

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

**Виконав:**

Студент групи ІІІ-11

Федоришин Микола Володимирович

**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 C++.
- Тема №2: Управління Виконанням Циклів.
- Тема №3: Вкладені Цикли.
- Тема №4: Основи Функцій у C++
- Тема №5: Перевантаження Функцій та Простір Імен.
- Тема №6: Розширені Можливості Функцій.
- Тема №7: Вбудовані Функції в C++

### 1) Індивідуальний план опрацювання теорії:

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

Запланований час на вивчення 1 година.  
Витрачений час 1 година.
  
- Тема №4 Основи функцій у C++:
  - 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](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 C++:
  - Джерела інформації:
    - Статті.

[https://www.youtube.com/watch?v=V\\_8XRRiUs7Y&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=49](https://www.youtube.com/watch?v=V_8XRRiUs7Y&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=49)
  - Що опрацьовано
    - Огляд вбудованих функцій у C++.

- 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

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

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

#### Завдання №2

##### VNS LAB 3 – TASK 1

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:

- для заданого  $n$ ;
- для заданої точності  $\varepsilon$  ( $\varepsilon=0.0001$ ).

Для порівняння знайти точне значення функції.

|    |                                          |                                 |    |                                                                          |
|----|------------------------------------------|---------------------------------|----|--------------------------------------------------------------------------|
| 14 | $y = \frac{1}{4}(x^2 - \frac{\pi^2}{3})$ | $\frac{\pi}{5} \leq x \leq \pi$ | 20 | $S = -\cos x + \frac{\cos 2x}{2^2} + \dots + (-1)^n \frac{\cos nx}{n^2}$ |
|----|------------------------------------------|---------------------------------|----|--------------------------------------------------------------------------|

#### Завдання №3

##### VNS LAB 7 – TASK 1

Написати функцію зі змінною кількістю параметрів для перетворення чисел із двійкової системи числення в трійкову. Написати викликаючу функцію `main`, що звертається до цієї функції не менше трьох разів з кількістю параметрів 3, 6, 7.

#### Завдання №4

##### VNS LAB 7 – TASK 2

Написати перевантажені функції й основну програму, що їх викликає.

- а) для обчислення натурального логарифму;
- б) для обчислення десяткового логарифму.

#### 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

Люди стоять у черзі до магазину. Чергу можна уявити як координатну пряму.  $i$ -а людина стоїть у координаті  $x_i$ .

Як відомо, у розпал пандемії COVID-19 необхідно дотримуватися соціальної дистанції.

Вам потрібно сказати, чи між кожною парою людей відстань є не меншою за 2 метри.

### Input

Перший рядок містить ціле число  $n$  — кількість людей у черзі.

Другий рядок містить  $n$  цілих чисел — координати людей у черзі в метрах, у порядку зліва направо.

### Output

В одному рядку виведіть **YES**, якщо люди дотримуються дистанції, і **NO** в іншому разі.

## 2. Дизайн та планувальна оцінка часу виконання завдань: Програма №1

- Плановий час на реалізацію 30 хвилин.

