

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



Звіт

про виконання лабораторних та практичних робіт блоку № 3

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 № 3

Виконав:

Студент групи ШІ-13
Тофан Максим Васильович

Львів 2024

Тема: Цикли. Вкладені Цикли. Завершення виконання циклів. Функції. Простір імен. Перевантаження функцій. Функції з змінною кількістю параметрів (еліпсис). Рекурсія. Вбудовані функції

Мета: ознайомлення з основами роботи з циклами та вкладеними циклами, а також способами завершення їх виконання. Розглянути функції, простір імен, перевантаження функцій, функції з еліпсисом, рекурсія та вбудовані функції для поглиблення розуміння структури і поведінки програм.

Теоретичні відомості:

1. Введення в Цикли та їх Види в C++:

- Значення та роль циклів у програмуванні.
- Огляд видів циклів: for, while, do-while.
- Синтаксис та основи використання кожного типу циклу.
- Приклади базових циклів для різних задач.

2. Управління Виконанням Циклів:

- Застосування операторів break та continue.
- Умови завершення циклів.
- Передчасне завершення виконання циклу.
- Приклади та вправи з управлінням циклами.

3. Вкладені Цикли:

- Поняття та важливість вкладених циклів.
- Реалізація вкладених циклів: приклади для різних сценаріїв.
- Практичні завдання на вкладені цикли.

4. Основи Функцій у C++:

- Визначення та оголошення функцій.
- Параметри функцій: передача за значенням і за посиланням.
- Параметри за замовчуванням.
- Повернення значень з функцій.
- Приклади створення та використання функцій.

5. Перевантаження Функцій та Простір Імен:

- Концепція перевантаження функцій.
- Правила та приклади перевантаження функцій.
- Поняття та використання просторів імен.
- Вкладені простори імен (C++ 17)
- Роль просторів імен у організації коду.

6. Розширені Можливості Функцій:

- Функції зі змінною кількістю параметрів (еліпсис): синтаксис та приклади.
- Область видимості функції – static, extern.
- Рекурсія: основи, приклади рекурсивних функцій та їх аналіз.
- Передача масивів та об'єктів як параметрів.
- Повернення масивів та об'єктів з функцій.

7. Вбудовані Функції в C++:

- Огляд вбудованих функцій у C++.
- Приклади використання стандартних функцій у програмуванні.
- Роль вбудованих функцій у спрощенні коду.
- Практичні завдання для розуміння вбудованих функцій.

Індивідуальний план опрацювання теорії:

Введення в Цикли та їх Види в C++

Управління Виконанням Циклів

Вкладені Цикл

Основи Функцій у C++

Перевантаження Функцій та Простір Імен

Розширені Можливості Функцій

Вбудовані Функції в C++

Джерела:

- Лекції
- Chat GPT
- Власний досвід

Виконання роботи:

VNS Lab 2 Task 1 (13)

Використовуючи оператор циклу, знайти суму елементів, зазначених у конкретному варіанті. Результат надрукувати, надавши відповідний заголовок.

13) Знайти суму ряду з точністю $\varepsilon=0.0001$, загальний член якого

$$a_n = \frac{3^n n!}{(3n)!}$$

VNS Lab 3 Task 1 (13)

Для x , що змінюється від a до b з кроком $(b-a)/k$, де $(k=10)$, обчислити функцію $f(x)$, використовуючи її розклад в степеневий ряд у двох випадках:

а) для заданого n ;

б) для заданої точності ε ($\varepsilon=0.0001$).

Для порівняння знайти точне значення функції.

функція	діапазон зміни аргументу	n	сума
13 $y = \frac{1}{2} \ln x$	$0,2 \leq x \leq 1$	10	$S = \frac{x-1}{x+1} + \frac{1}{3} \left(\frac{x-1}{x+1} \right)^3 + \dots + \frac{1}{2n+1} \left(\frac{x-1}{x+1} \right)^{2n+1}$

VNS Lab 7 Task 1 & 2 (13)

1: Розв'язати зазначене у варіанті завдання, використовуючи функції зі змінною кількістю параметрів.

