

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра систем штучного інтелекту



Звіт

про виконання лабораторних та практичних робіт блоку № 4

На тему: «Програмування: алгоритм, програма, код. Системи числення.
Двійкова система числення. Розробка та середовище розробки програми.»

з дисципліни: «Основи програмування»

до:

Практичних Робіт до блоку № 4

Виконав:

Студент групи ШІ-13
Федів Андрій Сергійович

Львів 2024

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

Мета: зрозуміти основи роботи з одновимірними та двовимірними масивами, а також освоїти концепції вказівників, динамічних масивів і структур даних, що дозволить ефективно реалізовувати алгоритми обробки даних та структури, щоб спростити вирішення складних завдань.

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.

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:

- Chat gpt
- Список відтворення на YouTube (
<https://youtube.com/playlist?list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&si=sXvmPdnGkwvJLXUi>)
- Власний досвід

Виконання роботи:

VNS Lab 4 Task 1 (15)

- 1) Реалізувати з використанням масиву двонаправлене кільце (перегляд можливий в обидва боки, від останнього елемента можна перейти до першого).
- 2) Роздрукувати отриманий масив, починаючи з K -ого елемента і до $K-1$ (по кільцю вліво).
- 3) Знищити з кільця перший й останній елементи.
- 4) Роздрукувати отриманий масив, починаючи з K -ого елемента (і до $K+1$ по кільцю вправо).

VNS Lab 5 Task 1 (15)

Задано двовимірний масив. Знайти суму елементів першого стовпця без одного останнього елемента, суму елементів другого стовпця без двох останніх, суму елементів третього стовпця без трьох останніх і т.д. Останній стовпець не обробляється. Серед знайдених сум знайти максимальну

Algotester Lab 2 (v1)

У вас є дорога, яка виглядає як N чисел.

Після того як ви по ній пройдете - вашу втому можна визначити як різницю максимального та мінімального елемента.

Ви хочете мінімізувати втому, але все що ви можете зробити - викинути одне число з дороги, тобто забрати його з масиву.

В результаті цієї дії, яку мінімальну втому ви можете отримати в кінці дороги?

Algotester Lab 3 (v2)

Вам дано 2 масиви розміром N та M . Значення у цих масивах унікальні.

Ваше завдання вивести у першому рядку кількість елементів, які наявні в обох масивах одночасно, у другому кількість унікальних елементів в обох масивах разом.

Class Practice Task:

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

Паліндром — це послідовність символів, яка однаково читається вперед і назад (наприклад, «radar», «level», «12321»).

Self Practice Task:

Вам дано N слів та число K .

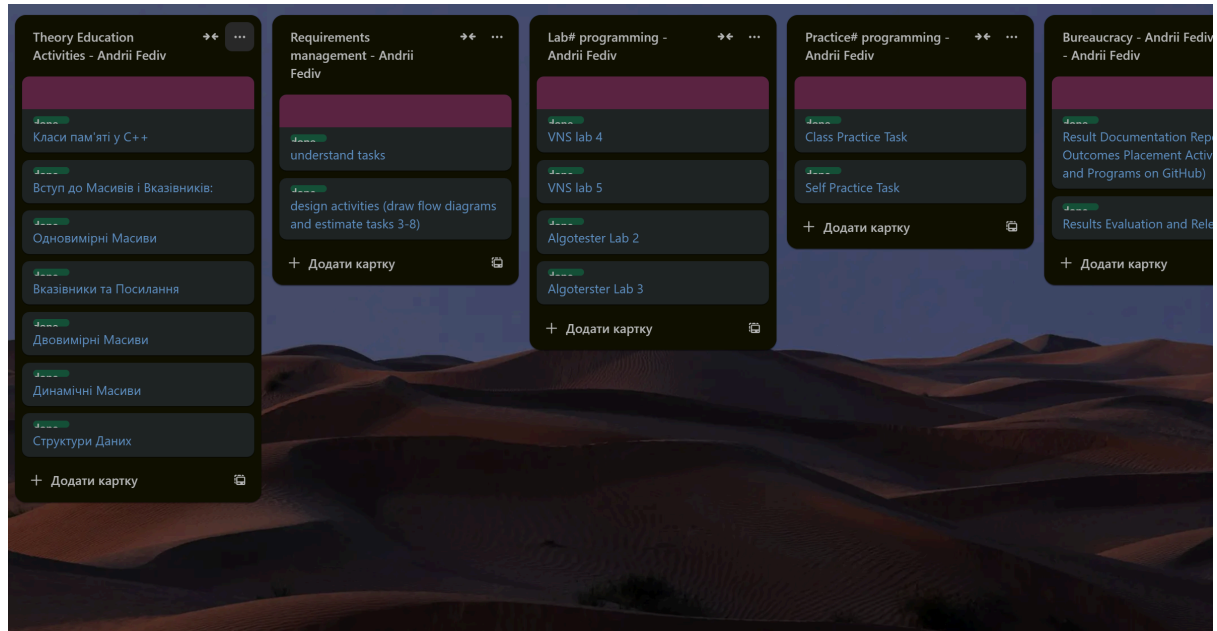
Ваше завдання перелічити букви в словах, які зустрічаються в тексті більше-рівне ніж K разів (саме слово, не буква!).