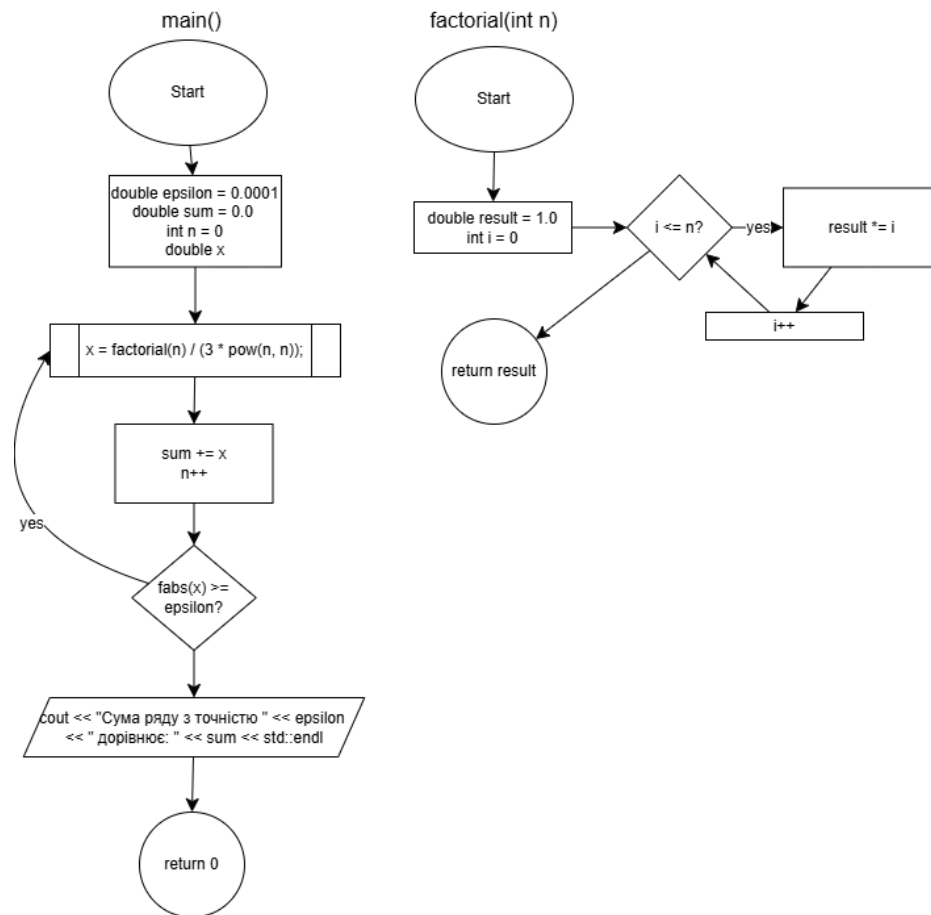


Рисунок 2.1. Блок-схема до програми 1

## Програма №2

- Плановий час на реалізацію 2.5 години.