Написати функцію зі змінною кількістю параметрів для перекладу чисел з десяткової системи числення в трійкову. Написати викликаючу функцію main, що звертається до цієї функції не менше трьох разів з кількістю параметрів 3, 4, 7

2: Написати перевантажені функції й основну програму, що їх викликає.

а) для перетворення десяткового дробу у звичайний;

б) для перетворення звичайного дробу у десятковий.

Class Practice Task:

Ви створюєте просту програму керування бібліотекою. Книги в бібліотеці є, користувачі можуть їх взяти або повернути.

Self Practice Task:

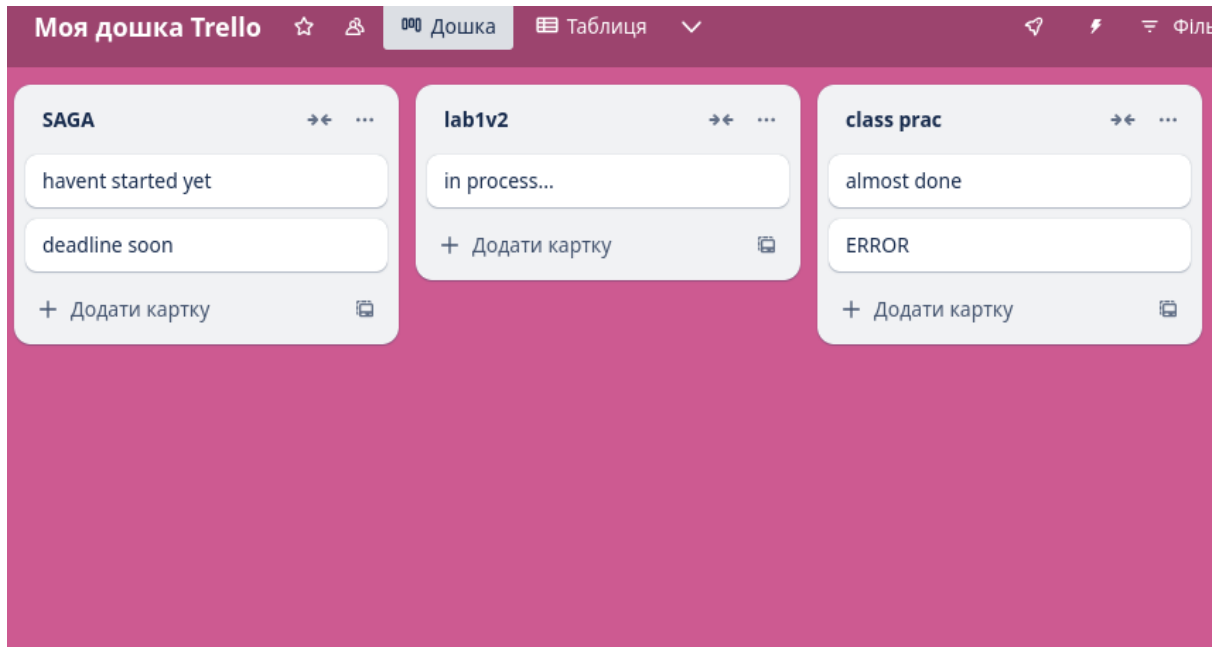
У вас є стіл, у якого є 4 ніжки, довжини яких вам дано.

Ви хочете зробити ніжки рівної довжини, для цього ви відпиляєте D від кожної ніжки (тобто вам буде дано 4 числа, кожне з яких буде означати відпилювання від відповідної ніжки стола).

