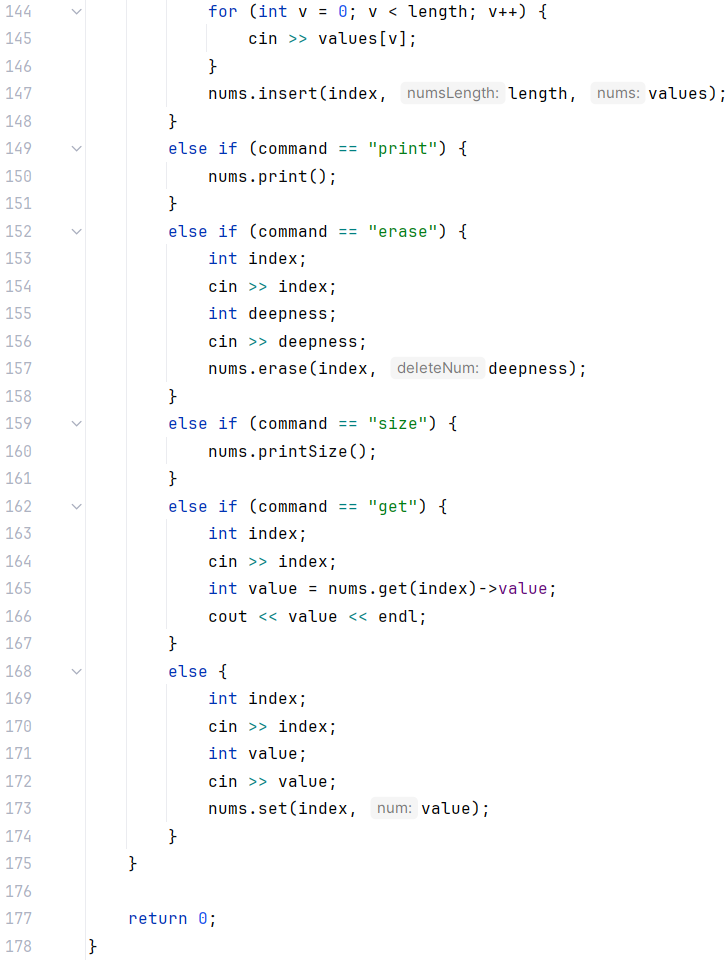


Рисунок : Algotester Lab 7-8 Task 1 Code

Завдання №4 “**Programming: Algotester Lab 7-8 Task 2**”

* Зміни в коді для виконання задачі:

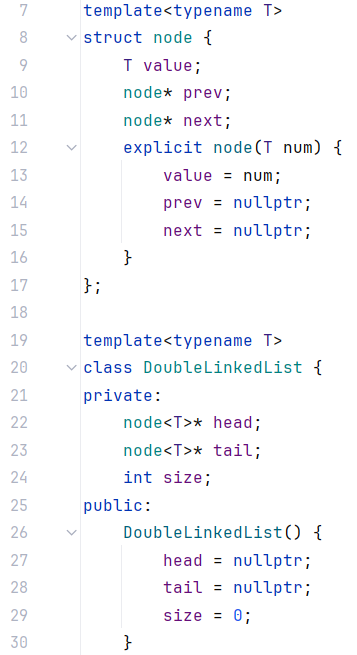


Рисунок : Algotester Lab 7-8 Task 2 Code Changes

* Замінено вхідні дані в інсерт на вектор з темплейт тайпом
* Замінено всі node\* на node<T>\*

Завдання №5 “**Programming:  Class Practice Task 1 (1-3)**”