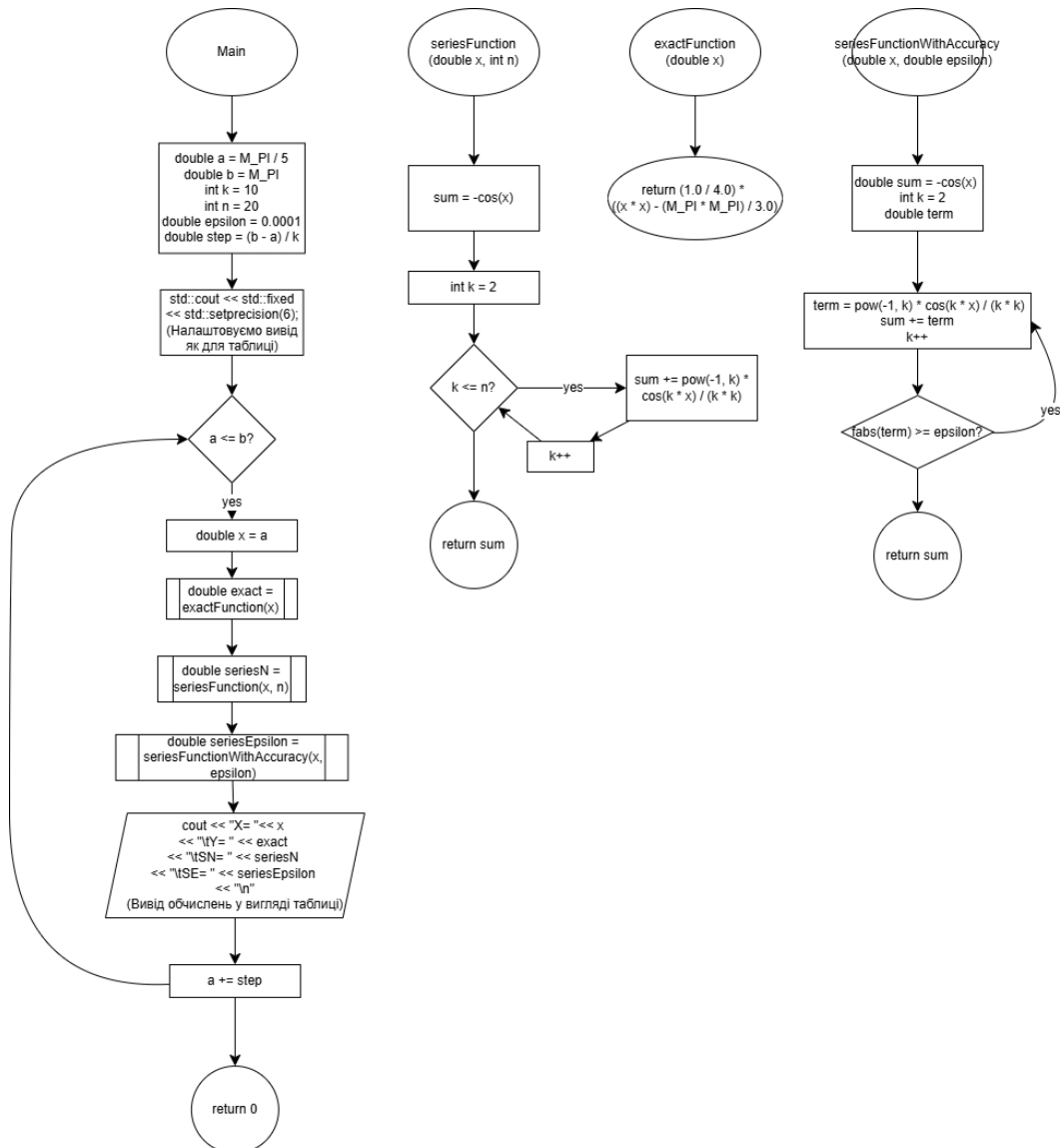


Рисунок 2.2. Блок-схема до програми 2

## Програма №3

- Плановий час на реалізацію 1.5 години.

