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 № 3**

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

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

до:

ВНС Лабораторної Роботи № 2

ВНС Лабораторної Роботи № 3

ВНС Лабораторної Роботи № 7

Практичних Робіт до блоку № 3

**Виконав:**

Студент групи ШІ-11

Фарина Арсеній Петрович

Львів 2024

**Тема роботи:**

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

**Мета роботи:**

Навчитися працювати з функціями, циклами та перевантаженими функціями. Застосувати на практиці вивчений теоретичний матеріал.

**Теоретичні відомості:**

- Тема №1: Введення в Цикли та їх Види в С++.

- Тема №2: Управління Виконанням Циклів.

- Тема №3: Вкладені Цикли.

- Тема №4: Основи Функцій у С++

- Тема №5: Перевантаження Функцій та Простір Імен.

- Тема №6: Розширені Можливості Функцій.

- Тема №7: Вбудовані Функції в С++

1. Індивідуальний план опрацювання теорії:

- Тема №1: Введення в цикли та їх види в С++:

○ Джерела інформації:

▪ Статті.

<https://www.youtube.com/watch?v=zBtcqNdiRf4&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=31>

https://www.youtube.com/watch?v=ckJtOMcIxyU&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=32

- Що опрацьовано:

○      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.

Запланований час на вивчення 1 година.

Витрачений час 1 година.

  - Тема №2: Управління виконанням циклів:

○ Джерела інформації:

▪ Статті.

<https://www.youtube.com/watch?v=rj1OLsBKazA&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=36>

https://www.youtube.com/watch?v=UY295pIdeoQ&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=37

- Що опрацьовано:

○      Застосування операторів break та continue.

○      Умови завершення циклів.

○      Передчасне завершення виконання циклу.

○      Приклади та вправи з управлінням циклами.

Запланований час на вивчення 30 хвилин.

Витрачений час 30 хвилин.

- Тема №3: Вкладені цикли:

○ Джерела інформації:

▪ Статті.

https://www.youtube.com/watch?v=mBPHKQx21eE

- Що опрацьовано

○      Поняття та важливість вкладених циклів.

○      Реалізація вкладених циклів: приклади для різних сценаріїв.

○      Практичні завдання на вкладені цикли.

Запланований час на вивчення 1 година.

Витрачений час 1 година.

- Тема №4 Основи функцій у С++:

○ Джерела інформації:

▪ Статті.

https://www.youtube.com/watch?v=G8P6SvdqU9s&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=43

- Що опрацьовано

○      Визначення та оголошення функцій.

○      Параметри функцій: передача за значенням і за посиланням.

○  Параметри за замовчуванням.

○      Повернення значень з функцій.

○      Приклади створення та використання функцій.

Запланований час на вивчення 2 години.

Витрачений час 2 години.

- Тема № 5 Перевантаження функцій та простір імен:

○ Джерела інформації:

▪ Статті.

https://www.youtube.com/watch?v=hcYgFCgeZzQ

- Що опрацьовано

○      Концепція перевантаження функцій.

○      Правила та приклади перевантаження функцій.

○      Поняття та використання просторів імен.

○  Вкладені простори імен (C++ 17)

○      Роль просторів імен у організації коду.

Запланований час на вивчення 1 година.

Витрачений час 1 година..

- Тема №6: Розширені можливості функцій:

○ Джерела інформації:

▪ Статті.

<https://acode.com.ua/urok-15-funktsiyi-i-operator-return/>

https://www.youtube.com/watch?v=\_N3zkbnCTw0

- Що опрацьовано

○      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.

Запланований час на вивчення 1 година.

Витрачений час 1 година.

- Тема №7: Вбудовані функції в С++:

○ Джерела інформації:

▪ Статті. https://www.youtube.com/watch?v=V\_8XRRIus7Y&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=49

- Що опрацьовано

○      Огляд вбудованих функцій у С++.

○      Приклади використання стандартних функцій у програмуванні.

○      Роль вбудованих функцій у спрощенні коду.

○      Практичні завдання для розуміння вбудованих функцій.

Запланований час на вивчення 1 година.

Витрачений час 1 година.

**Виконання роботи:**

**1. Опрацювання завдання та вимог до програм.**

Завдання №1  
**VNS LAB 2 – TASK 1**

Знайти суму цілих додатніх парних чисел, менших 100.

Завдання №2

**VNS LAB 3 – TASK 1**

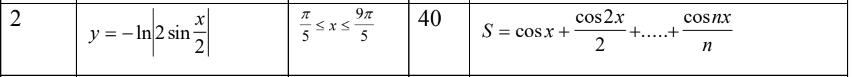
Для х, що змінюється від a до b з кроком (b-a) / k, де (k=10), обчислити

функцію f(x), використовуючи її розклад в степеневий ряд у двох випадках:

а) для заданого n;

б) для заданої точності ε (ε=0.0001).

Для порівняння знайти точне значення функції.



Завдання №3  
**VNS LAB 7 – TASK 1**Написати функцію mult зі змінною кількістю параметрів, що знаходить добуток чисел типу float. Написати викликаючу функцію main, що звертається до функції mult не менше трьох разів з кількістю параметрів 3, 7, 11.

Завдання №4  
**VNS LAB 7 – TASK 2**Написати перевантажені функції й основну програму, що їх викликає.

