

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



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

Виконала:

Студентка групи ШІ-11

Цибух Андріана

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:

Ознайомитися із значенням та роллю циклів у програмуванні, їхніми видами, синтаксисом та основою використання; вміти оголошувати функції та простір імен; розумітися у еліпсисі та рекурсії.

Теоретичні відомості:

1) Теоретичні відомості з переліком важливих тем:

- Тема №*.1: Цикли.
- Тема №*.2: Функції.

2) Індивідуальний план опрацювання теорії:

- Тема №*.1: Цикли.
 - Джерела Інформації
 - Лекції О. Пшеничного.
 - Сайт **W3Shools: While Loop, Do/While Loop, For Loop.**
 - Статус: Ознайомлена
- Тема №*.2: Функції.
 - Джерела Інформації:
 - Лекція О. Пшеничного.
 - Сайт **W3Schools C++ Recursion.**
 - Ютуб
 - Статус: Ознайомлена

Виконання роботи:

1. Опрацювання завдання та вимог до програм та середовища:

Завдання №1 Менеджмент бібліотеки

- Деталі завдання : Створити просту програму керування бібліотекою. Книги в бібліотеці є, користувачі можуть їх взяти або повернути.
- Важливі деталі для врахування в імплементації програми : використати цикли while, do while, for, for each, goto

Завдання №2 VNS Labs 2

- Варіант завдання : 13
- Деталі завдання : Використовуючи оператор циклу, знайти суму елементів, зазначених у конкретному варіанті. Результат надрукувати, надавши відповідний заголовок. Знайти суму ряду з точністю $\varepsilon=0.0001$, загальний член якого

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

Завдання №3 VNS Labs 3

- Варіант завдання : 13
- Деталі завдання : Для x , що змінюється від 0,2 до 1 з кроком 0,08, 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 $\frac{1}{2} \ln(x)$

Завдання №4 VNS Labs 7 Task 1

- Варіант завдання : 13
- Деталі завдання : 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.

Завдання №5 VNS Labs 7 Task 2

- Варіант завдання : 13
- Деталі завдання :
 - а) для перетворення десяткового дробу у звичайний;
 - б) для перетворення звичайного дробу у десятковий.

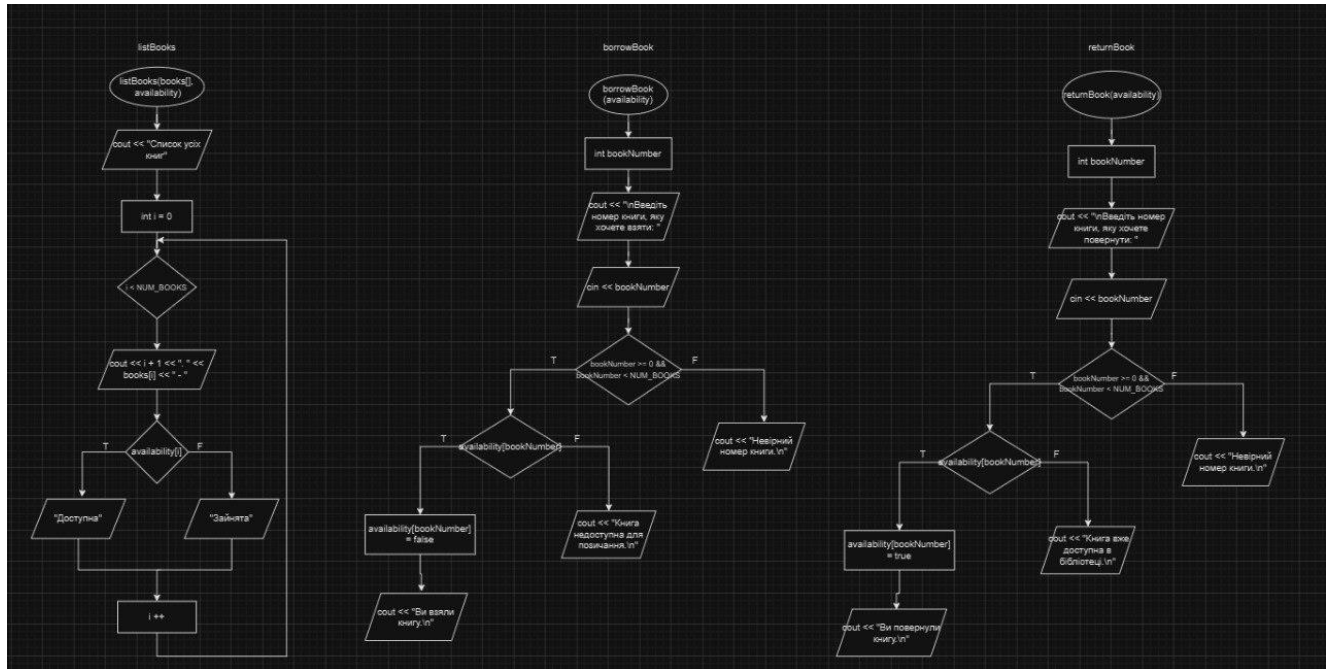
Завдання №6 Algotester: Офісна вулиця. Частина 1