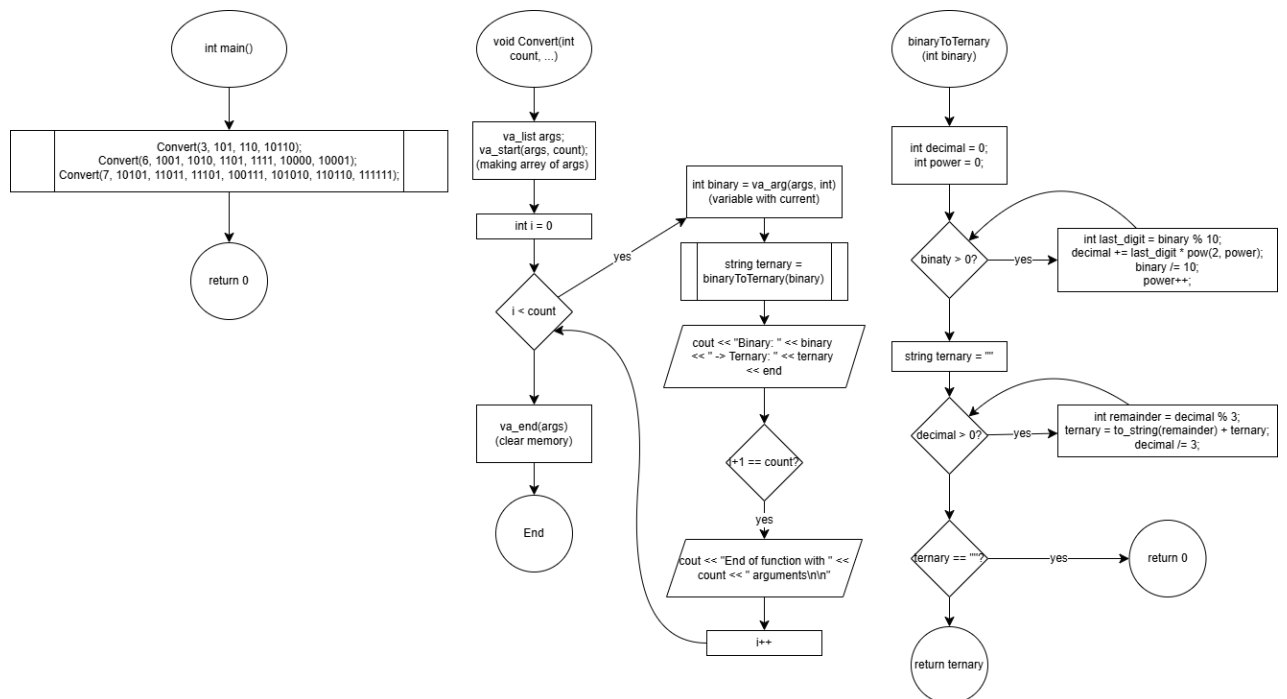


Рисунок 2.3. Блок-схема до програми 3

## Програма №4

- Плановий час на реалізацію 15 хв.