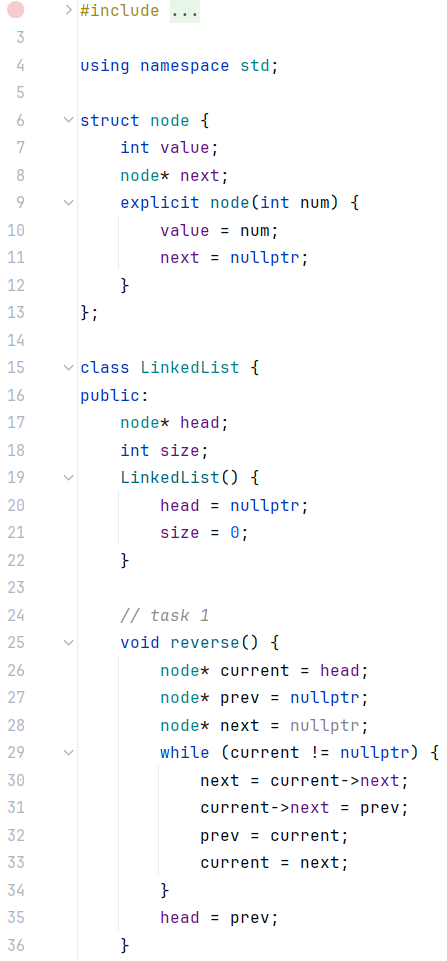
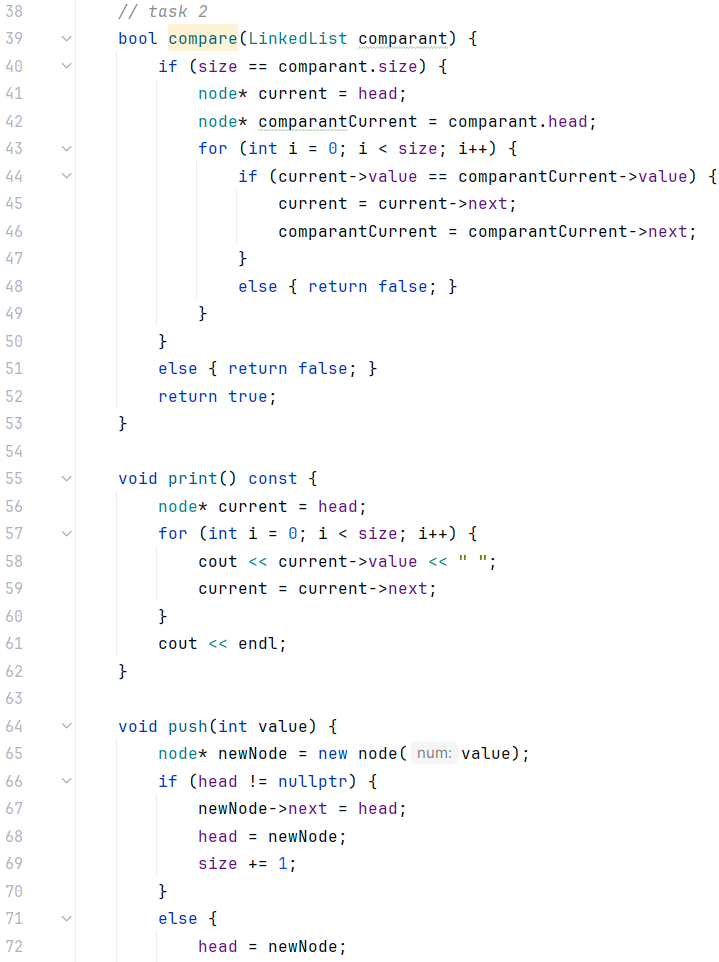
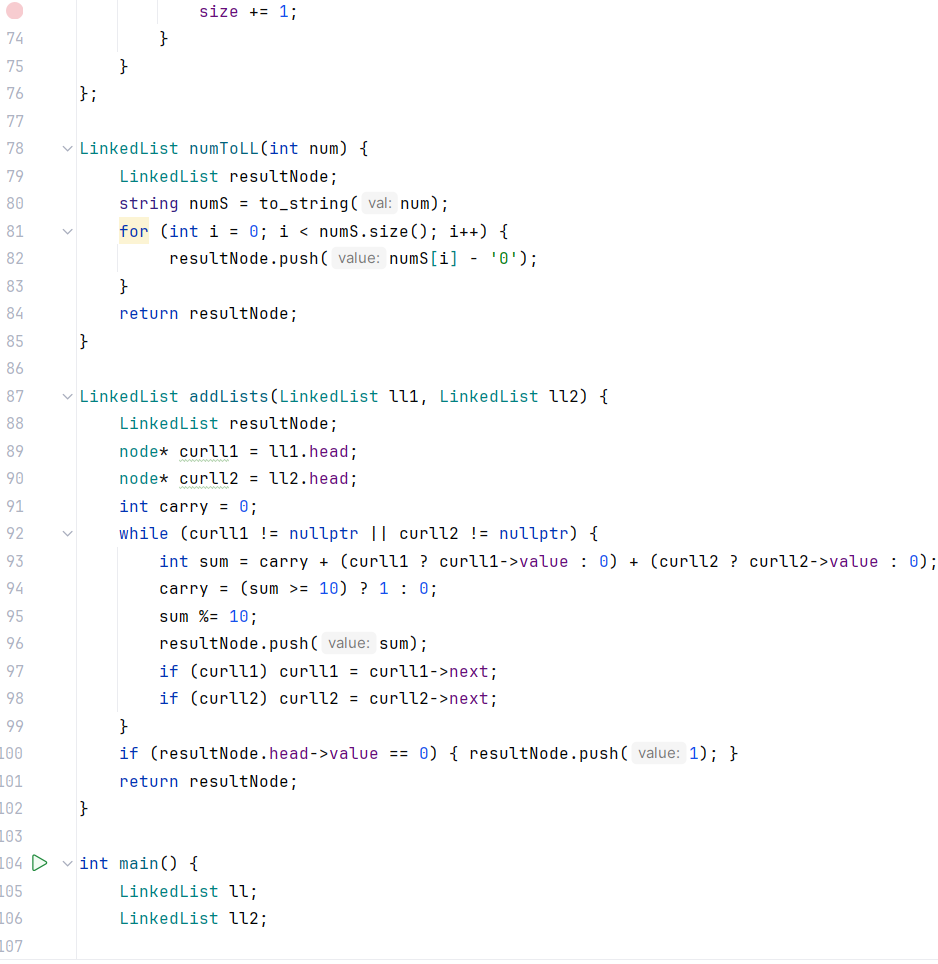
