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 № 4**

На тему: «Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та робота з масивами та структурами.

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

до:

Практичних Робіт до блоку № 4

**Виконав:**

Студент групи ШІ-13

Матрунич Олександр Іванович

**Тема роботи:**

Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та робота з масивами та структурами.

**Мета роботи:** вивчення пам’яті в C++, роботи з масивами, вказівниками, посиланнями та динамічним виділенням пам’яті, а також ознайомлення зі структурами даних і перевантаженням операторів для ефективного управління пам’яттю та обробки даних.

**Теоретичні відомості:**

**1. Класи пам’яті у C++**

○ Статична пам’ять.

○ Динамічна пам’ять.

○ Поняття стеку.

○ Виділення та вивільнення пам’яті.

**2. Вступ до Масивів і Вказівників:**

○ Основи масивів: визначення, важливість, приклади використання.

○ Різниця між статичними та динамічними масивами.

○ Основи вказівників: що це таке, як вони працюють.

○ Взаємозв’язок між масивами та вказівниками.

○ Вступ до посилань: основні концепції та відмінності від вказівників.

**3. Одновимірні Масиви:**

○ Створення та ініціалізація одновимірних масивів.

○ Основні операції: індексація, присвоєння, читання.

○ Цикли та обхід масивів.

○ Використання функцій для роботи з масивами.

○ Приклади алгоритмів сортування та пошуку.

**4. Вказівники та Посилання:**

○ Використання вказівників для доступу до елементів масиву.

○ Арифметика вказівників.

○ Різниця між вказівниками та посиланнями в контексті функцій.

○ Динамічне виділення пам’яті з використанням вказівників.

○ Використання вказівників для створення складних структур даних.

**5. Двовимірні Масиви:**

○ Оголошення та ініціалізація двовимірних масивів.

○ Вкладені цикли для обходу двовимірних масивів.

○ Практичні приклади використання двовимірних масивів.

○ Передача двовимірних масивів у функції.

○ Застосування двовимірних масивів для розв’язання задач.

**6. Динамічні Масиви:**

○ Основи динамічного виділення пам’яті.

○ Створення та управління динамічними масивами.

○ Використання операторів new та delete для управління пам’яттю.

○ Реалізація змінної розмірності масивів.

○ Передача динамічних масивів у функції.

**7. Структури Даних:**

○ Оголошення та використання структур.

○ Використання масивів та вказівників у структурах.

○ Функції для обробки даних у структурах.

○ Використання структур для представлення складних даних.

○ Вкладені структури та їх використання.

○ Об’єднання (Union)

○ Переліки (enumerations)

**8. Вкладені Структури:**

○ Поняття вкладених структур та їх оголошення.

○ Взаємодія з вкладеними структурами.

○ Використання вкладених структур для моделювання складних даних.

○ Передача вкладених структур у функції.

○ Приклади реального використання вкладених структур.

**9. Використання структур**

- Перевантаження операторів у структурі.

- Вивід/ввід структури (operator&lt;&lt;);

- Арифметичні операції з структурами (operator+, operator-);

- Практичні задачі на виведення структур та операції з ними

**10. Алгоритми обробки та робота з масивами та структурами:**

○ Алгоритми пошуку та сортування в масивах.

○ Обробка та маніпуляції з даними у структурах.

○ Використання циклів та умовних операторів для роботи з масивами та

структурами.

○ Інтеграція масивів та структур у алгоритми.

○ Розв’язання практичних задач з використанням масивів та структур.

**Індивідуальний план опрацювання теорії:**

* Класи пам'яті у C++
* Вступ до Масивів і Вказівників
* Одновимірні Масиви
* Вказівники та Посилання
* Двовимірні Масиви
* Динамічні Масиви
* Структури Даних

**Джерела**:

* Лекції О. Пшеничного
* Практичні М. Фаріон
* Chat gpt
* Список відтворення на YouTube ( <https://youtube.com/playlist?list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&si=sXvmPdnGkwvJLXUi> )
* Власний досвід

**Виконання роботи:**

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

**Завдання №1 - VNS Lab 4 Task 1 (6)**

1) Сформувати одновимірний масив цілих чисел, використовуючи генератор

випадкових чисел.

2) Роздрукувати отриманий масив.

3) Знищити елемент із заданим номером.

4) Додати після першого парного елемента масиву елемент зі значенням M[ I-1

]+2.

5) Роздрукувати отриманий масив.

**Завдання №2 - VNS Lab 5 Task 1 (6)**

Елемент матриці є сідловою точкою, якщо він є найменшим у своєму рядку й

найбільшим у своєму стовпці (або навпаки: найбільшим у своєму рядку й

найменшим у своєму стовпці). Для заданої матриці визначити всі сідлові

точки.

**Завдання №3 - Algotester Lab 2 (v2)**

У вас є масив r розмiром N. Також вам дано 3 цiлих числа.

Спочатку ви маєте видалити з масиву цi 3 числа, якi вам данi. Пiсля цього перетворити цей

масив у масив сум, розмiром Nnew − 1 (розмiр нового масиву пiсля видалення елементiв), який

буде вiдображати суми сусiднiх елементiв нового масиву.