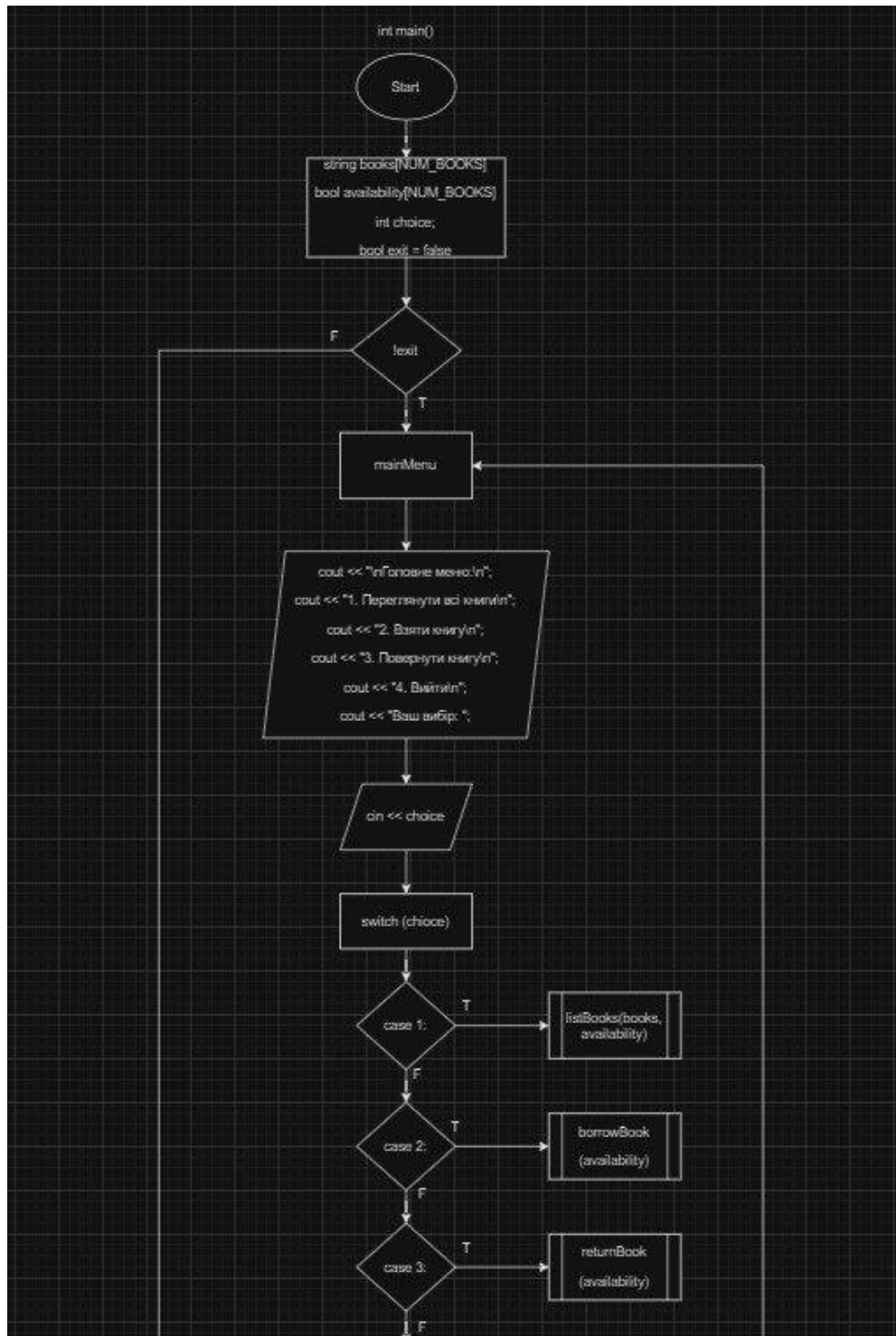
- Деталі завдання : Зустрілися якось працівники великих компаній і почали... Обговорювати план вулиці. Виявляється, всі приміщення, які орендуватимуть ці компанії, збудують вздовж однієї вулиці. i -та компанія орендуватиме офіс довжиною l_i метрів. Офіси будуватимуть один за одним, починаючи з точки 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 p_1, p_2, \dots, p_n , то перший офіс почнеться в точці 0 і закінчиться в точці l_{p_1} , другий почнеться в l_{p_1} і закінчиться в $l_{p_1} + l_{p_2}$ 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.

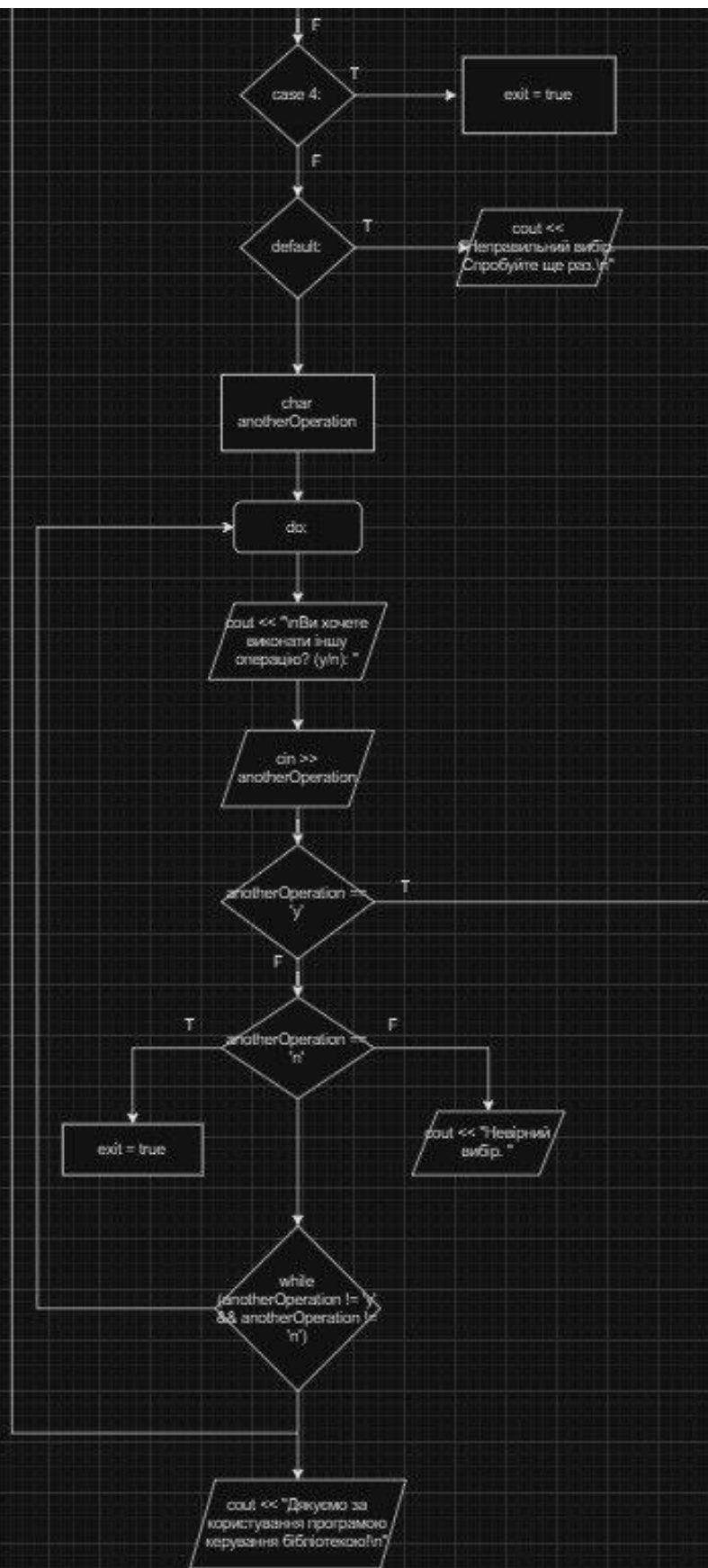
2. Дизайн та планована оцінка часу виконання завдань:

Програма №1 Менеджмент бібліотеки

- Планований час на реалізацію:
2 год

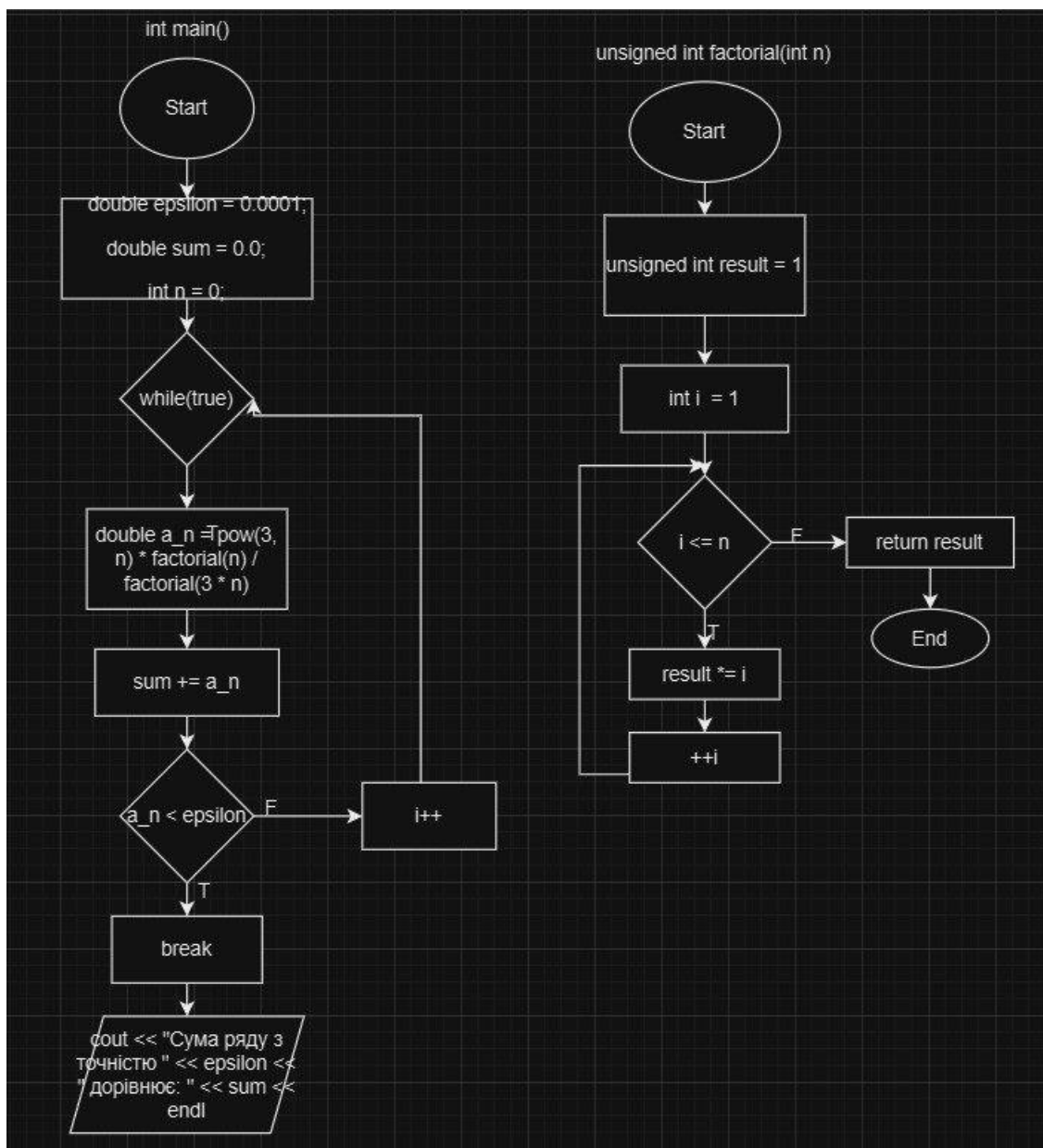






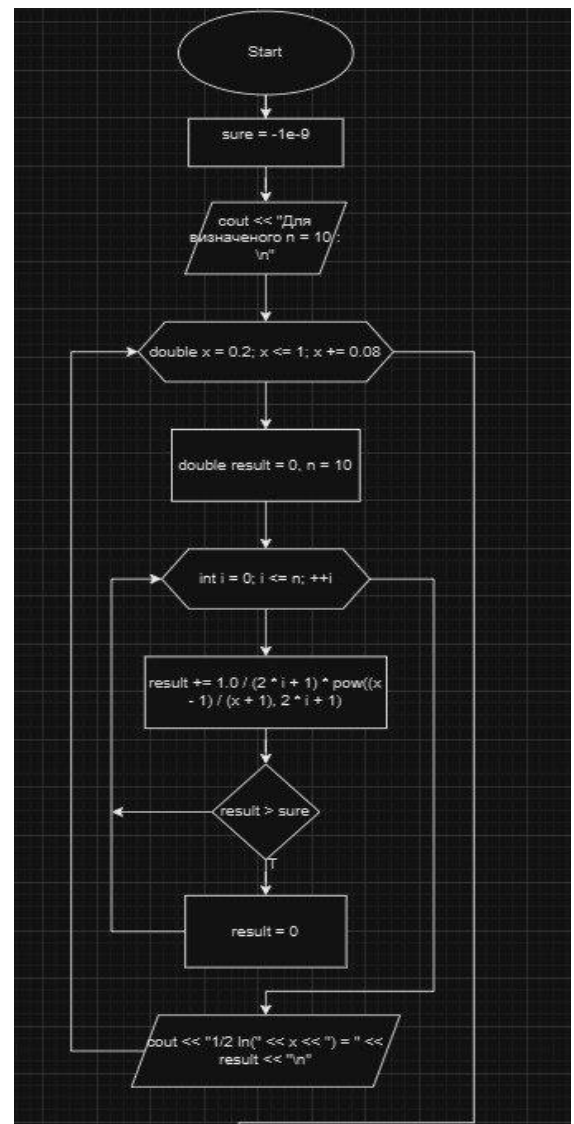
Програма №2 VNS Labs 2

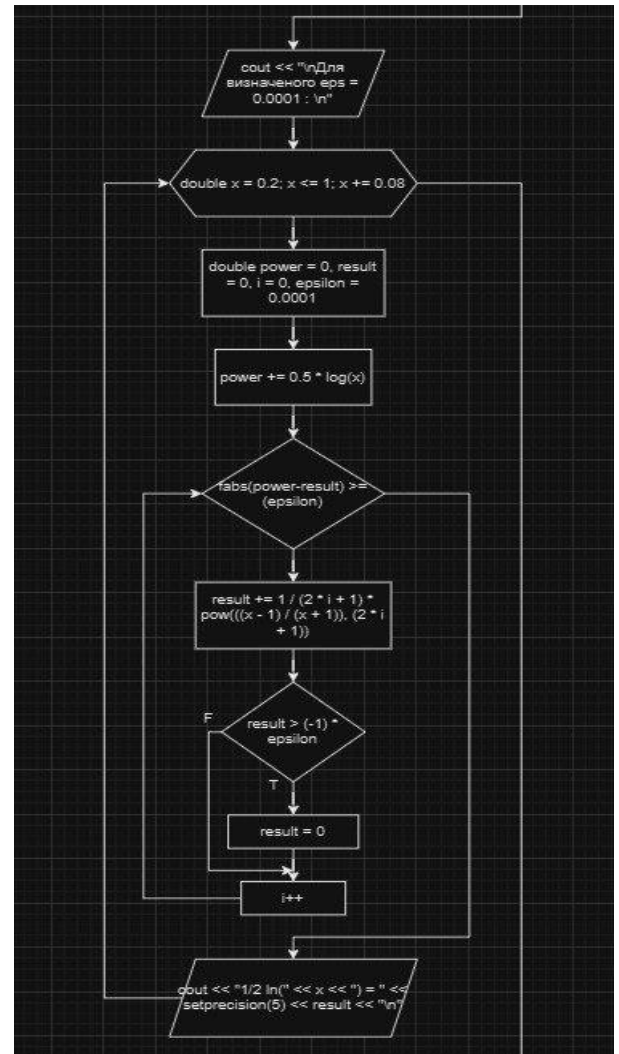
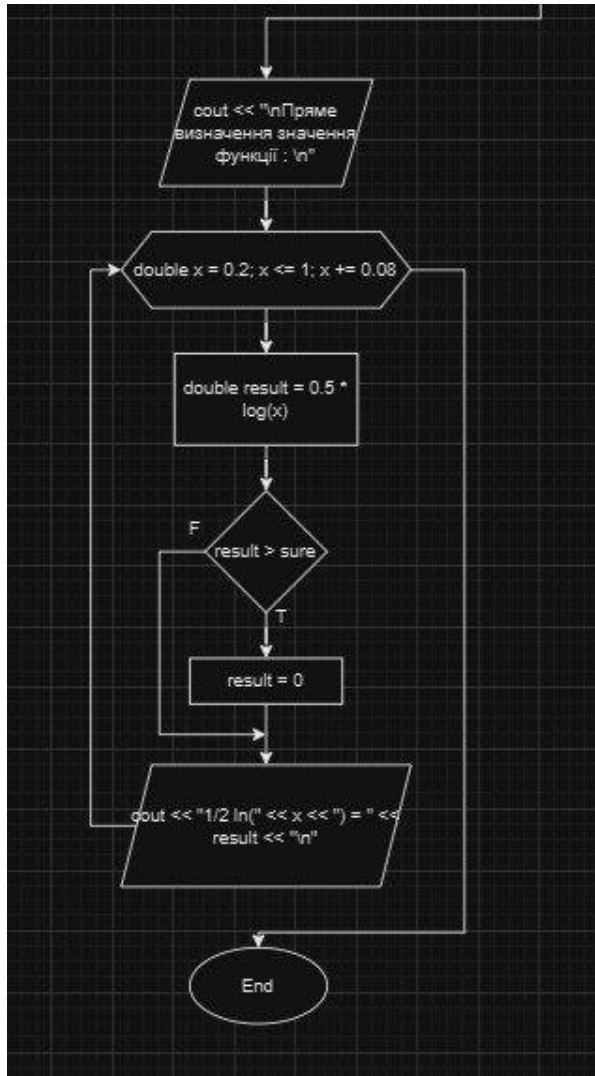
- Планований час на реалізацію : 20 хв