2.

а) для додавання дійсних чисел;

б) для додавання комплексних чисел.

Завдання №5  
**Class Practice work**

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

**Програма повинна вміти**

* Перерахувати всі книги.
* Дозволити взяти книгу (за наявності).
* Дозволити повернення книги.

**Структури даних**

* Використовуйте масив або вектор для зберігання назв книг.
* Використовуйте інший масив або вектор для збереження стану доступності кожної книги.

### Вимоги:

1. while: продовжувати працювати, доки користувач не вирішить вийти.
2. do while: Після кожної операції (позичити, повернути, перерахувати) запитуйте користувача, чи хоче він виконати іншу операцію. Якщо так, поверніться назад.
3. for: список усіх книг за допомогою циклу.
4. for each: перевірити наявність кожної книги.
5. goto: якщо користувач вводить неправильний вибір, використовуйте goto, щоб перенаправити його до головного меню.

Завдання №6

**Self practice work algotester**

Зустрілися якось працівники великих компаній і почали... Обговорювати план вулиці.

Виявляється, всі приміщення, які орендуватимуть ці компанії, збудують вздовж однієї вулиці.

ii-та компанія орендуватиме офіс довжиною lili метрів. Офіси будуватимуть один за одним, починаючи з точки 0. Всі працівники приїжджатимуть на стоянку, яку побудують в точці 0, та будуть йти до офісів своїх компаній.

Тобто, якщо офіси будуть збудовані в порядку p1,p2,...,pnp1,p2,...,pn, то перший офіс почнеться в точці 0 і закінчиться в точці lp1lp1, другий почнеться в lp1lp1 і закінчиться в lp1+lp2lp1+lp2 і т.д. Двері кожного офісу завжди є в кінці будинку, який є ближчим до стоянки.

Ваше завдання — допомогти розмістити офіси компаній на цій вулиці в такому порядку, щоб сумарна відстань від точки 0 до усіх офісів була мінімальною.

# Вхідні дані

У першому рядку задане ціле число n — кількість компаній.

У наступному рядку задано n цілих чисел li через пробіл — довжини офісів усіх компаній.

# Вихідні дані

У єдиному рядку виведіть n чисел від 1 до n — порядок компаній, в якому варто будувати офіси.

Якщо існує декілька оптимальних порядків — виведіть будь-який із них.

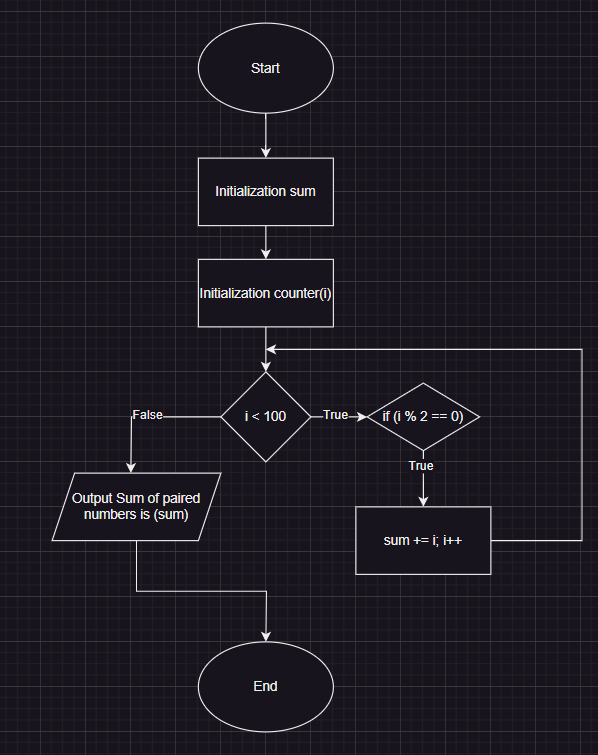
**2. Дизайн та планувальна оцінка часу виконання завдань:**

**Програма №1**

* Блок-схема
* Важливі деталі для реалізації програми

Використати цикл for для того щоб пройтися по числах від 0 до 100 і в середині циклу прописати if, щоб додавали тільки ті числа, які при діленні на два не дають остачі, тобто парні числа.

* Плановий час на реалізацію 30 хвилин.