Далi необхiдно вивести масив сум на екран.

**Завдання №4 - Algotester Lab 3 (v3)**

Вам дана стрiчка s.

Ваше завдання зробити компресiю стрiчки, тобто якщо якась буква йде бiльше одного разу

пiдряд у стрiчцi замiнити її на букву + кiлькiсть входжень пiдряд.

**Завдання №5 - Class Practice Work – Перевірка чи слово або число є паліндромом**

**Задача**

Реалізувати програму, яка перевіряє, чи дане слово чи число є паліндромом за допомогою рекурсії.

## Мета задачі

Навчитися користуватися механізмами перевантаження функції та використовувати рекурсію для вирішення задач обчислення.

### Вимоги:

1. Визначення функції:

Реалізуйте рекурсивну функцію *isPalindrome*, яка перевіряє, чи заданий рядок є паліндромом.

1. Приклад визначення функції:

*bool isPalindrome(const string& str, int start, int end);*

1. Перевантаження функцій:

Перевантажте функцію *isPalindrome* для роботи з цілими значеннями.

*bool isPalindrome*(ціле число);

1. Рекурсія:

Рекурсивна функція для рядків перевірить символи в поточній початковій і кінцевій позиціях. Якщо вони збігаються, він буде рекурсивно перевіряти наступні позиції, поки початок не перевищить кінець, після чого рядок буде визначено як паліндром.

### Кроки реалізації

* Визначте та реалізуйте рекурсивну функцію isPalindrome для рядків.
* Визначте та реалізуйте перевантажену функцію isPalindrome для цілих чисел. Використати математичний підхід щоб перевірити чи число є паліндромом.

**Завдання №6 - Self Practice - Task Lab 4v2:**

Вам дано масив aa з NN цілих чисел.  
Спочатку видаліть масиву aa усі елементи що повторюються, наприклад масив [1, 3, 3, 4] має перетворитися у [1, 3, 4].  
Після цього оберніть посортовану версію масиву aa на KK, тобто при K=3K=3 масив [1, 2, 3, 4, 5, 6, 7] перетвориться на [4, 5, 6, 7, 1, 2, 3].  
Виведіть результат.

*2) Дизайн*

Зображення, що містить текст, схема, знімок екрана, План

Автоматично згенерований опис

Зображення, що містить схема, текст, ескіз, малюнок

Автоматично згенерований опис

**Завдання №5 - Class Practice Work – Перевірка чи слово або число є паліндромом**

*3) Конфігурація середовища до виконання завдань:*

**Зустріч з командою**

Зображення, що містить Обличчя людини, текст, знімок екрана, відео

Автоматично згенерований опис

*4) Код програми з посиланням на зовнішні ресурси*

*До завдання №1код* ***vns\_lab\_4\_task\_1\_variant\_6\_oleksandr\_matrunych.cpp***

*До завдання №2 код* ***vns\_lab\_5\_task\_1\_variant\_6\_oleksandr\_matrunych.cpp***

*До завдання №3 код* ***algotester\_lab\_2\_task\_2\_oleksandr\_matrunych.cpp***

*До завдання №4 код* ***algotester\_lab\_3\_task\_3\_oleksandr\_matrunych.cpp***

*До завдання №5 код* ***practice\_work\_oleksandr\_matrunych.cpp***

*До завдання №6 код* ***practice\_work\_self\_oleksandr\_matrunych.cpp***

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

**Завдання №1 - VNS Lab 4 Task 1 (6)**

Зображення, що містить текст, Шрифт, знімок екрана, типографія

Автоматично згенерований опис

Час виконання: 100 хв

**Завдання №2 - VNS Lab 5 Task 1 (6)**

**Зображення, що містить текст, Шрифт, знімок екрана

Автоматично згенерований опис**

Час виконання: 120 хв

**Завдання №3 - Algotester Lab 2 (v2)**

**Зображення, що містить знімок екрана, Шрифт, число, текст

Автоматично згенерований опис**

Час виконання: 80 хв

**Зображення, що містить текст, Шрифт, ряд, знімок екрана

Автоматично згенерований опис**

**Завдання №4 - Algotester Lab 3 (v3)**

****

Час виконання: 100 хв

**Зображення, що містить текст, Шрифт, ряд, число

Автоматично згенерований опис**

**Завдання №5 - Class Practice Work – Перевірка чи слово або число є паліндромом**

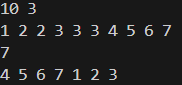
*Зображення, що містить текст, Шрифт, знімок екрана, число

Автоматично згенерований описЗображення, що містить текст, Шрифт, знімок екрана, число

Автоматично згенерований опис*

**Час виконання: 1.5 год**

**Завдання №6 - Self Practice – Lab 4v2**



Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Час виконання: 140 хв

Висновок: на лабораторній роботі вивчав пам’ять в C++, роботи з масивами, вказівниками, посиланнями та динамічним виділенням пам’яті, а також ознайомився зі структурами даних і перевантаженням операторів для ефективного управління пам’яттю та обробки даних.