Великі та маленькі букви вважаються однаковими, виводити необхідно малі, посортовані від останньої до першої у алфавіті. Букву потрібно виводити лише один раз.

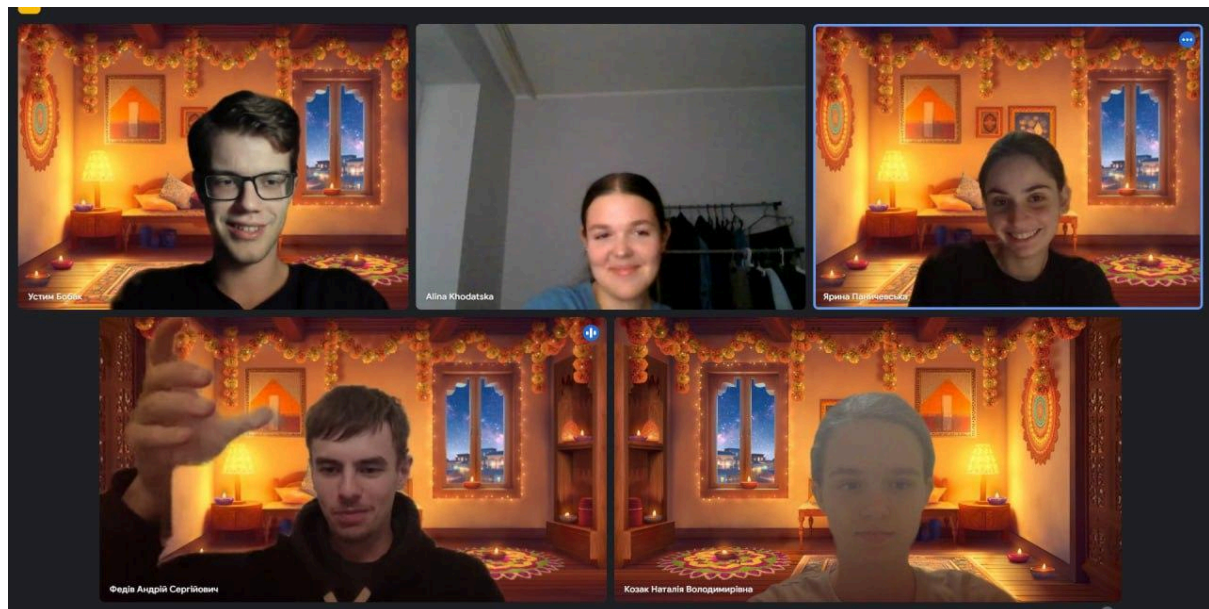
У випадку якщо таких букв немає - вивести "Empty!".