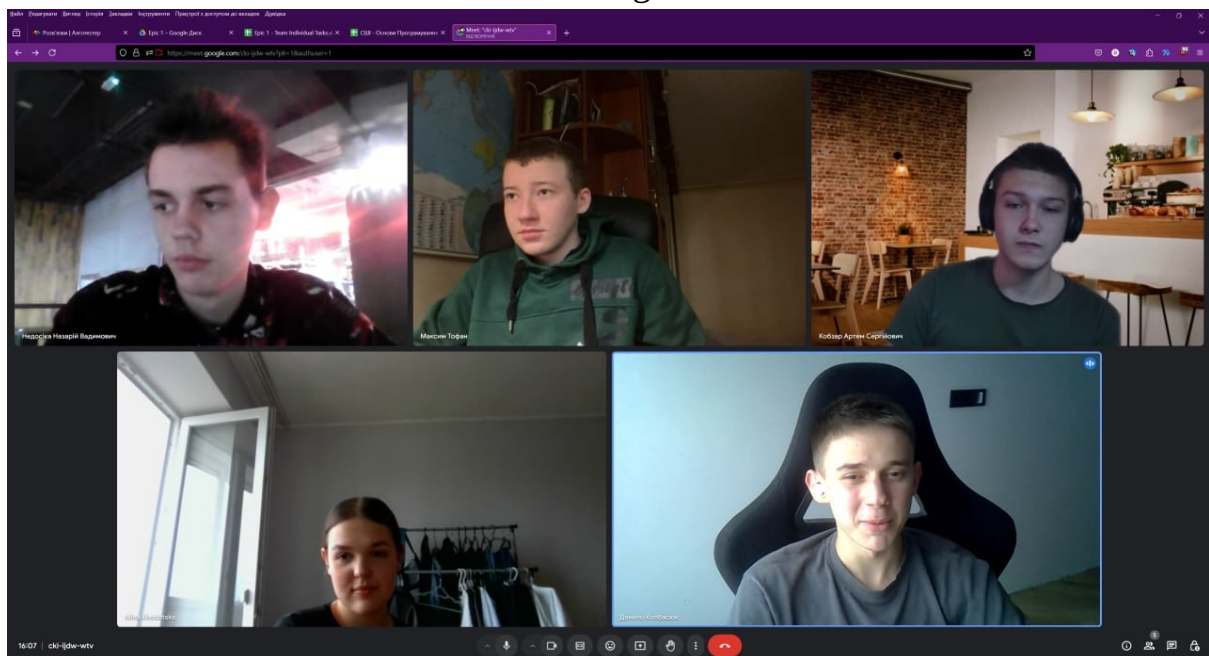
Якщо під час відпилювання найдовша ніжка стола буде у 2 рази більша-рівна ніж найменша ніжка - стіл перевернеться, але відпилювати ніжки це вам не завадить.