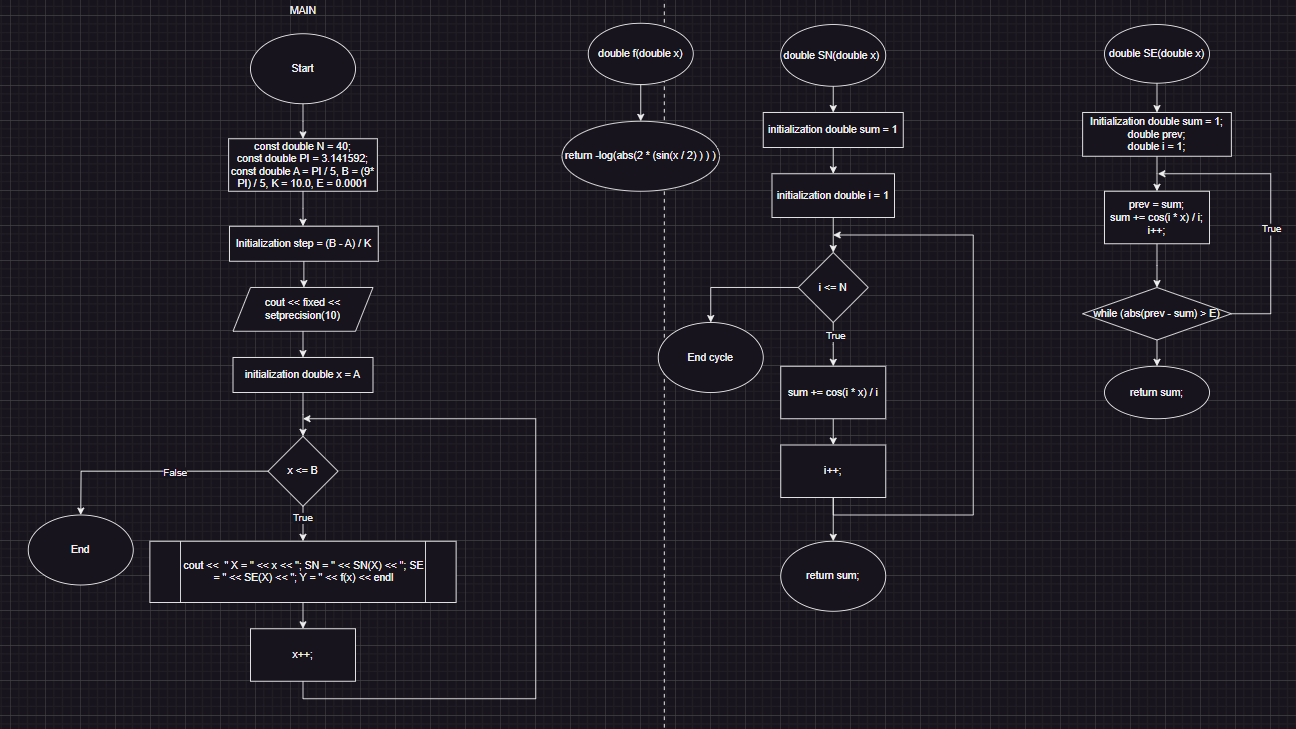
  
*Рисунок 2.1. Блок-схема до програми 1*

**Програма №2**

* Блок-схема
* Важливі деталі для реалізації програми

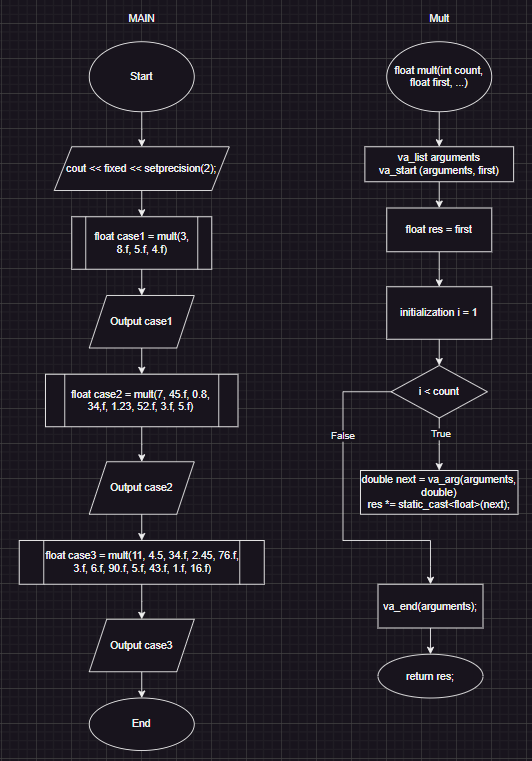
Написати функції для обрахунку значення з точністю ε (ε=0.0001), функцію для значення суми SN, і функцію SE - для заданої точності, а також виводити на екран X- значення параметра; SN- значення суми для заданого n; SE- значення суми для заданої точності; Y-точне значення функції.

* Плановий час на реалізацію 2.5 години.