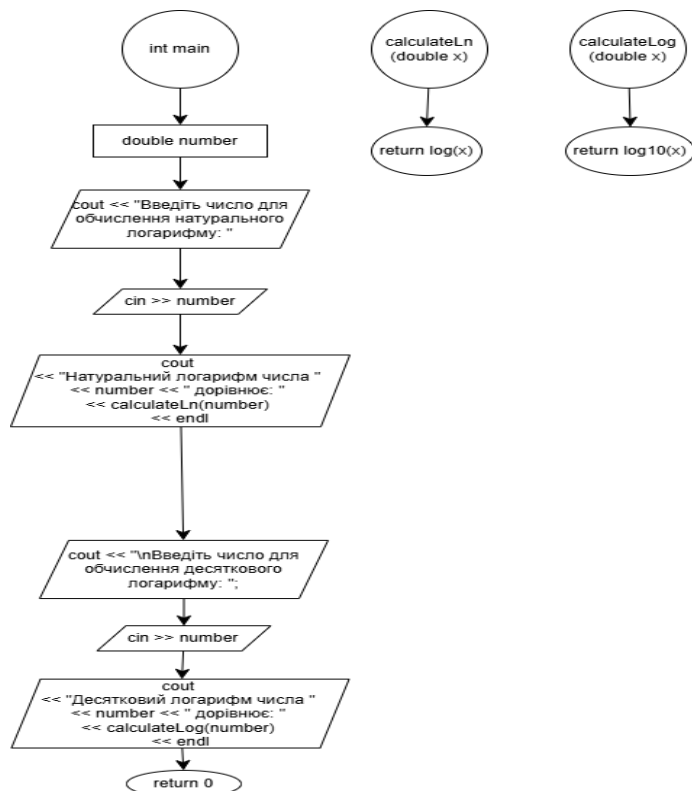


Рисунок 2.4. Блок-схема до програми 4



## Програма №5

- Плановий час на реалізацію 1 день

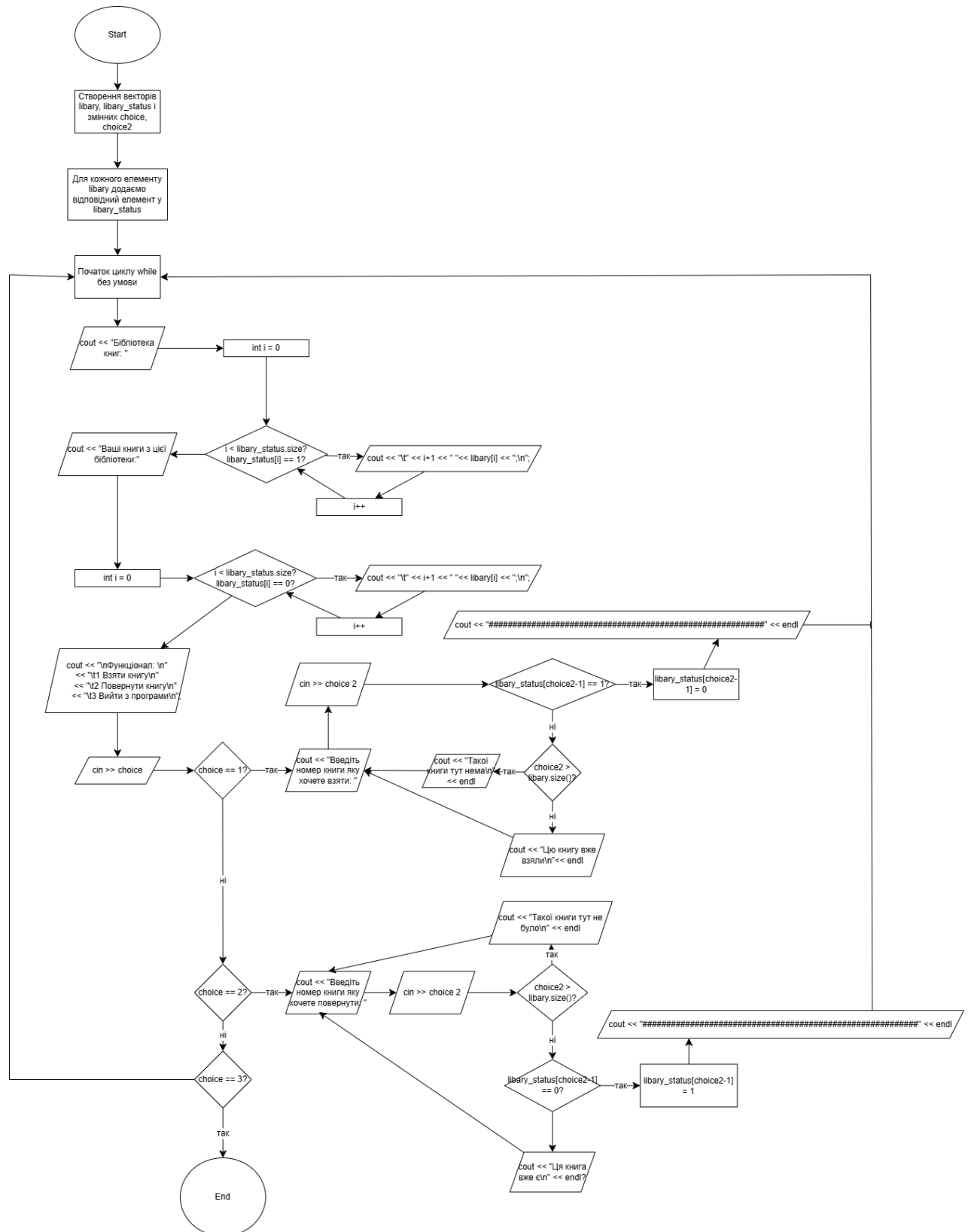


Рисунок 2.5. Блок-схема до програми №5

## Програма №6

- Плановий час на реалізацію 30 хв.