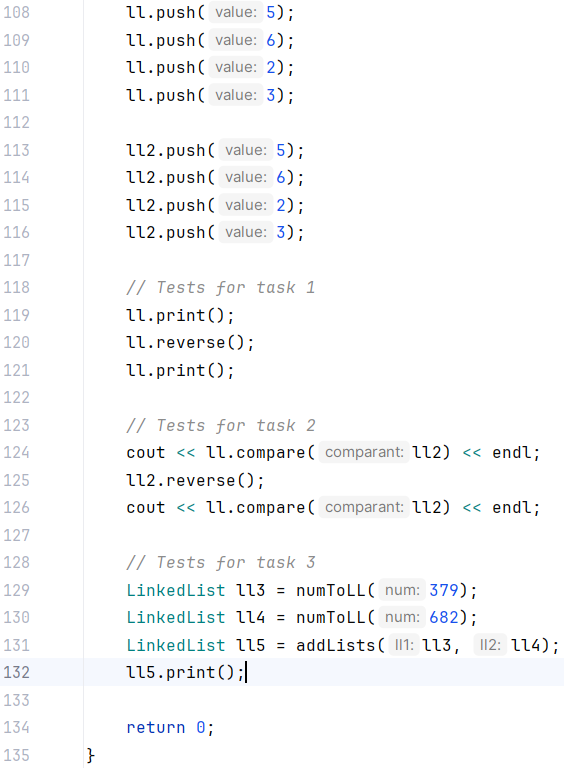
* Написаний код для виконання задачі:
* 
* 
* 
* 