  
*Рисунок 2.2. Блок-схема до програми 2*

**Програма №3**

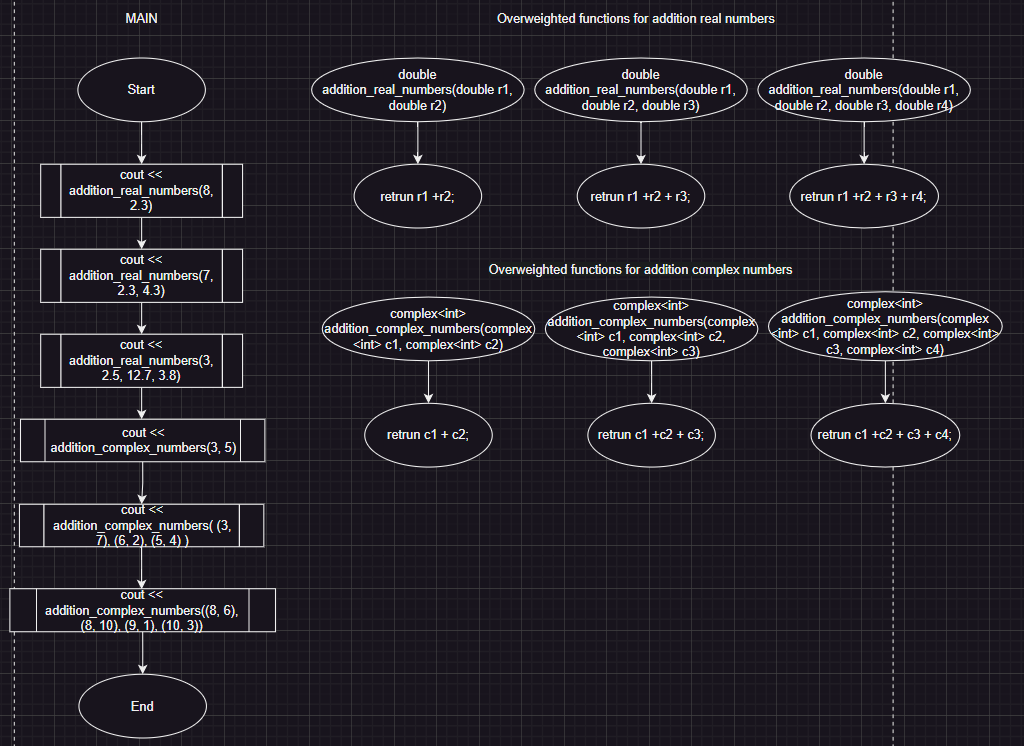
* Блок-схема
* Важливі деталі для реалізації програми
* Використати перевантажену функцію mult, і навчитися працювати з бібліотекою cstdarg.
* Плановий час на реалізацію 1.5 години.



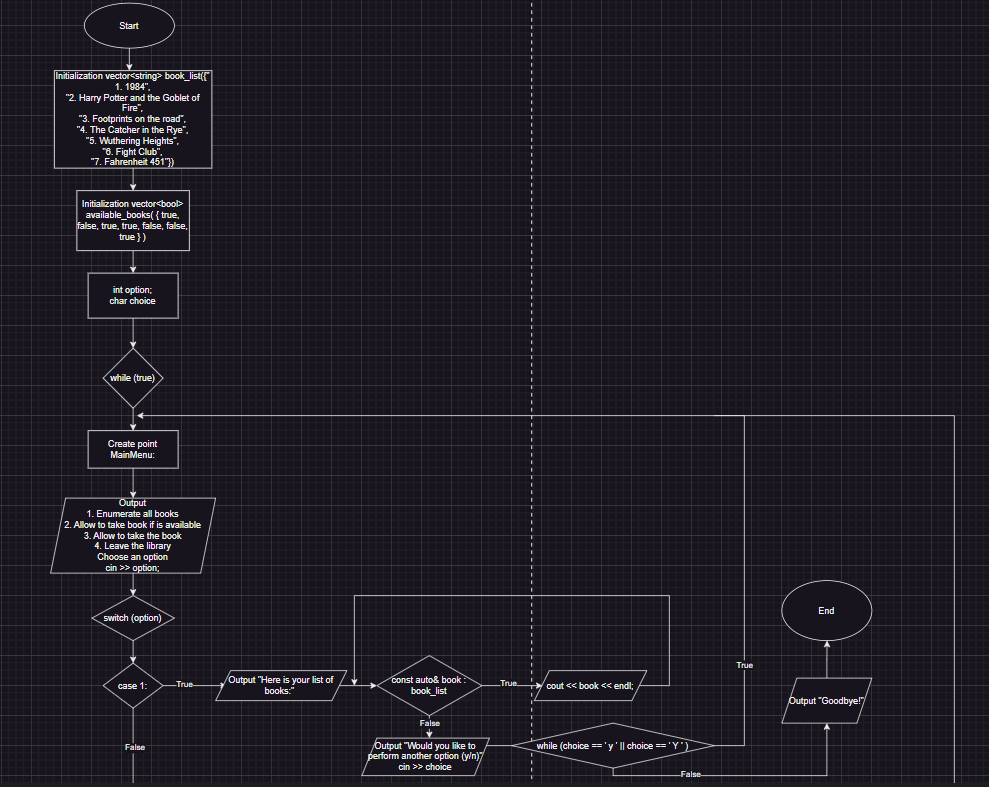
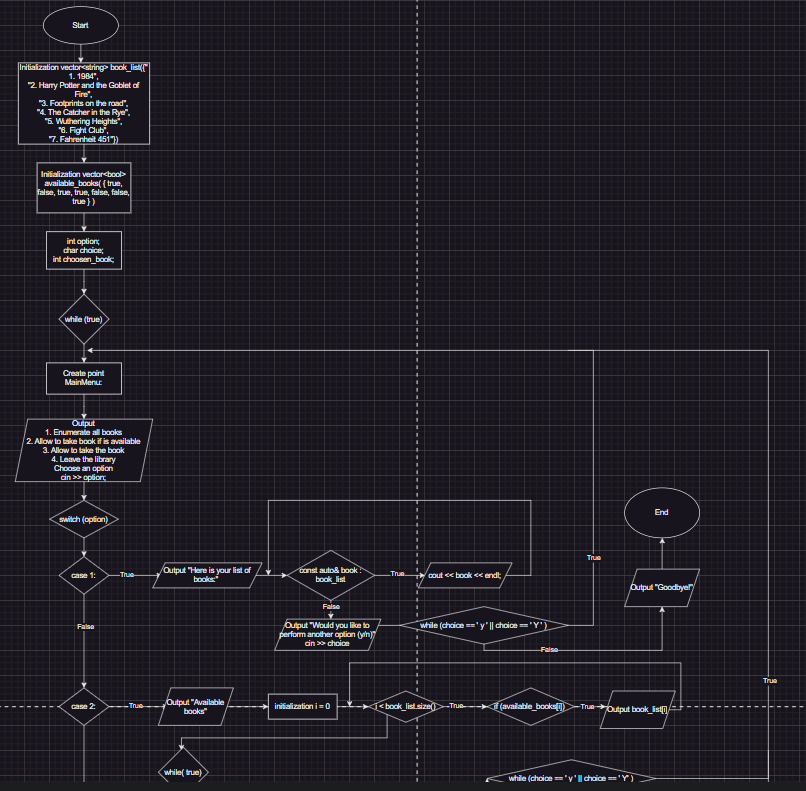
*Рисунок 2.3. Блок-схема до програми 3*