1. Requirements management and design activities

Team Trello dashboard for task control

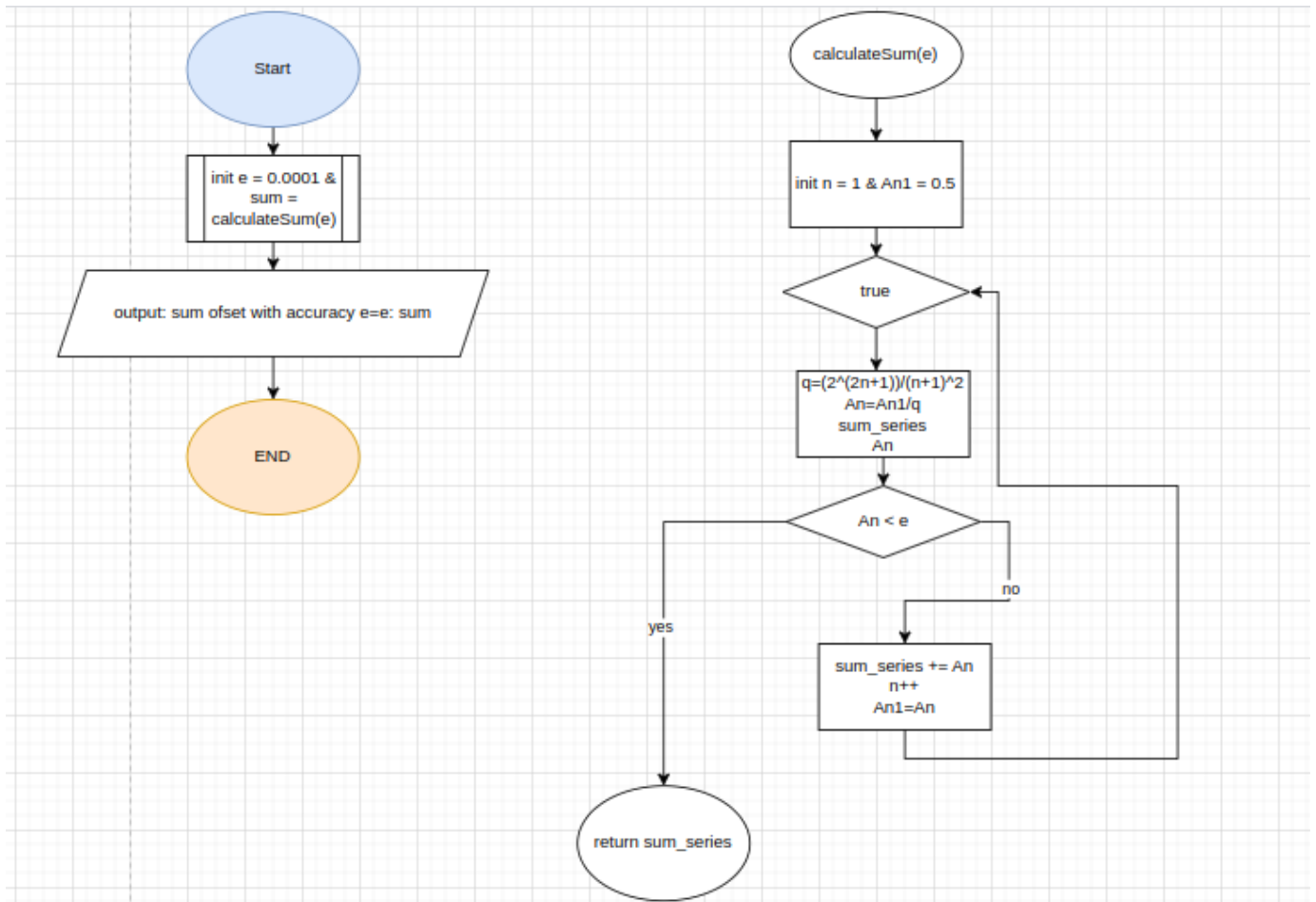


Team meeting in zoom



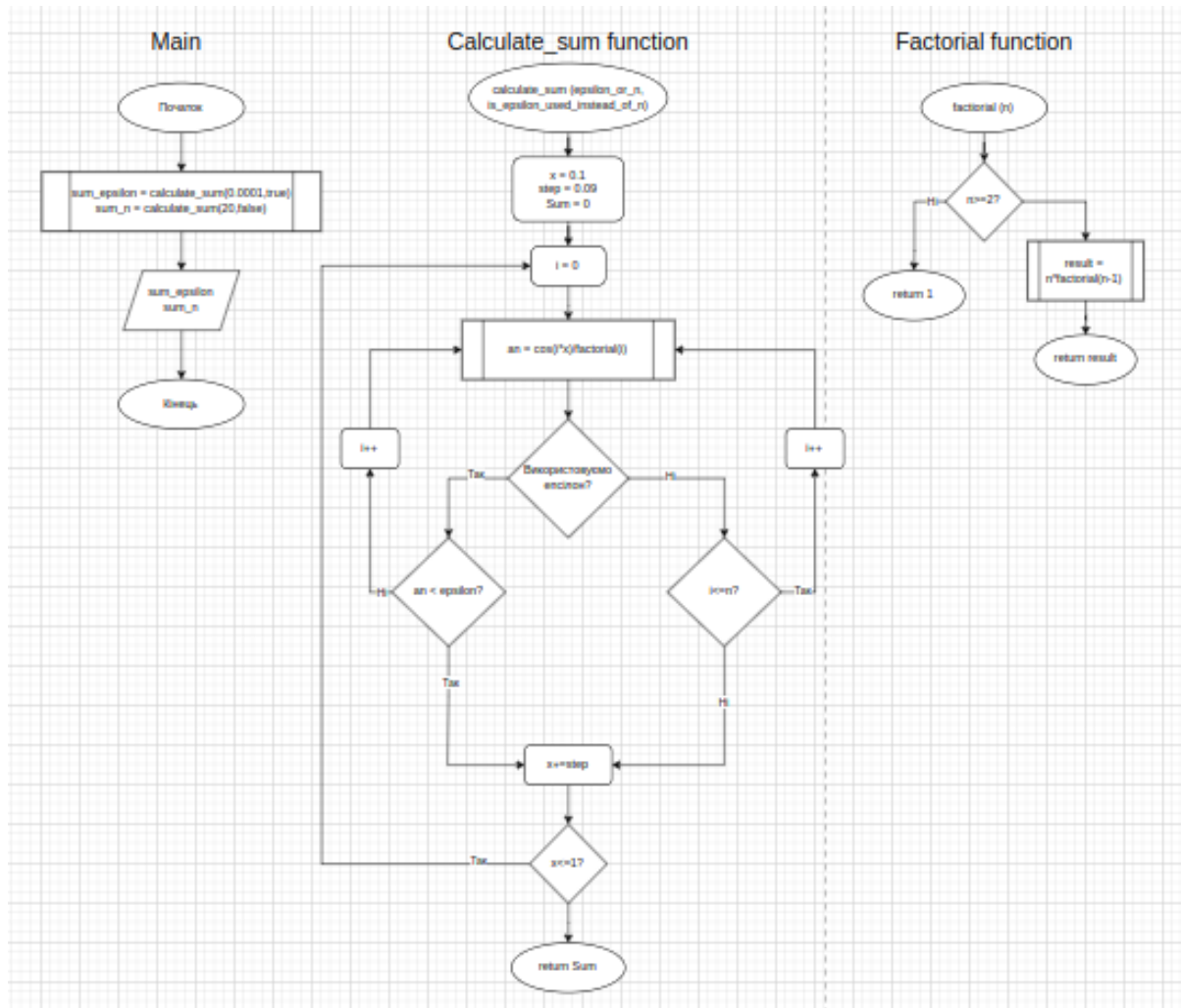