Програма №3 VNS Labs 3

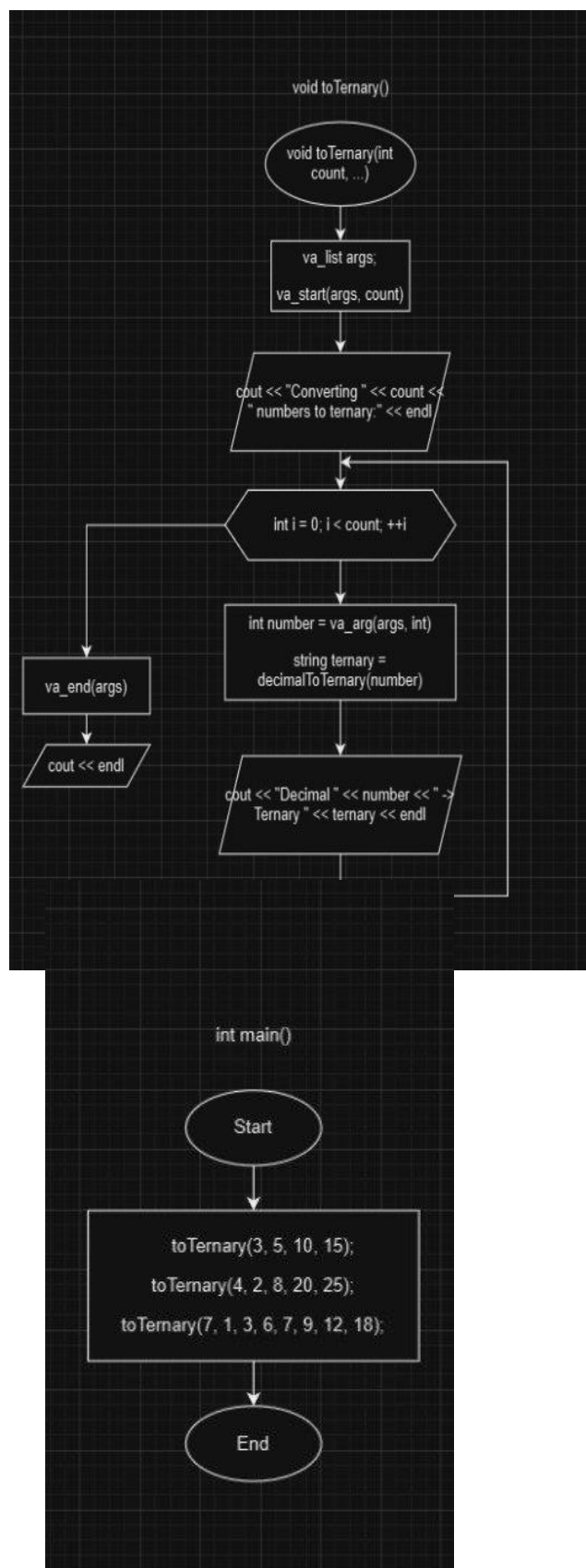
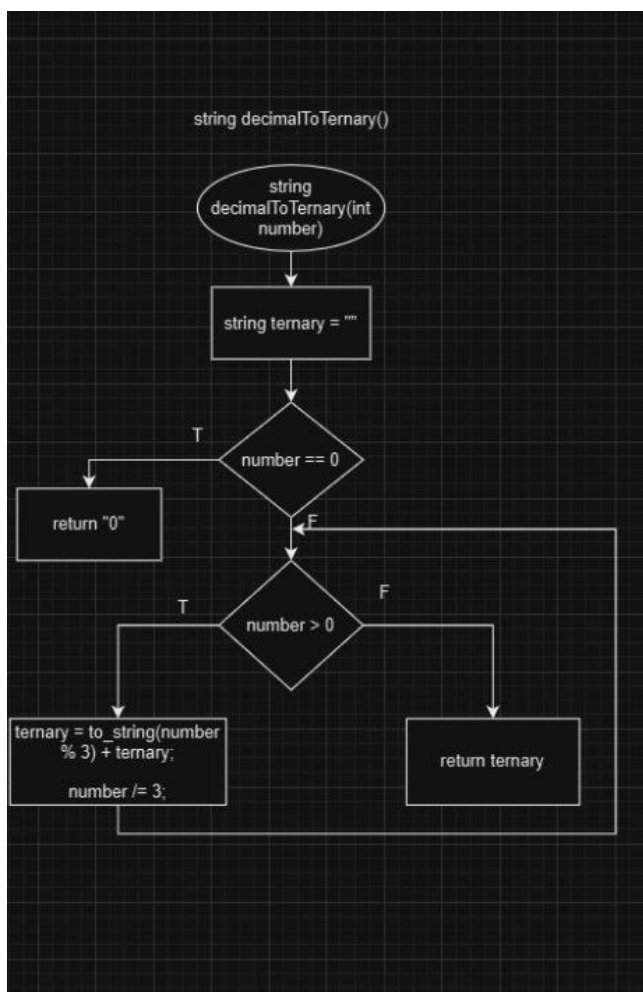
- Планований час на реалізацію :
1 година