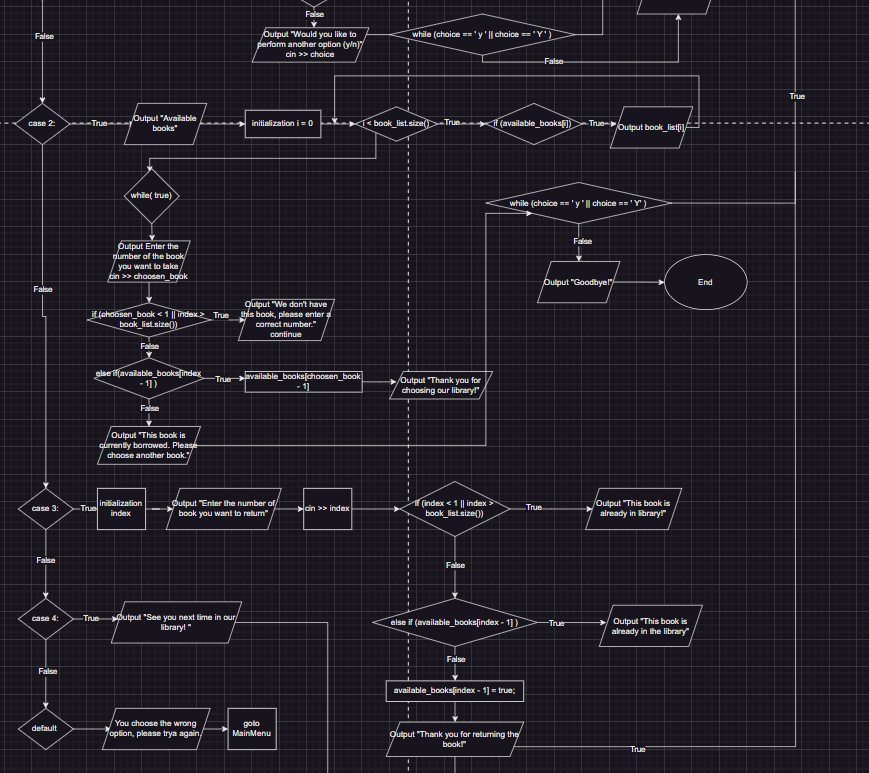
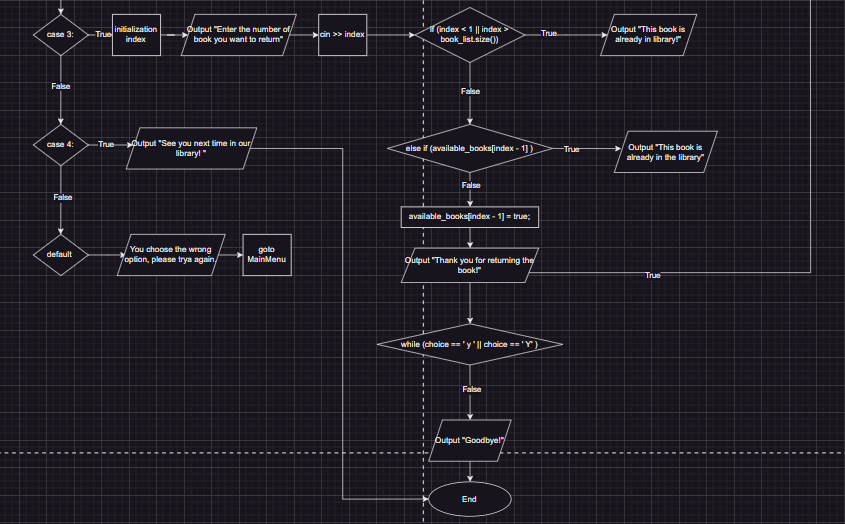
**Програма №4**