1. Requirements management and design activities

Team Trello dashboard for task control

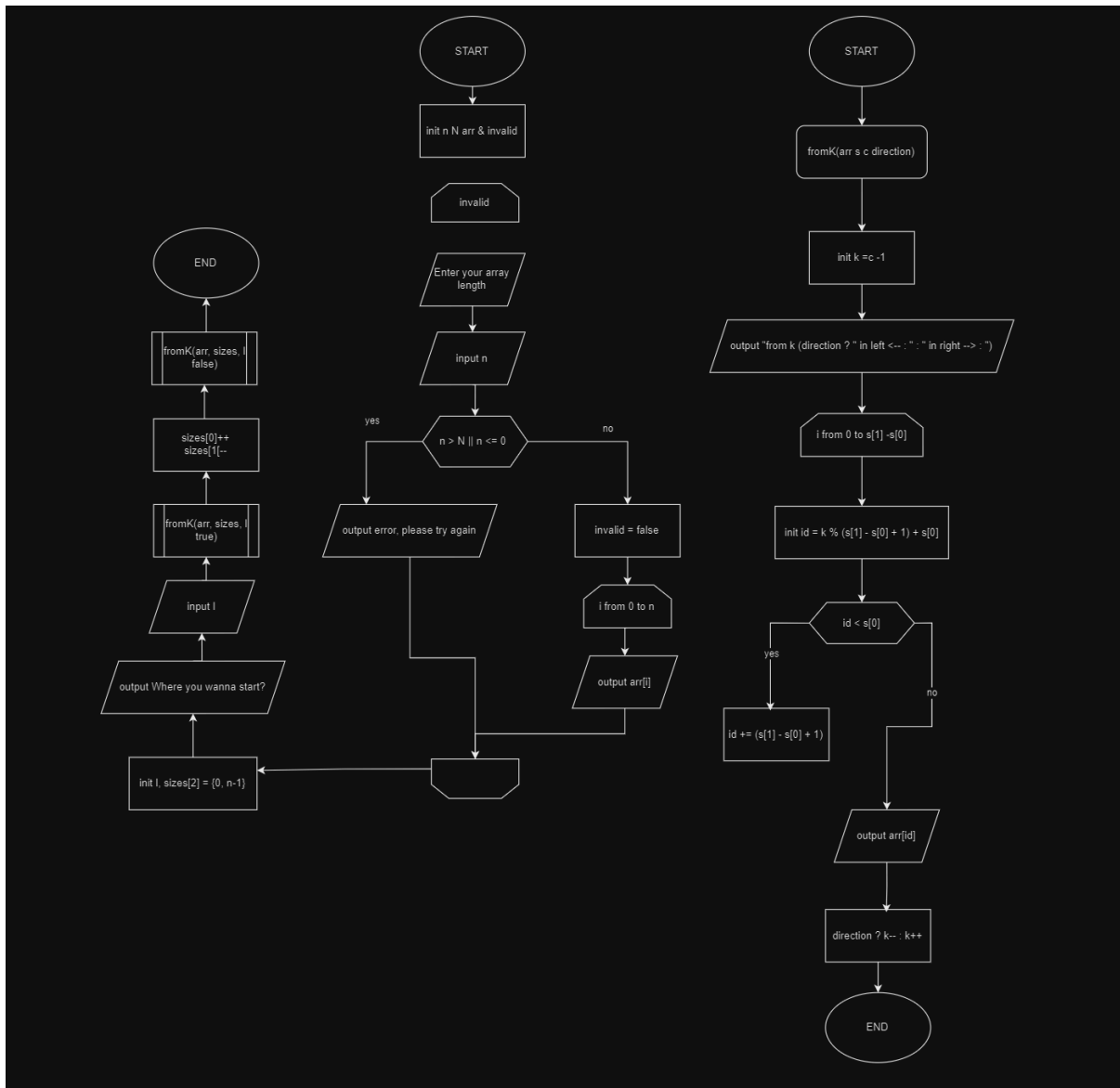


Team meeting in zoom



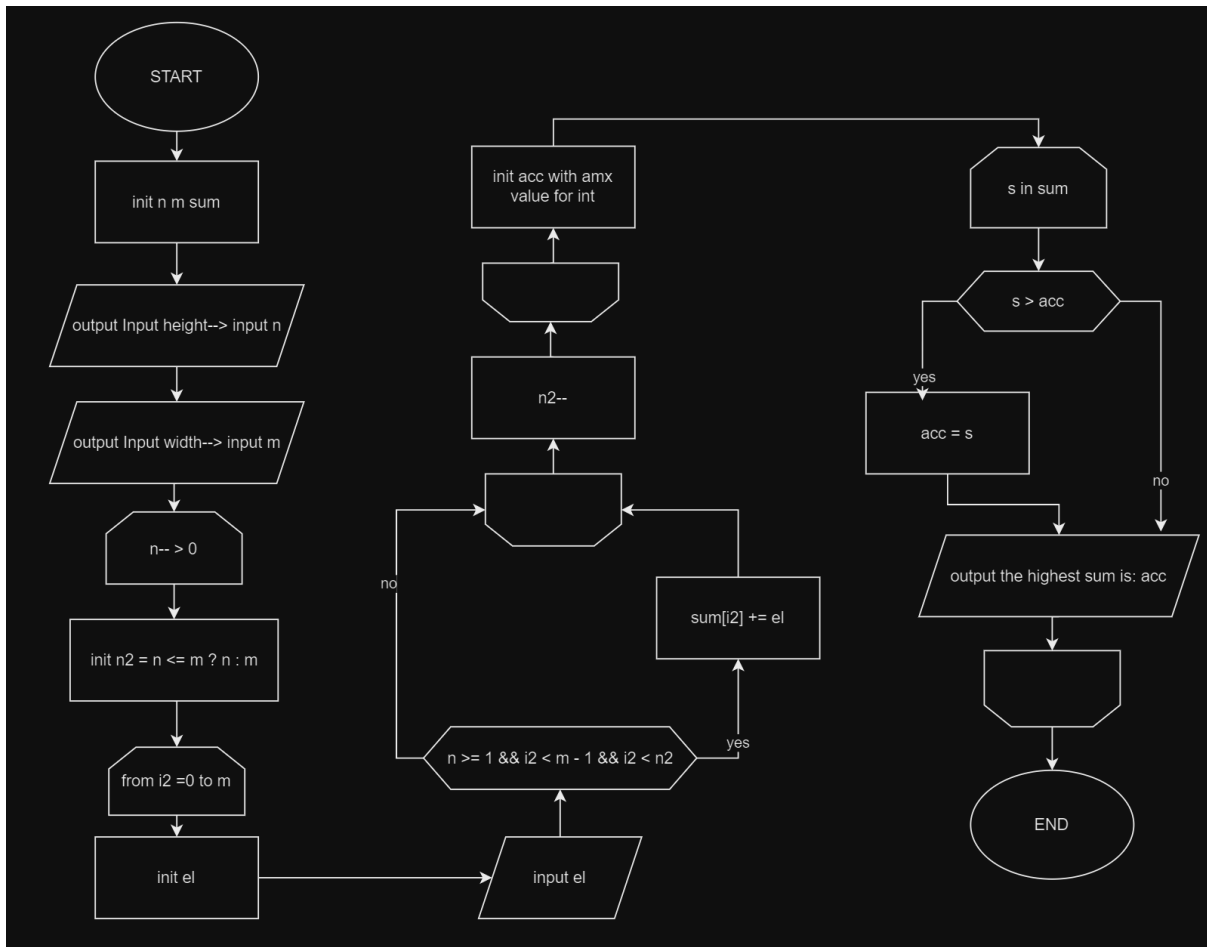
UML-diagram block-scheme for each task

VNS Lab 4 Task 1