Рисунок : Class Practice Task 1 (1-3) Code

Завдання №5 “**Programming:  Class Practice Task 2 (4-5)**”

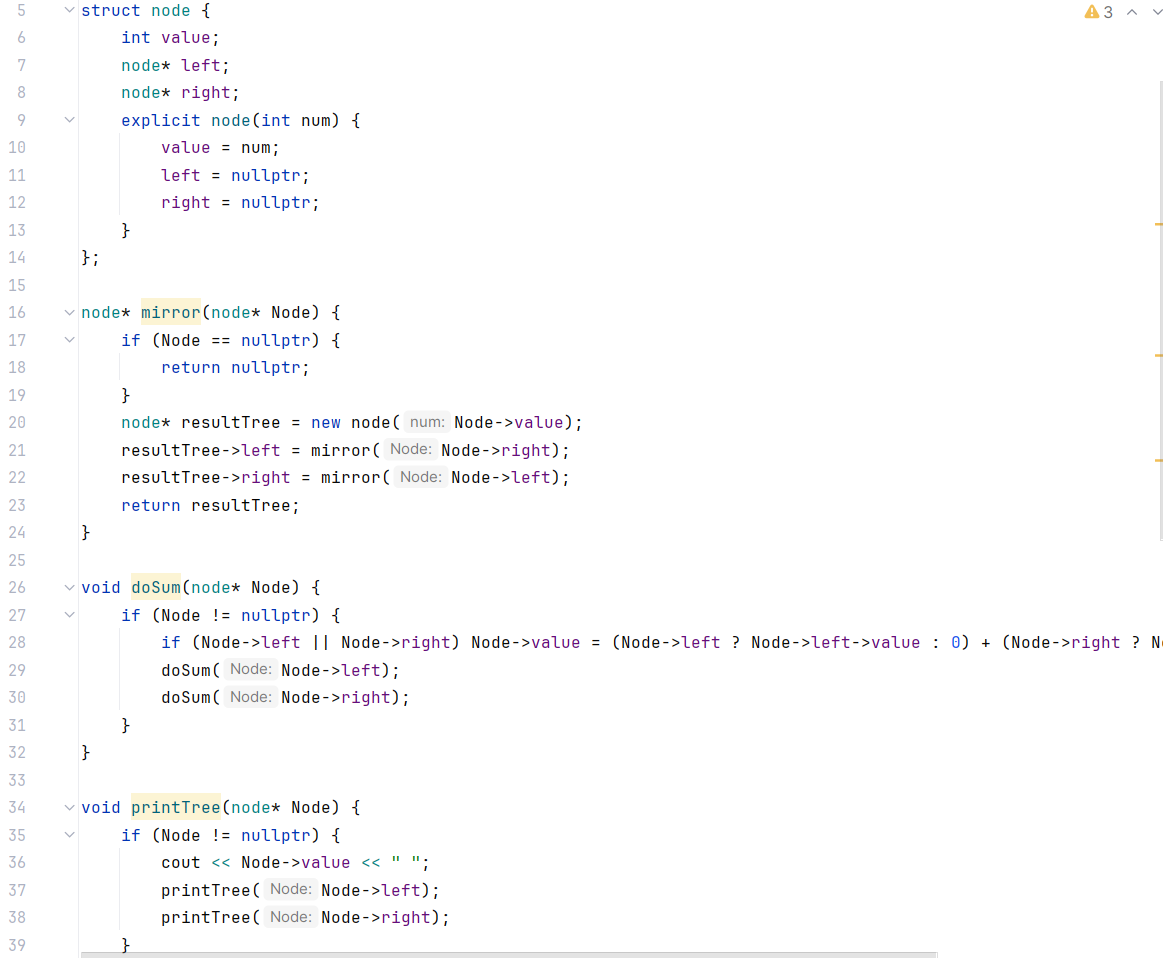
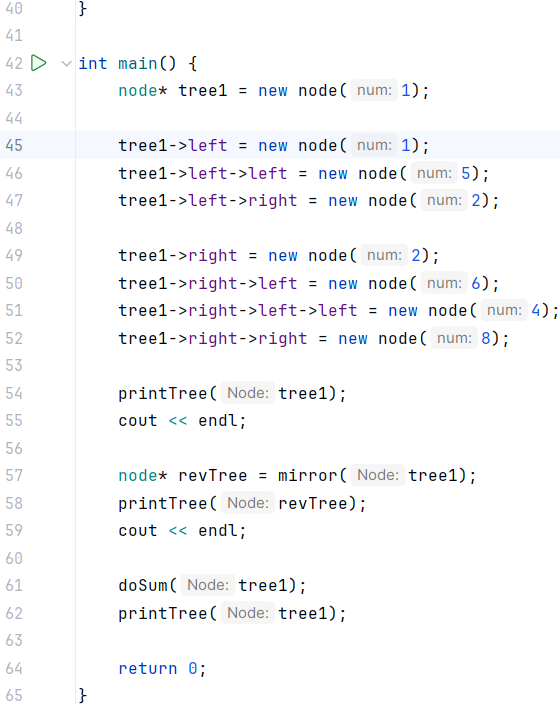
* Написаний код для виконання задачі:
* 
* 

Рисунок : Class Practice Task 2 (4-5) Code

## **4. Результати виконання завдань та фактично затрачений час:**

Завдання №1 “**Programming: VNS Lab 10 Task 1**”

* Результат виконання коду:

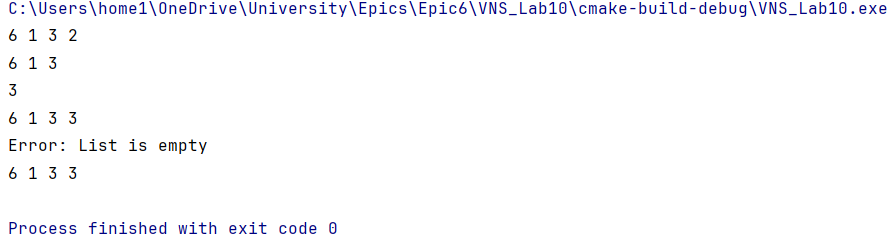


Рисунок : VNS Lab 10 Result

* Практично необхідний час для виконання: 60 хв

Завдання №2 “**Programming: Algotester Lab 5 Task 1**”