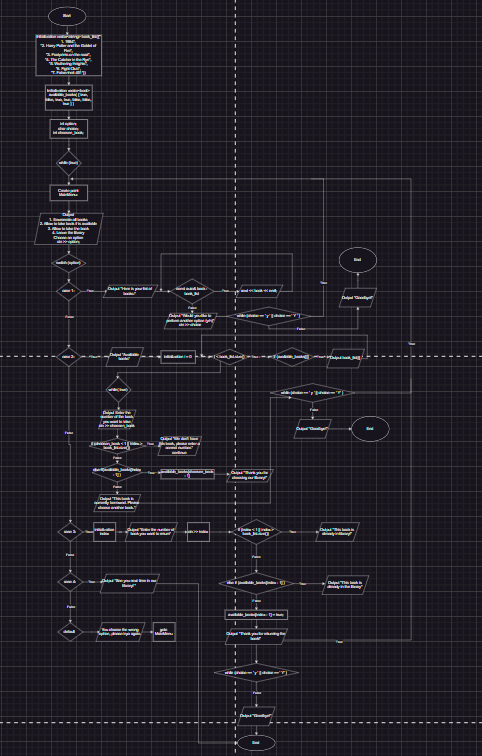
* Блок-схема
* Важливі деталі для реалізації програми
* Написати перевантажені функції та викликати їх у головній функції
* Плановий час на реалізацію 2 години.

  
*Рисунок 2.4. Блок-схема до програми 4*

**Програма №5**

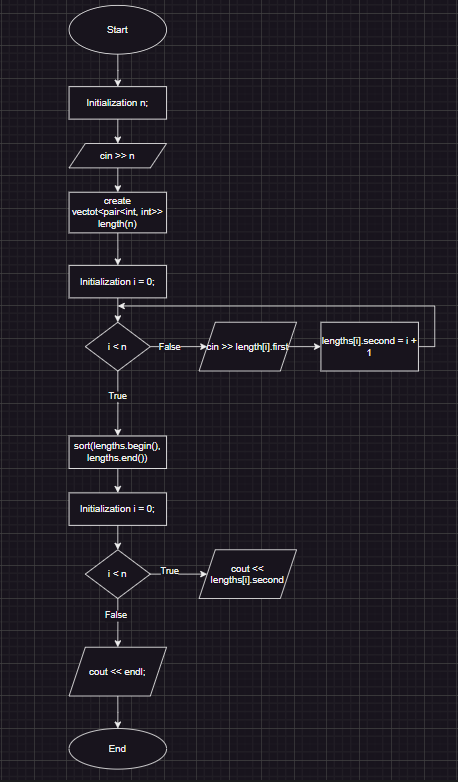
* Блок-схема
* Важливі деталі для реалізації програми
* Плановий час на реалізацію 1 день 

 **  
*Рисунок 2.5. Блок-схема до програми №5*

  
*Рисунок 2.6. Загальна блок-схема до програми №5*

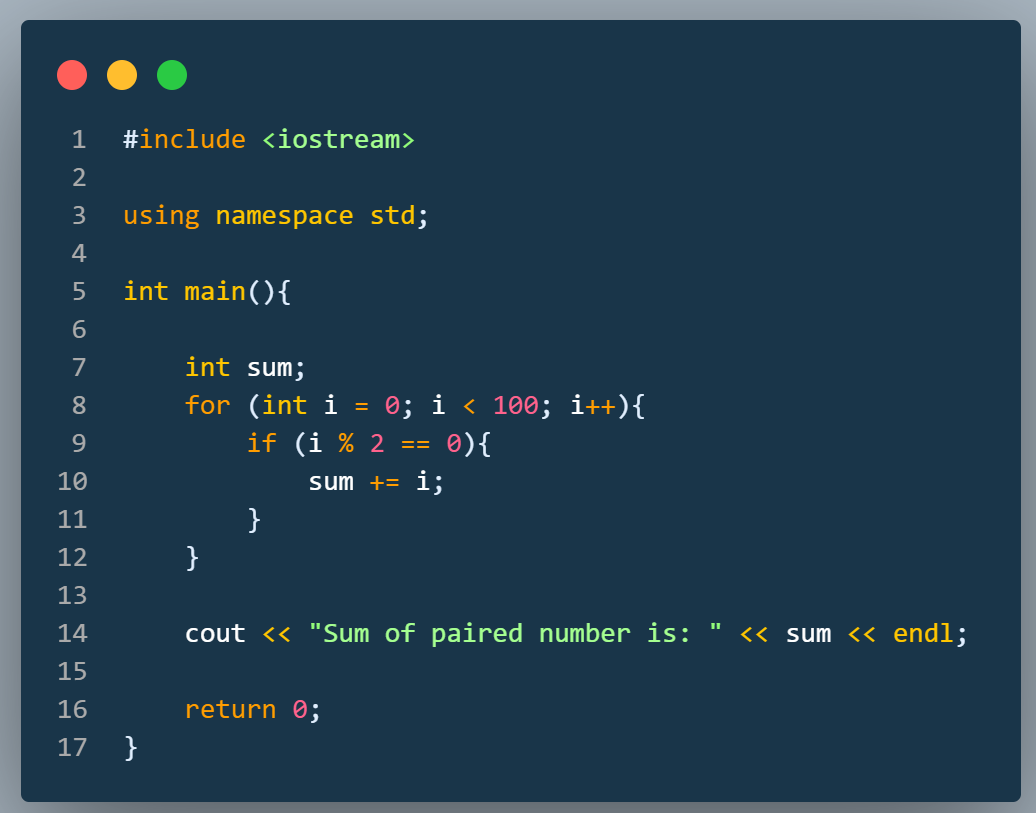
Програма №6

* Блок-схема
* Важливі деталі для реалізації програми
* Використати цикл для перевірки елементів у векторі.
* Плановий час на реалізацію 1 день