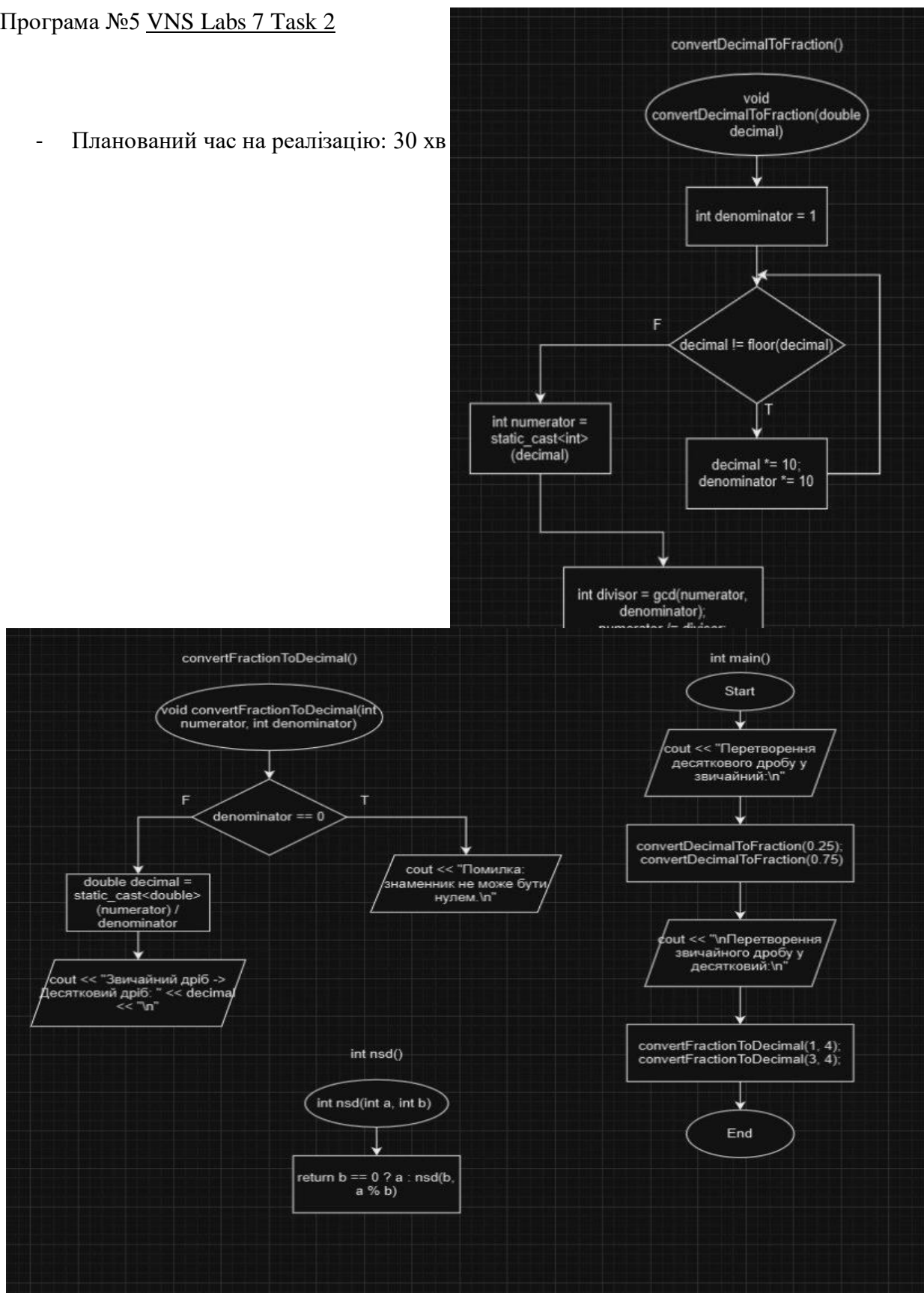
Програма №4 VNS Labs 7 Task 1

- Планований час на реалізацію :
40 хв



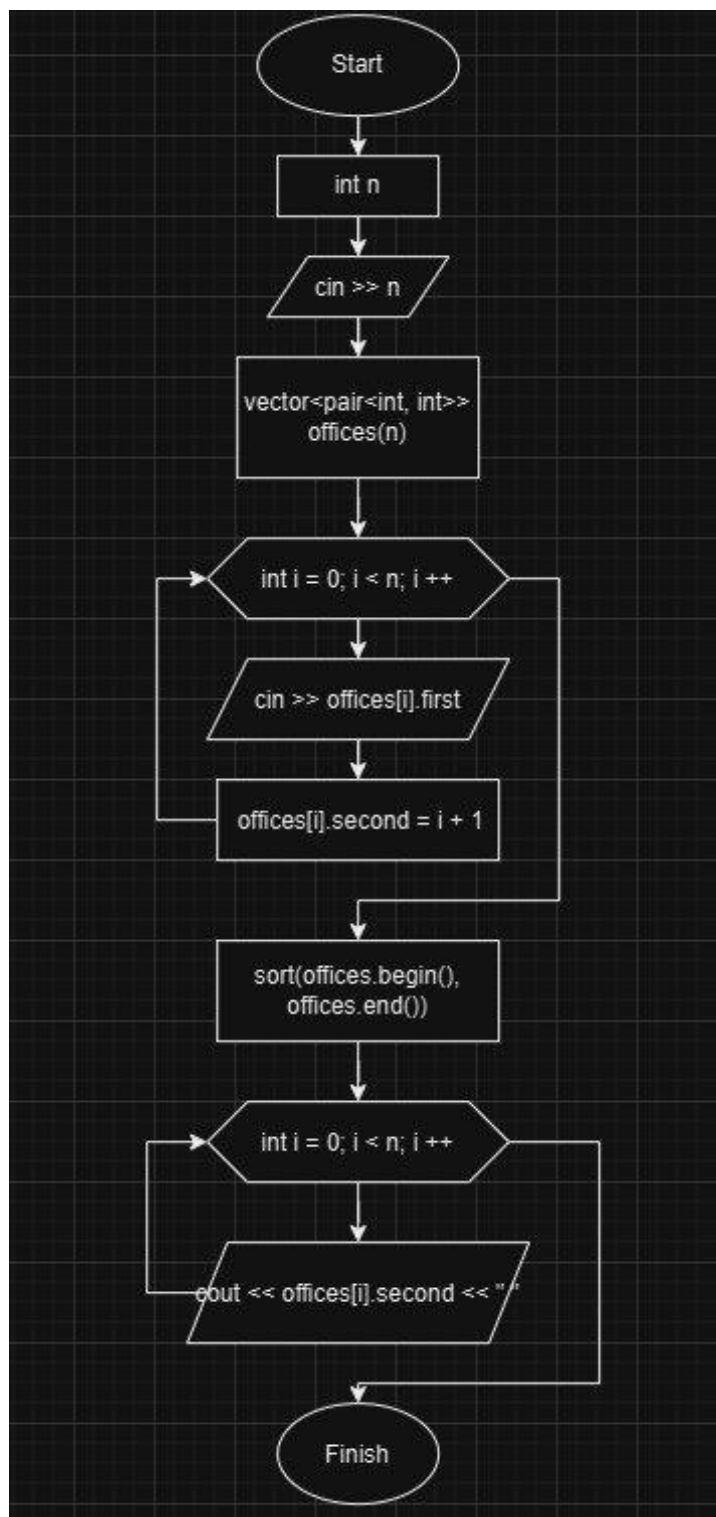
Програма №5 VNS Labs 7 Task 2

- Планований час на реалізацію: 30 хв



Програма №6 Algotester: Офісна вулиця. Частина 1

- Планований час на реалізацію: 15 хв



3. Код програм з посиланням на зовнішні ресурси:

Завдання №1 Менеджмент бібліотеки

Посилання на файл програми у пул-запиті GitHub

```
1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  const int NUM_BOOKS = 4;
7
8  void listBooks(const string books[], const bool availability[]) {
9      cout << "\nВсі книги:\n";
10     for (int i = 0; i < NUM_BOOKS; ++i) {
11         cout << i + 1 << ". " << books[i] << " - "
12             << (availability[i] ? "Доступна" : "Зайнята") << "\n";
13     }
14 }
15
16 void borrowBook(bool availability[]) {
17     int bookNumber;
18     cout << "\nВведіть номер книги, яку хочете взяти: ";
19     cin >> bookNumber;
20     --bookNumber; // Зменшуємо, щоб привести до індексу масиву
21
22     if (bookNumber >= 0 && bookNumber < NUM_BOOKS) {
23         if (availability[bookNumber]) {
24             availability[bookNumber] = false;
25             cout << "Ви взяли книгу.\n";
26         } else {
27             cout << "Книга недоступна для позичання.\n";
28         }
29     } else {
30         cout << "Невірний номер книги.\n";
31     }
32 }
33
34 void returnBook(bool availability[]) {
35     int bookNumber;
36     cout << "\nВведіть номер книги, яку хочете повернути: ";
37     cin >> bookNumber;
38     --bookNumber;
39
40     if (bookNumber >= 0 && bookNumber < NUM_BOOKS) {
41         if (!availability[bookNumber]) {
42             availability[bookNumber] = true;
43             cout << "Ви повернули книгу.\n";
44         } else {
45             cout << "Книга вже доступна в бібліотеці.\n";
46         }
47     } else {
48         cout << "Невірний номер книги.\n";
49     }
50 }
51
```

```

62 int main() {
63     string books[NUM_BOOKS] = {"Із крові й попелу", "Гаррі Поттер", "Панк 57", "Самотність простих чисел"};
64     bool availability[NUM_BOOKS] = {true, true, true, true}; // Ініціалізуємо всі книги як доступні
65
66     int choice;
67     bool exit = false;
68
69     while (!exit) {
70         mainMenu: // Мітка для повернення до головного меню за допомогою goto
71         cout << "\nГоловне меню:\n";
72         cout << "1. Переглянути всі книги\n";
73         cout << "2. Взяти книгу\n";
74         cout << "3. Повернути книгу\n";
75         cout << "4. Вийти\n";
76         cout << "Ваш вибір: ";
77         cin >> choice;
78
79         switch (choice) {
80             case 1:
81                 listBooks(books, availability);
82                 break;
83             case 2:
84                 borrowBook(availability);
85                 break;
86             case 3:
87                 returnBook(availability);
88                 break;
89             case 4:
90                 exit = true;
91                 continue; // Переходимо до завершення циклу while
92             default:
93                 cout << "Неправильний вибір. Спробуйте ще раз.\n";
94                 goto mainMenu; // Використання goto для повернення до меню
95         }
96
97         char anotherOperation;
98         do {
99             cout << "\nВи хочете виконати іншу операцію? (y/n): ";
100             cin >> anotherOperation;
101             if (anotherOperation == 'y') {
102                 goto mainMenu;
103             } else if (anotherOperation == 'n') {
104                 exit = true;
105             } else {
106                 cout << "Невірний вибір. ";
107             }
108         } while (anotherOperation != 'y' && anotherOperation != 'n');
109     }
110
111     cout << "Дякуємо за користування програмою керування бібліотекою!\n";
112     return 0;
113 }

```

Завдання №2 VNS Labs 2

```
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  // Факторіал
6  unsigned int factorial(int n) {
7      unsigned int result = 1;
8      for (int i = 1; i <= n; ++i) {
9          result *= i;
10     }
11     return result;
12 }
13
14 int main() {
15     double epsilon = 0.0001;
16     double sum = 0.0;
17     int n = 0;
18
19     while (true) {
20         double a_n = pow(3, n) * factorial(n) / factorial(3 * n);
21         sum += a_n;
22         if (a_n < epsilon) {
23             break;
24         }
25         n++;
26     }
27
28     cout << "Сума ряду з точністю " << epsilon << " дорівнює: " << sum << endl;
29
30     return 0;
31 }
32
```

Завдання №3 VNS Labs 3

Посилання на файл програми у пул-запиті GitHub

```
1  #include <iostream>
2  #include <iomanip>
3  #include <cmath>
4  using namespace std;
5
6  int main() {
7      const double sure = -1e-9;
8
9      // Для визначеного n = 10
10     cout << "Для визначеного n = 10 : \n";
11     for(double x = 0.2; x <= 1; x += 0.08) {
12         double result = 0, n = 10;
13
14         for(int i = 0; i <= n; ++i) {
15             result += 1.0 / (2 * i + 1) * pow((x - 1) / (x + 1), 2 * i + 1);
16             if (result > sure) {
17                 result = 0;
18             }
19         }
20         cout << "1/2 ln(" << x << ") = " << result << "\n";
21     }
22
23     // Для визначеного eps = 0.0001
24     cout << "\nДля визначеного eps = 0.0001 : \n";
25     for(double x = 0.2; x <= 1; x += 0.08) {
26         double power = 0, result = 0, i = 0, epsilon = 0.0001;
27         power += 0.5 * log(x);
28
29         while (fabs(power - result) >= (epsilon)) {
30             result += 1 / (2 * i + 1) * pow(((x - 1) / (x + 1)), (2 * i + 1));
31             if (result > (-1) * epsilon) {
32                 result = 0;
33             }
34             i++;
35         }
36         cout << "1/2 ln(" << x << ") = " << setprecision(5) << result << "\n";
37     }
38
39     // Пряме визначення значення функції
40     cout << "\nПряме визначення значення функції : \n";
41     for(double x = 0.2; x <= 1; x += 0.08) {
42         double result = 0.5 * log(x);
43
44         if (result > sure) {
45             result = 0;
46         }
47         cout << "1/2 ln(" << x << ") = " << result << "\n";
48     }
49
50     return 0;
51 }
52
```