* Результат виконання коду:



Рисунок : Algotester Lab 5 Result

* Практично необхідний час для виконання: 100 хв

Завдання №3 “**Programming: Algotester Lab 7-8 Task 1**”

* Результат виконання коду:





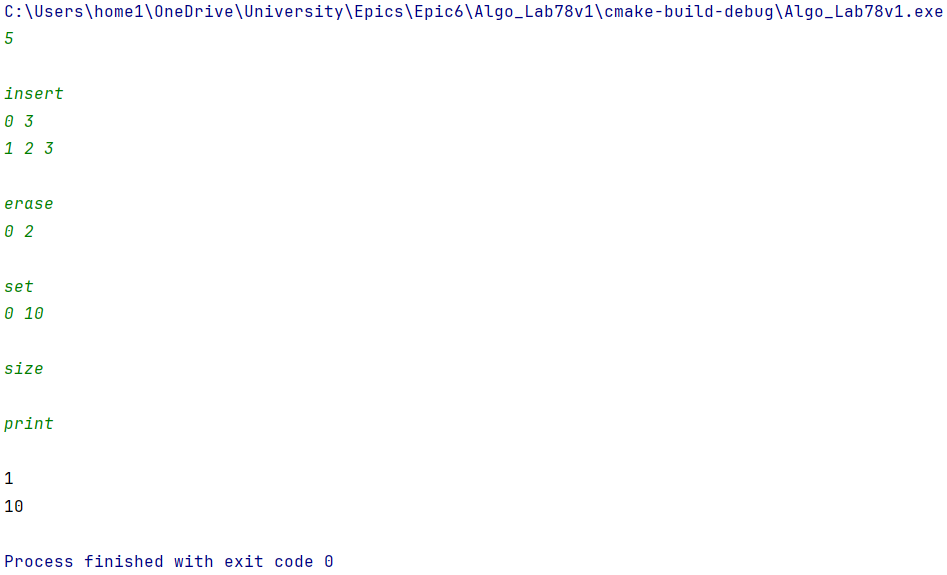


Рисунок : Algotester Lab 7-8 Result

* Практично необхідний час для виконання: 75 хв

Завдання №4 “**Programming: Algotester Lab 7-8 Task 2**”

* Результат виконання коду:

Та ж картина що і в попередньому таску.

* Практично необхідний час для виконання: 5 хв

Завдання №5 “**Programming:  Class Practice Task Task 1 (1-3)**”

* Результат виконання коду:

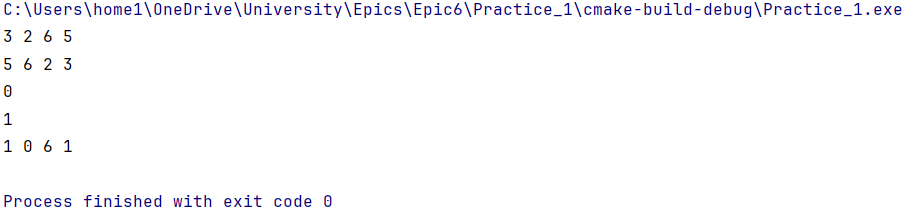


Рисунок : Algotester Practice Task 1 Result

* Практично необхідний час для виконання: 50 хв

Завдання №6 “**Programming:  Class Practice Task Task 2 (4-5)**”

* Результат виконання коду:

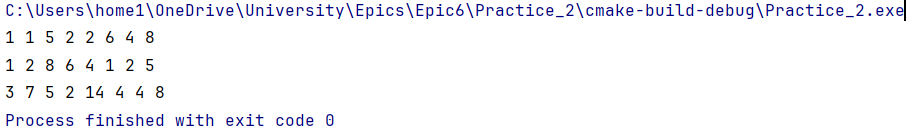


Рисунок : Class Practice Task 2 Result

* Практично необхідний час для виконання: 60 хв

**Практично необхідний час на виконання всіх завдань: 6 годин 10 хв.**

# **Висновки:**

Получив практичні навички з використання рекурсії та перевантаження функцій під час створення алгоритмів, використав розгалужені алгоритми при написанні коду для задач.