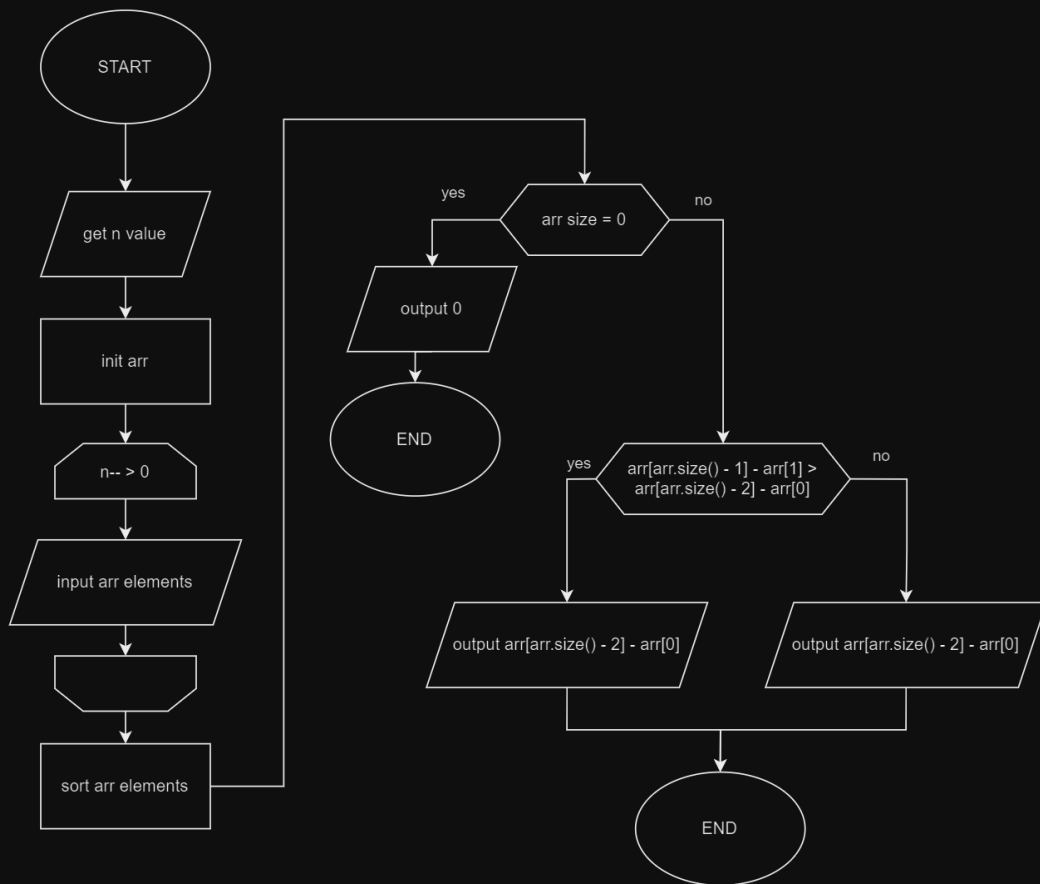
~40x6

VNS Lab 5 Task 1

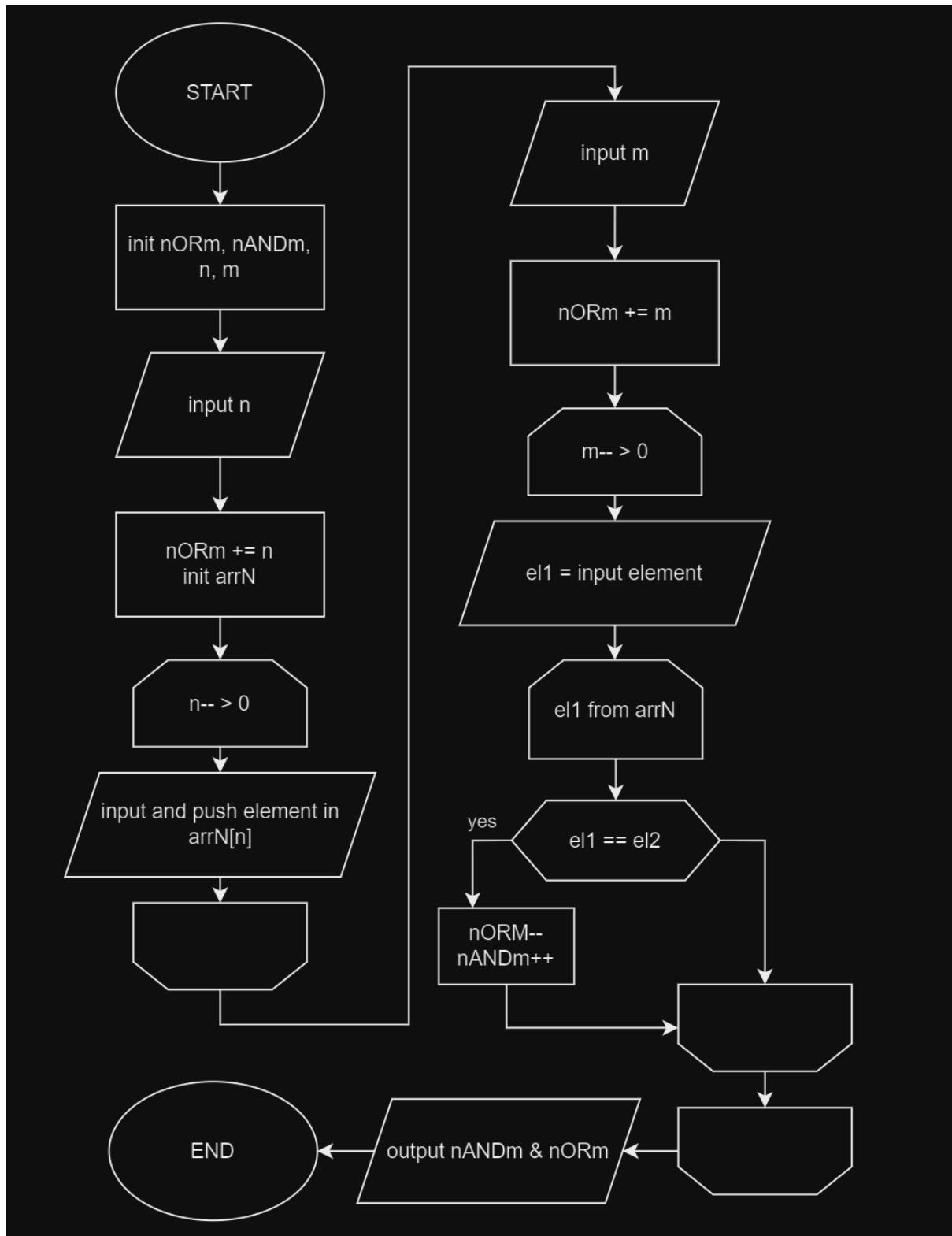


~20xB

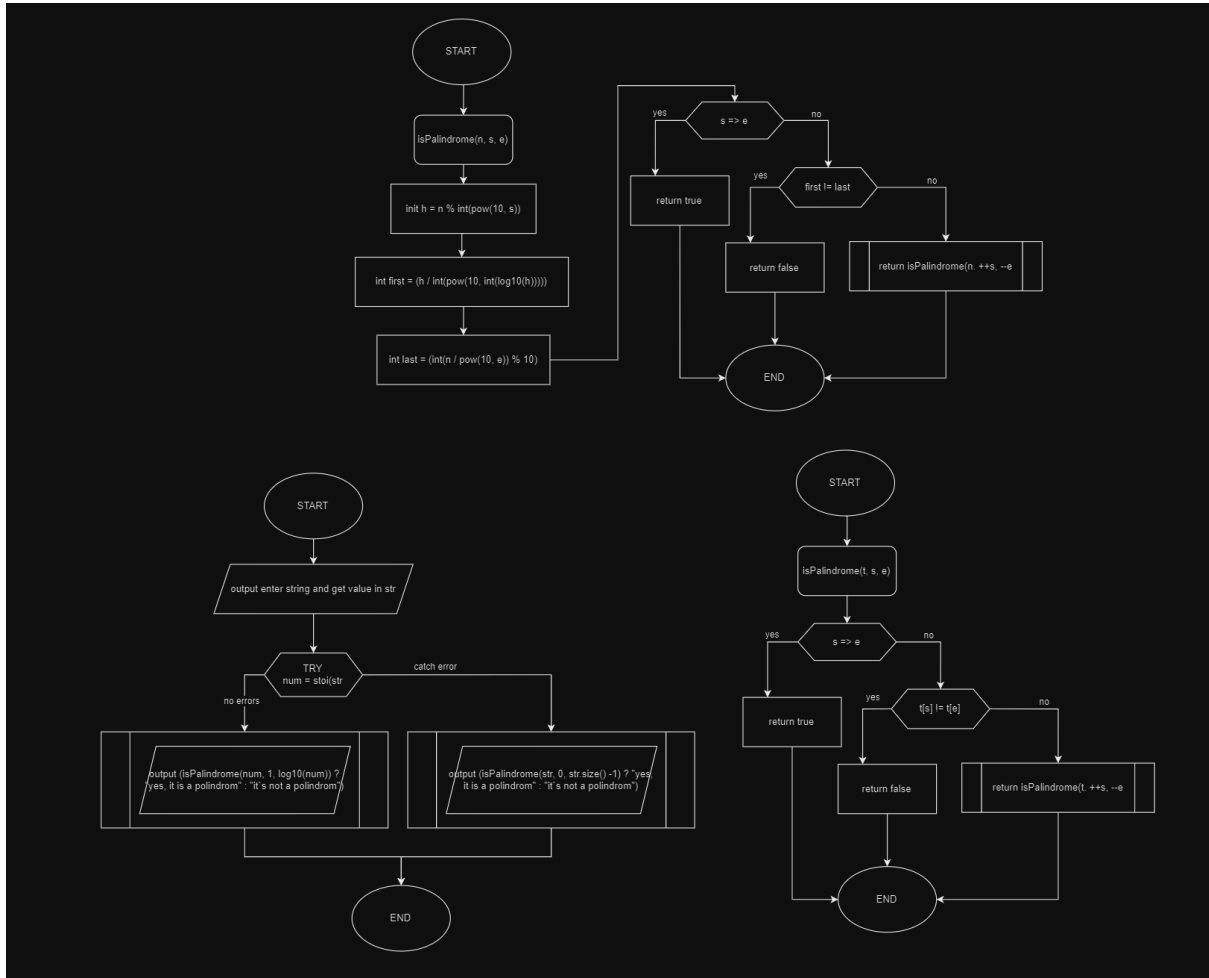
Algotester Lab 2 v1



ALgotester lab 3 v2

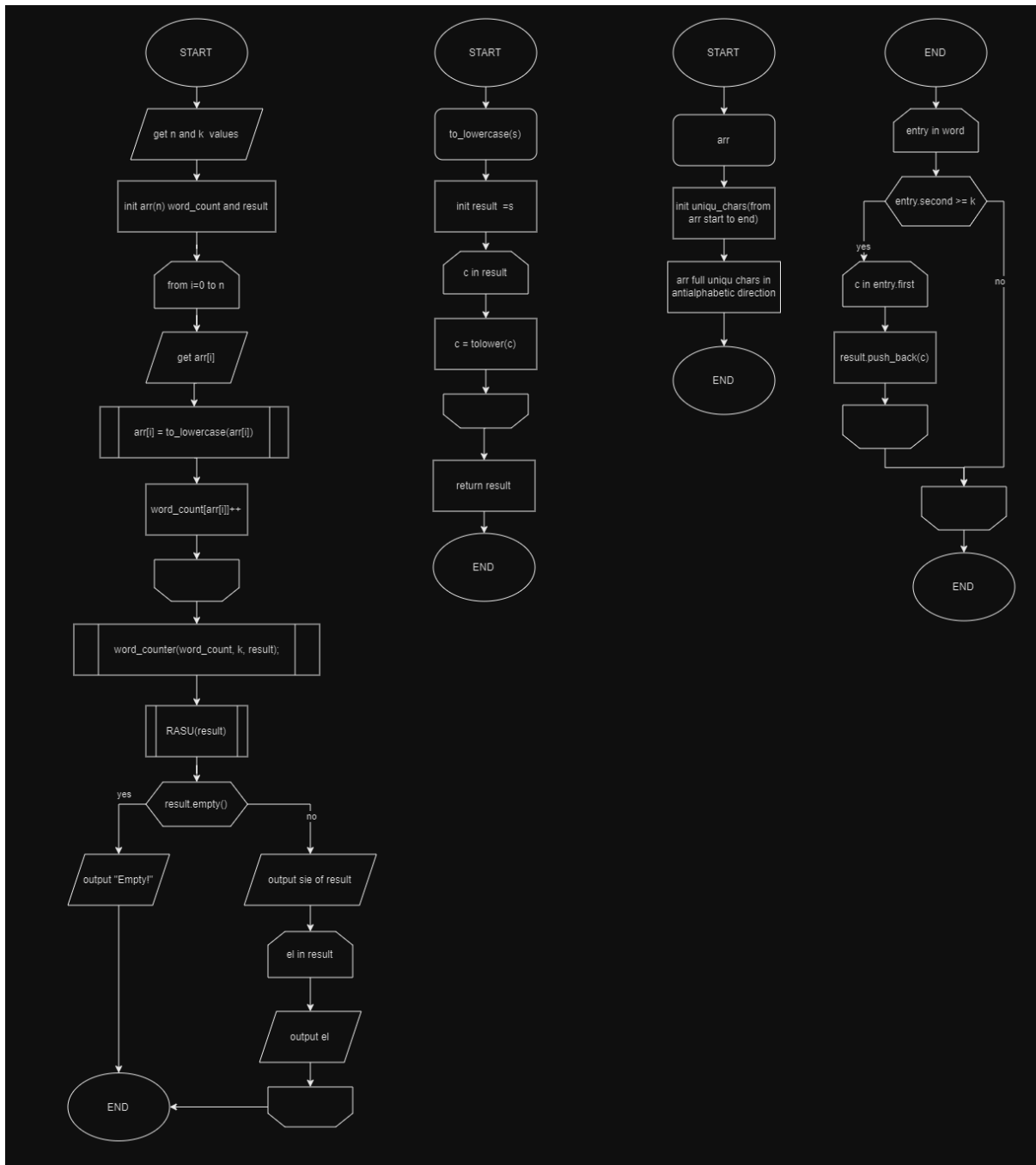


Class Practice Task



~20xB

Self Practice Task



Код програми з посиланням на зовнішні ресурси

VNS Lab 2 Task 1: `./src/vns_lab_4_task_andrii_fediv.cpp`

VNS Lab 3 Task 1: `./src/vns_lab_5_task_andrii_fediv.cpp`

Algotester Lab 2v1: `./src/algotester_lab_2_task_andrii_fediv.cpp`

Algotester Lab 3v2: `./src/algotester_lab_3_task_andrii_fediv.cpp`

Class Practice Task: `./src/practice_work_team_tasks_andrii_fediv.cpp`

Self Practice Task:

`./src/practice_work_self_algotester_tasks_andrii_fediv.cpp`

Результати виконаних завдань, тестування та фактично затрачений час

VNS Lab 4 Task 1

Затратність ~1год

VNS Lab 5 Task 1

```

(y or n)
inal]
ed]
OP\ai_pro
ndrii_fed
rii_fedi
x0000000
v.kta' '--stdout=Microsoft-MIEngine-Out-jbgtemfd
gdb.exe' '--interpreter=mi'
Input height-->4
Input width-->3
1 2 3
4 5 6
7 34 5
22 66 77
the highest sum is: 12
PS C:\WORK_FILES\cpp>