UML-diagram block-scheme for each task

VNS Lab 2 Task 1



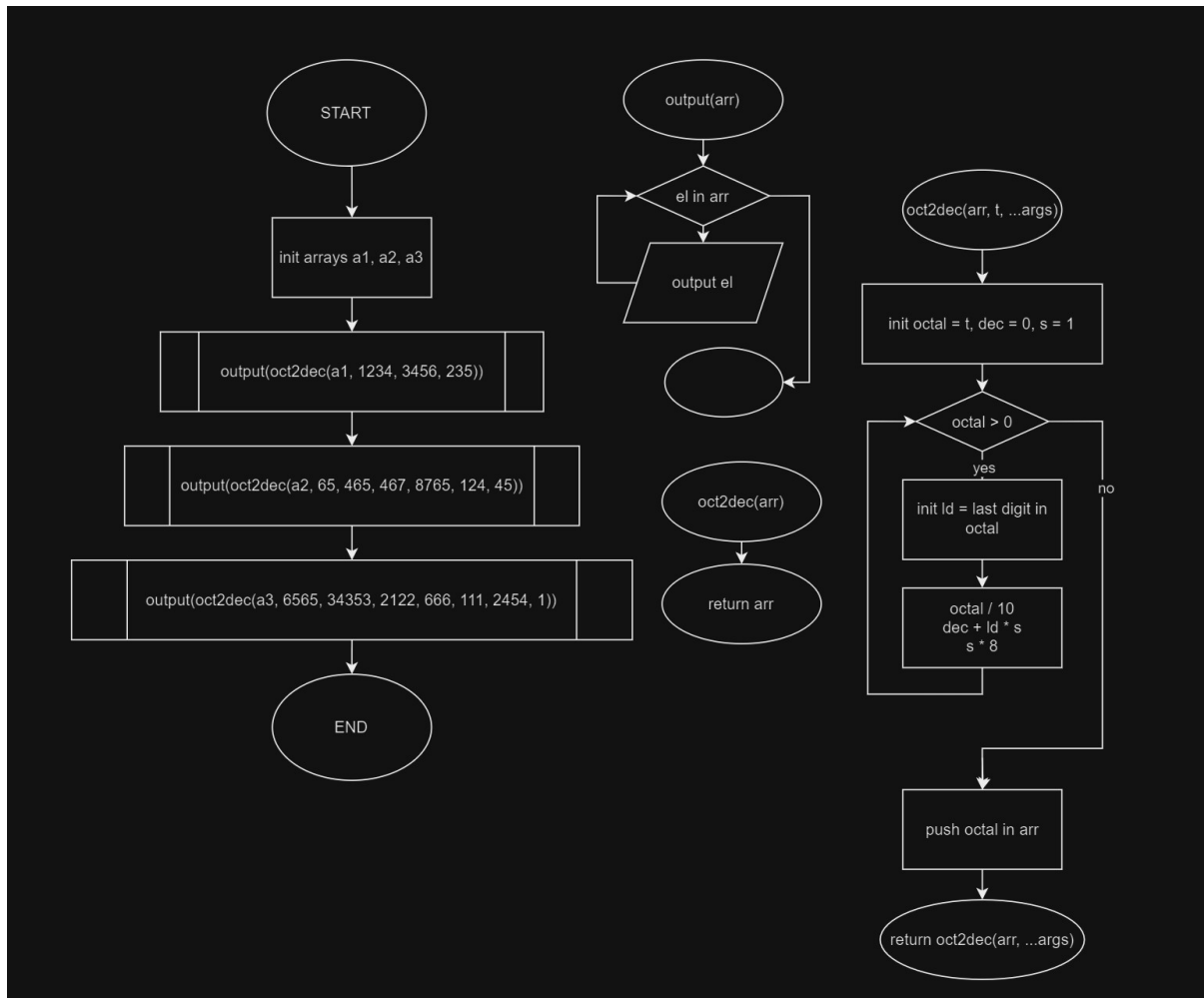
~30x6

VNS Lab 3 Task 1



~1год

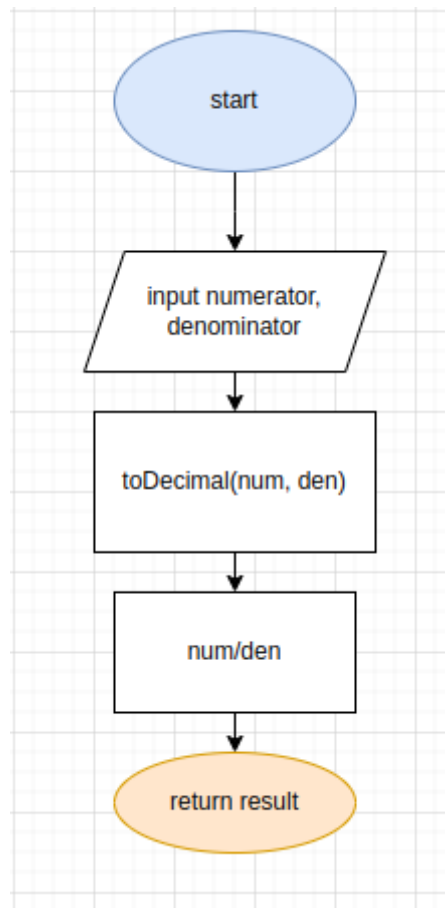
VNS Lab 7 Task 1