Завдання №4 VNS Labs 7 Task 1

Посилання на файл програми у пул-запиті GitHub

```
1  #include <iostream>
2  #include <cstdarg>
3  #include <string>
4  using namespace std;
5
6  // перетворення числа з десятичної в трійкову
7  string decimalToTernary(int number) {
8      string ternary = "";
9      if (number == 0) {
10         return "0";
11     }
12     while (number > 0) {
13         ternary = to_string(number % 3) + ternary;
14         number /= 3;
15     }
16     return ternary;
17 }
18
19 // перетворення з десятичної в трійкову
20 void toTernary(int count, ...) {
21     va_list args;
22     va_start(args, count);
23     cout << "Converting " << count << " numbers to ternary:" << endl;
24
25     for (int i = 0; i < count; ++i) {
26         int number = va_arg(args, int);
27         string ternary = decimalToTernary(number);
28         cout << "Decimal " << number << " -> Ternary " << ternary << endl;
29     }
30
31     va_end(args);
32     cout << endl;
33 }
34
35 int main() {
36     toTernary(3, 5, 10, 15);
37     toTernary(4, 2, 8, 20, 25);
38     toTernary(7, 1, 3, 6, 7, 9, 12, 18);
39
40     return 0;
41 }
42
```

Завдання №5 VNS Labs 7 Task 2

Посилання на файл програми у пул-запиті GitHub

```
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  // Функція для знаходження НСД (найбільшого спільного дільника)
7  int nsd(int a, int b) {
8      return b == 0 ? a : nsd(b, a % b);
9  }
10
11 // Перетворення десяткового дробу у звичайний
12 void convertDecimalToFraction(double decimal) {
13     int denominator = 1;
14
15     // Перетворення у ціле число
16     while (decimal != floor(decimal)) {
17         decimal *= 10;
18         denominator *= 10;
19     }
20
21     int numerator = static_cast<int>(decimal);
22
23     // Спрощення дробу
24     int divisor = nsd(numerator, denominator);
25     numerator /= divisor;
26     denominator /= divisor;
27
28     cout << "Десятковий дріб -> Звичайний дріб: " << numerator << "/" << denominator << "\n";
29 }
30
31 // Перетворення звичайного дробу у десятковий
32 void convertFractionToDecimal(int numerator, int denominator) {
33     if (denominator == 0) {
34         cout << "Помилка: знаменник не може бути нулем.\n";
35         return;
36     }
37
38     double decimal = static_cast<double>(numerator) / denominator;
39     cout << "Звичайний дріб -> Десятковий дріб: " << decimal << "\n";
40 }
41
42 int main() {
43     // Тестування функцій
44     cout << "Перетворення десяткового дробу у звичайний:\n";
45     convertDecimalToFraction(0.25); // Очікуваний результат: 1/4
46     convertDecimalToFraction(0.75); // Очікуваний результат: 3/4
47
48     cout << "\nПеретворення звичайного дробу у десятковий:\n";
49     convertFractionToDecimal(1, 4); // Очікуваний результат: 0.25
50     convertFractionToDecimal(3, 4); // Очікуваний результат: 0.75
51
52     return 0;
53 }
54
```

Завдання №6 Algotester: Офісна вулиця. Частина 1

Посилання на файл програми у пул-запиті GitHub

```
1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4  using namespace std;
5
6  int main() {
7      int n;
8      cin >> n;
9      vector<pair<int, int>> offices(n);
10
11     for(int i = 0; i < n; i++) {
12         cin >> offices[i].first;
13         offices[i].second = i + 1;
14     }
15
16     sort(offices.begin(), offices.end());
17
18     for(int i = 0; i < n; i++) {
19         cout << offices[i].second << " ";
20     }
21
22     return 0;
23 }
24
```

4. Результати виконання завдань, тестування та фактично затрачений час:
Завдання №1 Менеджмент бібліотеки

Головне меню:

1. Переглянути всі книги
2. Взяти книгу
3. Повернути книгу
4. Вийти

Ваш вибір: 1

Список усіх книг:

1. Із крові й попелу - Доступна
2. Гаррі Поттер - Доступна
3. Панк 57 - Доступна
4. Самотність простих чисел - Доступна

Ви хочете виконати іншу операцію? (y/n): y

Головне меню:

1. Переглянути всі книги
2. Взяти книгу
3. Повернути книгу
4. Вийти

Ваш вибір: 2

Введіть номер книги, яку хочете взяти: 1

Ви взяли книгу.

Ви хочете виконати іншу операцію? (y/n): y

Головне меню:

1. Переглянути всі книги
2. Взяти книгу
3. Повернути книгу
4. Вийти

Ваш вибір: 3

Введіть номер книги, яку хочете повернути: 1

Ви повернули книгу.

Ви хочете виконати іншу операцію? (y/n): y

Головне меню:

1. Переглянути всі книги
2. Взяти книгу
3. Повернути книгу
4. Вийти

Ваш вибір: 2

Введіть номер книги, яку хочете взяти: 6

Невірний номер книги.

Ви хочете виконати іншу операцію? (y/n): y

Головне меню:

1. Переглянути всі книги
2. Взяти книгу
3. Повернути книгу
4. Вийти

Ваш вибір: 4

Дякуємо за користування програмою керування бібліотекою!

Час затрачений на виконання завдання : 2 год

Завдання №2 VNS Labs 2

Сума ряду з точністю 0.0001 дорівнює: 1.52545

Час затрачений на виконання завдання : 20 хв

Завдання №3 VNS Labs 3

```
Для визначеного n = 10 :
1/2 ln(0.2) = -0.804712
1/2 ln(0.28) = -0.636483
1/2 ln(0.36) = -0.510826
1/2 ln(0.44) = -0.41049
1/2 ln(0.52) = -0.326963
1/2 ln(0.6) = -0.255413
1/2 ln(0.68) = -0.192831
1/2 ln(0.76) = -0.137218
1/2 ln(0.84) = -0.0871767
1/2 ln(0.92) = -0.0416908
1/2 ln(1) = 0

Для визначеного eps = 0.0001 :
1/2 ln(0.2) = -0.80462
1/2 ln(0.28) = -0.63642
1/2 ln(0.36) = -0.5108
1/2 ln(0.44) = -0.41046
1/2 ln(0.52) = -0.32691
1/2 ln(0.6) = -0.2554
1/2 ln(0.68) = -0.19278
1/2 ln(0.76) = -0.13721
1/2 ln(0.84) = -0.087176
1/2 ln(0.92) = -0.041667
1/2 ln(1) = 0

Пряме визначення значення функції :
1/2 ln(0.2) = -0.80472
1/2 ln(0.28) = -0.63648
1/2 ln(0.36) = -0.51083
1/2 ln(0.44) = -0.41049
1/2 ln(0.52) = -0.32696
1/2 ln(0.6) = -0.25541
1/2 ln(0.68) = -0.19283
1/2 ln(0.76) = -0.13722
1/2 ln(0.84) = -0.087177
1/2 ln(0.92) = -0.041691
1/2 ln(1) = 0
```

Час затрачений на виконання завдання : 30 хв

Завдання №4 VNS Labs 7 Task 1

```
Converting 3 numbers to ternary:  
Decimal 5 -> Ternary 12  
Decimal 10 -> Ternary 101  
Decimal 15 -> Ternary 120  
  
Converting 4 numbers to ternary:  
Decimal 2 -> Ternary 2  
Decimal 8 -> Ternary 22  
Decimal 20 -> Ternary 202  
Decimal 25 -> Ternary 221  
  
Converting 7 numbers to ternary:  
Decimal 1 -> Ternary 1  
Decimal 3 -> Ternary 10  
Decimal 6 -> Ternary 20  
Decimal 7 -> Ternary 21  
Decimal 9 -> Ternary 100  
Decimal 12 -> Ternary 110  
Decimal 18 -> Ternary 200
```