  
*Рисунок 2.7. Блок-схема до програми №6*

**3. Код програм з посиланням на зовнішні ресурси та фактично затрачений час:**

**Завдання №1**

  
*Рисунок 3.1. Код до програми №1*

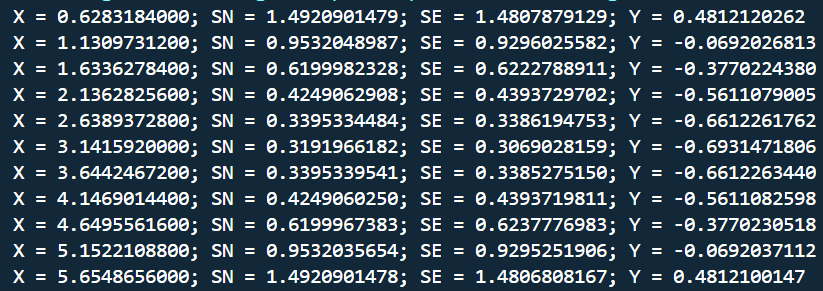
*  
Рисунок 3.2. Приклад виконання програми №1*

На початку ми ініціалізуємо змінну sum в яку запишемо кінцевий результат, тоді за допомогою циклу ми проходимося по числах від 0 до 100 при тому будемо перевіряти їхню остачу від ділення, якщо вона 0 то це число парне і ми його додаємо.

Фактично затрачений час 25 хвилин.  
**Посилання на файл у пулл реквесті**https://github.com/artificial-intelligence-department/ai\_programming\_playground\_2024/pull**/147**

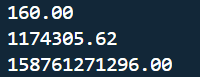
**Завдання №2**

*Рисунок 3.3. Код до програми №2*

*****Рисунок 3.4. Приклад виконання програми №2*Спочатку ми ініціалізуємо константи типу double, далі створюємо три функції які повертають значення f (для обчислення точного значення функції), SN(для обчислення функції із заданою точністю n), SE (для обчислення функції із точністю епсілон).

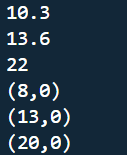
Фактично затрачений час 2 години.  
**Посилання на файл у пулл реквесті**https://github.com/artificial-intelligence-department/ai\_programming\_playground\_2024/pull/147

**Завдання №3**

  
*Рисунок 3.5. Код до програми №3* *Рисунок 3.6. Приклад виконання програми №3*

Спочатку ми створюємо функцію mult зі змінною кількістю параметрів типу float, яка буде їх додавати. У функції за допомогою va\_list arguments ми створили список аргументів далі va\_start(arguments, first) ініціалізували доступ до додаткових аргументів. У циклі за допомогою команди va\_arg(arguments, double) отримали наступний елемент зі списку ну і va\_end(arguments) завершили роботу зі списком. Відповідно у головній функції ми спочатку встановили скільки хочемо цифр після коми для кожного результату, потім викликали функцію mult де вказали спочатку число наших параметрів і написали, які саме параметри потрібно додати.  
Фактично затрачений час 2 години.  
**Посилання на файл у пулл реквесті**https://github.com/artificial-intelligence-department/ai\_programming\_playground\_2024/pull/147

**Завдання №4**  
*Рисунок 3.7. Код до програми №4*

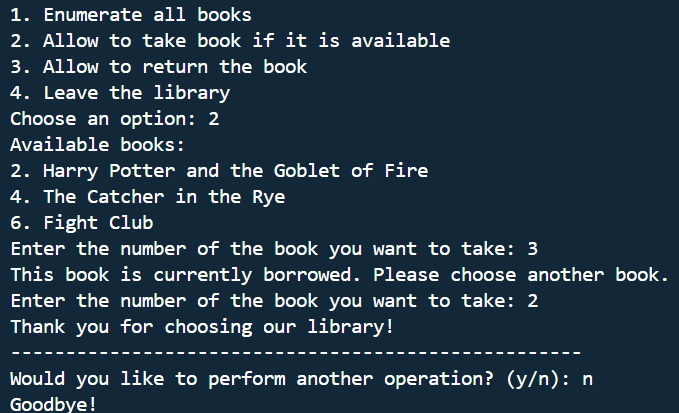
  
*Рисунок 3.8. Приклад виконання програми №4*

Спочатку створюємо 3 перевантажені функції addition\_real\_numbers, які будуть додавати три цілі числа, потім 3 перевантажені функції addition\_complex\_numbers, які відповідно будуть додавати комплексні числа. У головній функції ми будемо викликати тричі одну функцію і тричі другу, щоб додати різну кількість чисел.

**Посилання на файл у пулл реквесті**https://github.com/artificial-intelligence-department/ai\_programming\_playground\_2024/pull/147

**Завдання №5**

  
*Рисунок 3.9. Код до програми №5*

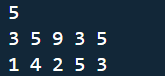
*  
Рисунок 3.10. Приклад виконання програми №5*

Програма №5 це онлайн бібліотека. Користувачу на екран виводимо можливі дії і відносно того, що користувач уведе програма буде виводити різні результати на екран.  
Фактично затрачений час 5 години.