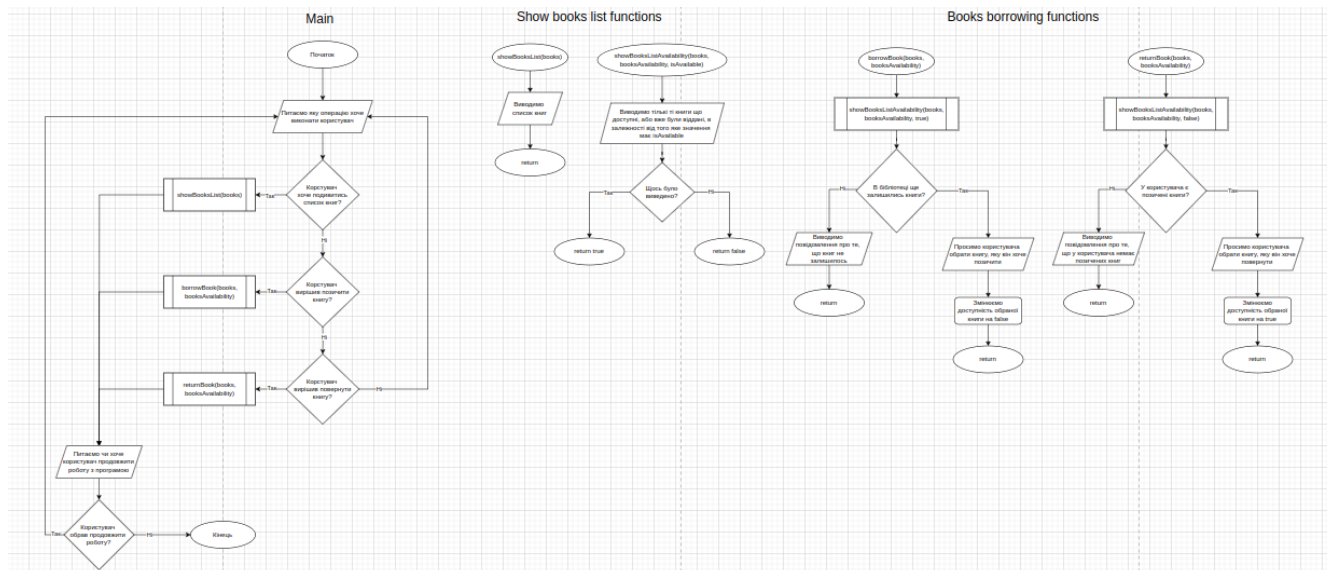
~35XB

VNS Lab 7 Task 2

~15XB



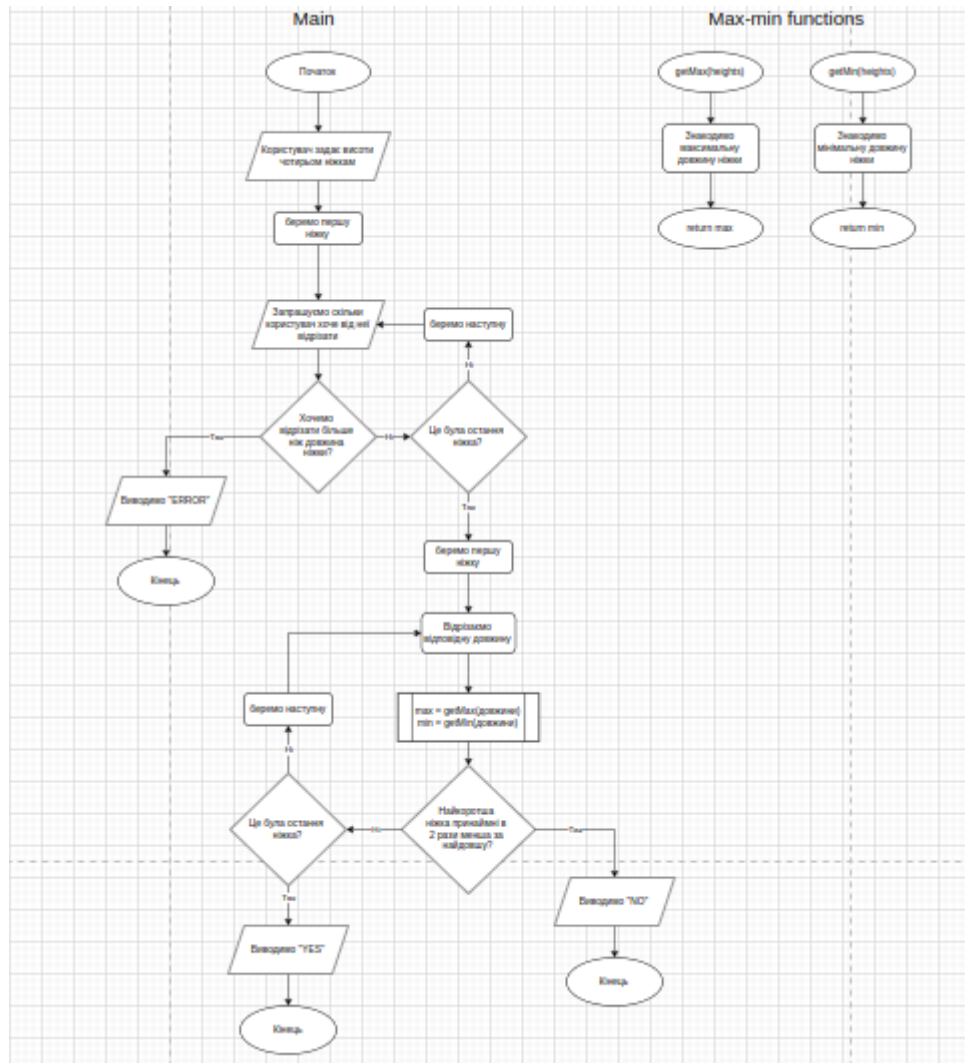
Class Practice Task



~50xb

Self Practice Task

~20xB



Результати виконаних завдань, тестування та фактично затрачений час

VNS Lab 2 Task 1

```
Сума ряду з точністю 0.0001 дорівнює: 1.52545  
[1] + Done "/usr/bin/gdb"  
g20vz3g.sbq" 1>"/tmp/Microsoft-MIEngine-Out-iid  
max@max-user:~$
```