Час затрачений на виконання завдання : 30 хв

Завдання №5 VNS Labs 7 Task 2

```
Перетворення десяткового дробу у звичайний:  
Десятковий дріб -> Звичайний дріб: 1/4  
Десятковий дріб -> Звичайний дріб: 3/4  
  
Перетворення звичайного дробу у десятковий:  
Звичайний дріб -> Десятковий дріб: 0.25  
Звичайний дріб -> Десятковий дріб: 0.75
```

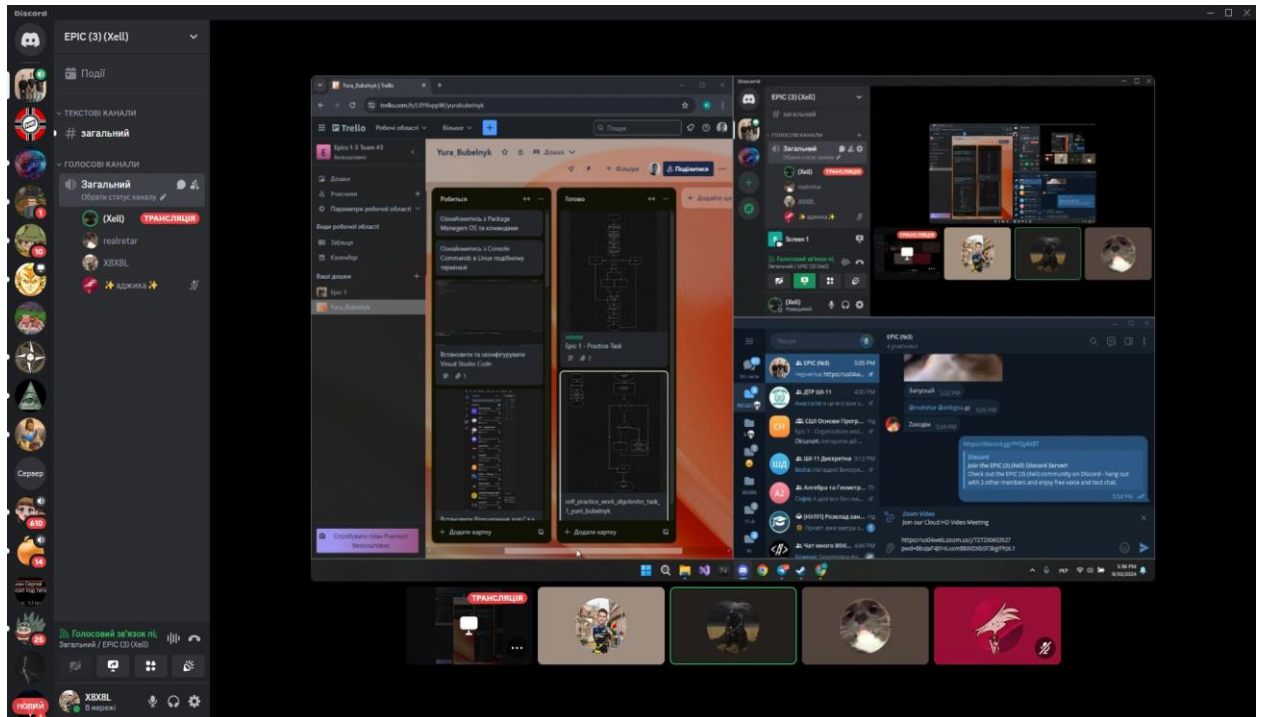
Час затрачений на виконання завдання : 40 хв

Завдання №6 Algotester: Офісна вулиця. Частина 1

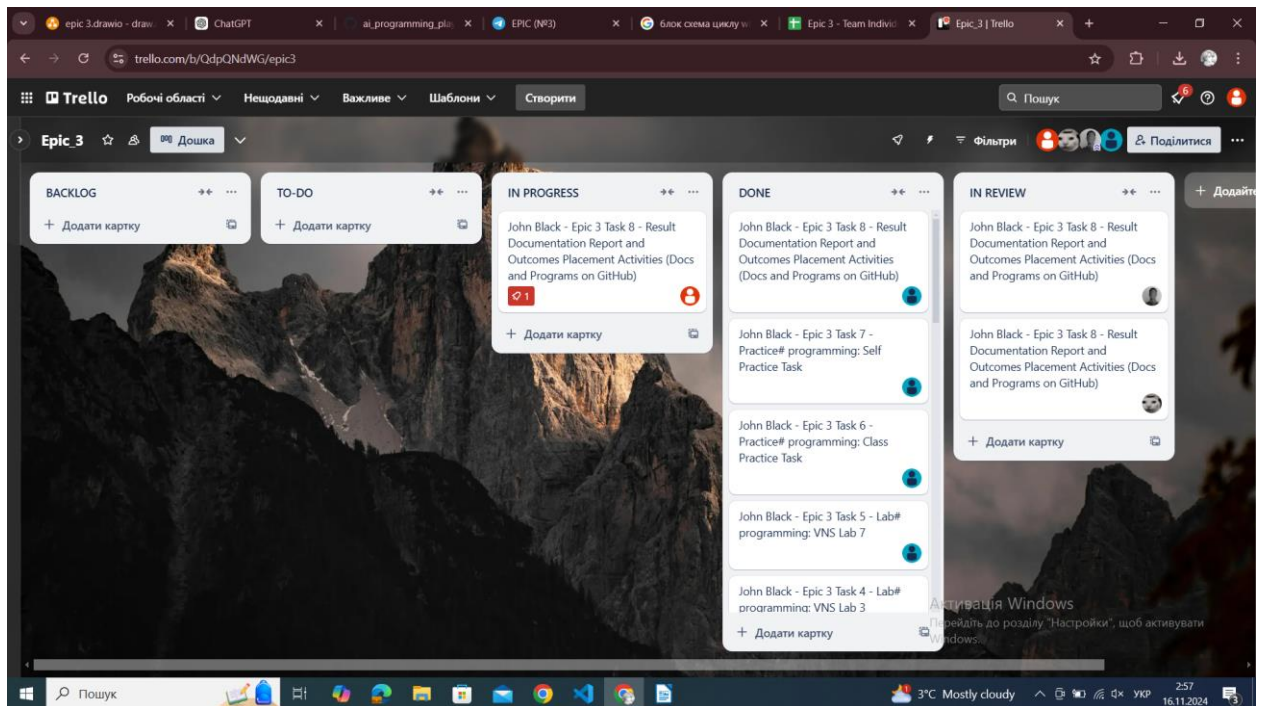
```
4  
4 1 2 3  
2 3 4 1
```

Час затрачений на виконання завдання : 10 хв

5. Кооперація з командою:



6. Trello команди:



Висновки:

На цьому епіку я ознайомилась із роллю та значенням циклів у програмуванні, їхніми видами, синтаксисом та основою використання; навчилась оголошувати функції та простір імен; розуміюся у еліпсисі та рекурсії.