**Посилання на файл у пулл реквесті**https://github.com/artificial-intelligence-department/ai\_programming\_playground\_2024/pull/147

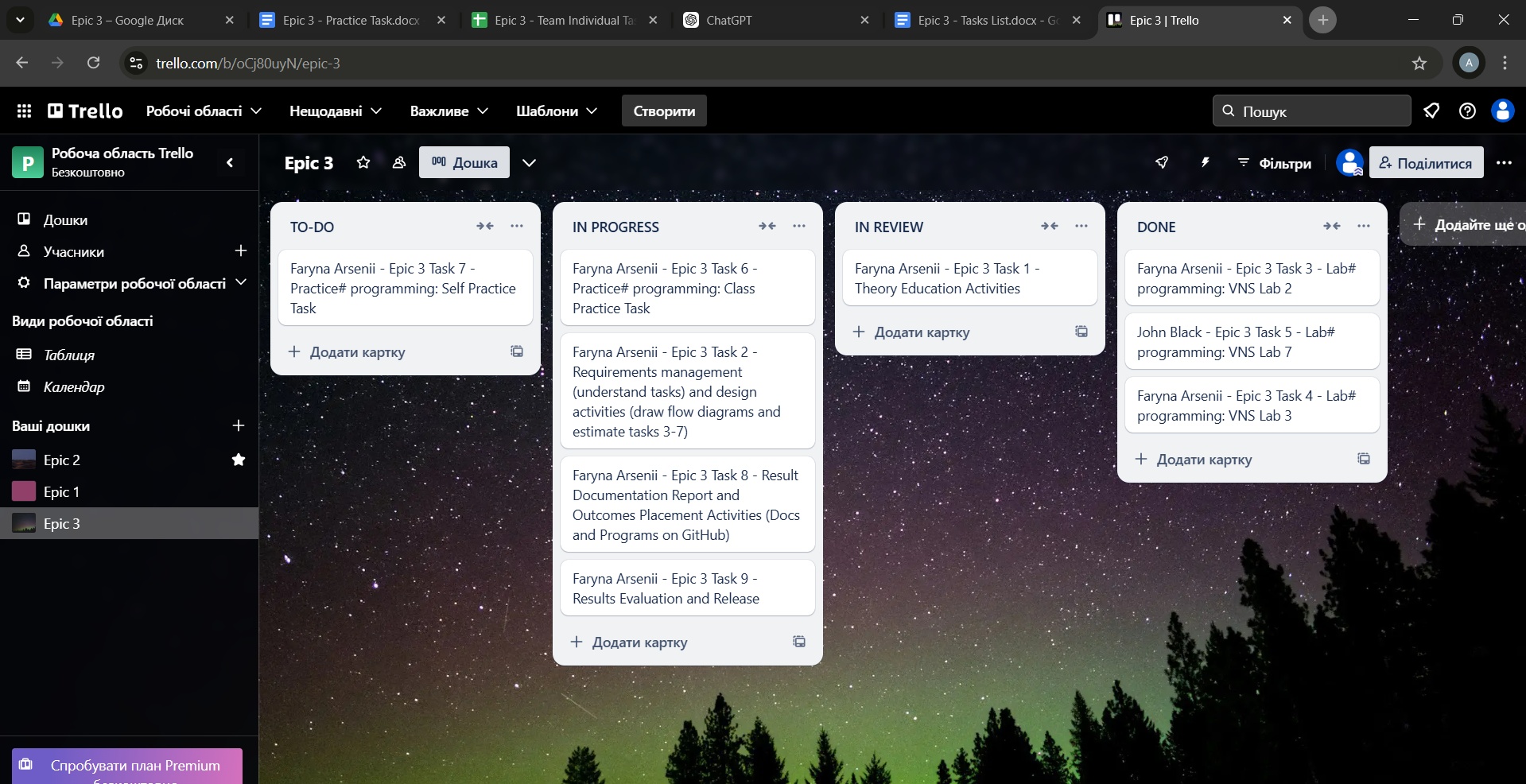
**Завдання №6**

*Рисунок 3.11. Код до програми №6*

*  
Рисунок 3.12. Приклад виконання програми №6*

Програма приймає від користувача число, тобто кількість офісів і записує вектор пар для довжин і індексів. Далі користувач заповнює вектор і ми зберігаємо індекси. Потім сортуємо і виводимо порядок індексів (компаній), які треба розташувати в оптимальному порядку, щоб відстань була найменшою.  
Фактично затрачений час 2 години.  
**Посилання на файл у пулл реквесті**

1. **Робота з командою:**

  
*Рисунок 4.1. Командна дошка в Trello*

**Висновок:** У межах практичних та лабораторних робіт блоку №3 я вивчив низку нових понять, таких як: функція, перевантажена функція, функція зі змінною кількістю параметрів (еліпсис), цикли, вкладені цикли, простір імен, завершення виконання циклів, рекурсія. Деякі з них довелося застосувати на практиці, що дало краще розуміння як це все працює на ділі. Для кращого розуміння як працює та чи інша програма я створив діаграми в Draw.io, тому також покращив роботу з цим редактором .Також створив чергову дошку в Trello для роботи в команді.