Затратність ~45хв

VNS Lab 3 Task 1

x	Exact	Series (n=10)	Series (epsilon=0.0001)
0.2	-0.804719	-0.804703	-0.804679
0.28	-0.636483	-0.636482	-0.636466
0.36	-0.510826	-0.510826	-0.51082
0.44	-0.41049	-0.41049	-0.410487
0.52	-0.326963	-0.326963	-0.326959
0.6	-0.255413	-0.255413	-0.255412
0.68	-0.192831	-0.192831	-0.19283
0.76	-0.137218	-0.137218	-0.137218
0.84	-0.0871767	-0.0871767	-0.0871767
0.92	-0.0416908	-0.0416908	-0.0416908
1	0	0	0

```
[1] + Done "/usr/bin/gdb" --interpret  
210hvrq.gza" 1>"/tmp/Microsoft-MIEngine-Out-yhyttyyd.mpa"  
max@max-user:~/lpnu/epic 3$
```

Затратність ~1.5год

VNS Lab 7 Task 1

```
Введіть десятковий дріб: 0.75
Звичайний дріб: 3/4
[1] + Done
k3sep4f.hyu" 1>"/tmp/Microsoft-MIE
max@max-user:~/lpnu/epic 3$
```

Затратність ~20хв

VNS Lab 7 Task 2

```
Введіть чисельник і знаменник звичайного дробу: 1
4
Десятковий дріб: 0.25
[1] + Done
"/usr/bin/gdb" --inte
v4hzkq2.pui" 1>"/tmp/Microsoft-MIEngine-Out-p4mimpfe.c
max@max-user:~/lpnu/epic 3$
```

Затратність ~15хв

Class Practice Task

```
Меню:
1. Список всіх книг
2. Позичити книгу
3. Повернути книгу
4. Вийти
Ваш вибір: 2
Введіть номер книги, яку хочете позичити: 1
Ви позичили книгу.

Меню:
1. Список всіх книг
2. Позичити книгу
3. Повернути книгу
4. Вийти
Ваш вибір: 3
Введіть номер книги, яку хочете повернути: 1
Ви повернули книгу.

Меню:
1. Список всіх книг
2. Позичити книгу
3. Повернути книгу
4. Вийти
Ваш вибір: 4
Програма завершена.
[1] + Done
"/usr/bin/gdb" --
ffzxhea.ypk" 1>"/tmp/Microsoft-MIEngine-Out-mgj0hm
max@max-user:~/lpnu/epic 3$
```

Затратність ~50хв

Self Practice Task

```
10 10 10 10
2 2 2 2
YES[1] + Done
n-yt5f5xkc.0mr" 1>"/tmp/Microsof
max@max-user:~/lpnu/epic 3$
```

годину тому	Lab 1v2 - Lab 1v2	C++ 23	Зараховано	0.003	1.418	1863983
день тому	Lab 1v2 - Lab 1v2	C++ 23	Зараховано	0.003	1.289	1863133
день тому	Lab 1v2 - Lab 1v2	C++ 23	Неправильна відповідь 2	0.002	1.043	1863132
день тому	Lab 1v2 - Lab 1v2	C++ 23	Неправильна відповідь 2	0.002	0.945	1863131

Затратність ~15хв

Висновки:

Під час роботи над цим епіком я поглибив свої знання про цикли, що дозволить мені ефективніше оптимізувати виконання програм. Розуміння функцій, їх перевантаження та рекурсії відкриває можливості для створення більш гнучких і потужних рішень для різноманітних завдань. Знання вбудованих функцій і принципів роботи з простором імен сприяли уникненню конфліктів та забезпеченню чіткої організації коду.