```

Затратність ~40хв

Algotester Lab 2v1

```

w.mnb' '--stdout=Microsoft-MIEngine-Out-rf231ec
gdb.exe' '--interpreter=mi'
4
3 45 6 7
4
PS C:\WORK_FILES\cpp>

```

7 днів тому	C++ 23	Зараховано	0.003	1.293	Перегляд
7 днів тому	C++ 23	Зараховано	0.003	1.316	Перегляд
7 днів тому	C++ 23	Неправильна відповідь 5	0.002	0.953	Перегляд
7 днів тому	C++ 23	Зараховано	0.003	1.234	Перегляд
7 днів тому	C++ 23	Неправильна відповідь 5	0.002	0.953	Перегляд
7 днів тому	C++ 23	Неправильна відповідь 5	0.002	0.949	Перегляд
7 днів тому	C++ 23	Неправильна відповідь 4	0.002	1.176	Перегляд

Затратність ~10хв

Algotester Lab 3v2

```
primiti
1.ibs' '--stdout=Microsoft-MIEngine-Out-3
gdb.exe' '--interpreter=mi'
4
code 0] 23 4 76 1
code 0] 3
code 0] 23 23 2
code 0] 2
normal 5
PS C:\WORK_FILES\cpp>
```

4 дні тому	C++ 23	Зараховано	0.002	1.422	Перегляд
4 дні тому	C++ 23	Неправильна відповідь 1	0.002	0.914	Перегляд
4 дні тому	C++ 23	Неправильна відповідь 1	0.002	0.922	Перегляд

Затратність ~15хв

Class Practice Task

```
cryptprimiti
1.5ul' '--stdout=Microsoft-MIEngine-Out-uok1pvv4.1g1' '--std
gdb.exe' '--interpreter=mi'
enter string: думуд
yes, it is a polindrom
PS C:\WORK_FILES\cpp>
PS C:\WORK_FILES\cpp> ^C
PS C:\WORK_FILES\cpp> & 'c:\Users\toper\.vscode\extensions\
u.ye4' '--stdout=Microsoft-MIEngine-Out-yvunr5ju.1wv' '--std
gdb.exe' '--interpreter=mi'
enter string: 43211234
yes, it is a polindrom
PS C:\WORK_FILES\cpp>
```

Затратність ~50хв (а на легшу версію 3хв ☺)

Self Practice Task

```
code 0]
code 0]
ed normal

OP\ai_pro
ndrii_fed
_algotest
xited wit

PS C:\WORK_FILES\cpp> & "c:\Users\toper\.vscode\extensions\ms-vscode.cmake-tools\bin\cmake.exe" --std-out=Microsoft-MIEngine-Out-tialmgev.45y'
gdb.exe' '--interpreter=mi'
5 2
stugna
neptune
grim
oplot
Grim
4
r m i g
PS C:\WORK_FILES\cpp> 
```

декілька секунд тому	C++ 23	Неправильна відповідь 1	0.002	0.914	Перегляд
годину тому	C++ 23	Зараховано	0.049	7.184	Перегляд
2 години тому	C++ 23	Зараховано	0.046	7.227	Перегляд
2 години тому	C++ 23	Зараховано	0.046	7.246	Перегляд
2 години тому	C++ 23	Ліміт часу 48	2.010	4.668	Перегляд
2 години тому	C++ 23	Ліміт часу 48	2.010	4.629	Перегляд
2 години тому	C++ 23	Ліміт часу 48	2.004	4.813	Перегляд
2 години тому	C++ 23	Неправильна відповідь 36	0.787	4.609	Перегляд
2 години тому	C++ 23	Неправильна відповідь 36	0.785	4.641	Перегляд
2 години тому	C++ 23	Неправильна відповідь 36	0.800	4.742	Перегляд
2 години тому	C++ 23	Ліміт часу 48	2.010	4.785	Перегляд
2 години тому	C++ 23	Неправильна відповідь 1	0.003	0.918	Перегляд
3 години тому	C++ 23	Неправильна відповідь 1	0.003	0.941	Перегляд
3 години тому	C++ 23	Неправильна відповідь 3	0.850	2.852	Перегляд
3 години тому	C++ 23	Неправильна відповідь 1	0.002	0.922	Перегляд
3 години тому	C++ 23	Неправильна відповідь 3	0.847	2.879	Перегляд

Затратність ~1.5год

Висновки:

Виконуючи цей епік я поглибив свої знання у зберіганні та обробці даних, що суттєво спрощує програмування. Застосування вказівників і динамічних масивів дозволяє зекономити пам'ять і гнучко управляти даними, а розуміння структур даних та алгоритмів обробки допомагає створювати оптимізовані рішення для різноманітних задач.