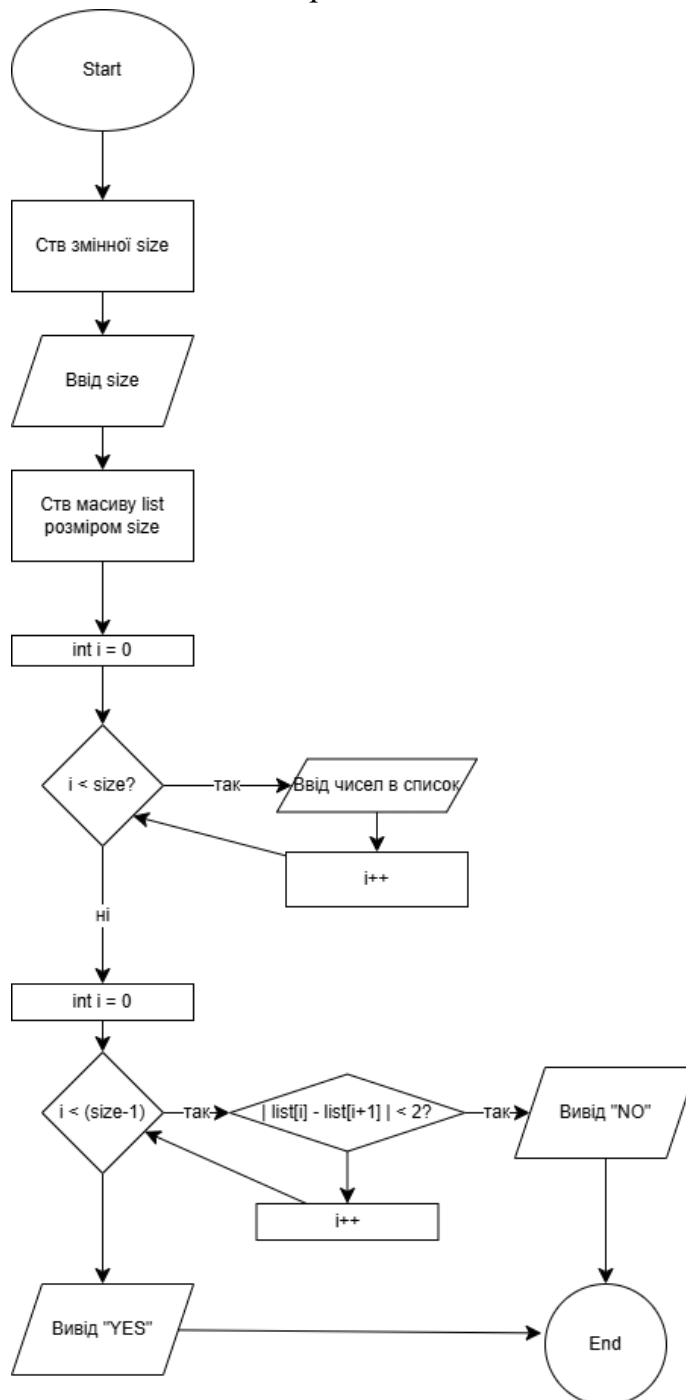


Рисунок 2.7. Блок-схема до програми №6

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

## Завдання №1

```
g+ vns_lab_2_task_1_variant_14_mykola_fedoryshyn.cpp U ●
g+ vns_lab_2_task_1_variant_14_mykola_fedoryshyn.cpp > main()
1  #include <iostream>
2  #include <cmath>
3  double factorial(int n) {
4      double result = 1.0;
5      for (int i = 1; i <= n; ++i) {
6          result *= i;
7      }
8      return result;
9  }
10
11 int main() {
12     double epsilon = 0.0001;
13     double sum = 0.0;
14     int n = 0;
15     double x;
16
17     do {
18         x = factorial(n) / (3 * pow(n, n));
19         sum += x;
20         n++;
21     } while (fabs(x) >= epsilon);
22
23     std::cout << "Сума ряду з точністю " << epsilon
24     << " дорівнює: " << sum << std::endl;
25
26     return 0;
27 }
28
```

Рисунок 3.1. Код до програми №1

```
Сума ряду з точністю 0.0001 дорівнює: 0.959922
```

Рисунок 3.2. Приклад виконання програми №1

Фактично затрачений час 25 хвилин.

**Посилання на файл у пулл реквест**

## Завдання №2

```

G+ vns_lab_3_task_1_variant_1_mykola_fedoryshyn.cpp > main()
1  #include <iostream>
2  #include <cmath>
3  #include <iomanip>
4
5  double exactFunction(double x) {
6      return (1.0 / 4.0) * ((x * x) - (M_PI * M_PI) / 3.0);
7  }
8  double seriesFunction(double x, int n) {
9      double sum = -cos(x);
10     for (int k = 2; k <= n; ++k) {
11         sum += pow(-1, k) * cos(k * x) / (k * k);
12     }
13     return sum;
14 }
15 double seriesFunctionWithAccuracy(double x, double epsilon) {
16     double sum = -cos(x);
17     int k = 2;
18     double term;
19     do {
20         term = pow(-1, k) * cos(k * x) / (k * k);
21         sum += term;
22         ++k;
23     } while (fabs(term) >= epsilon);
24     return sum;
25 }
26
27 int main() {
28     double a = M_PI / 5;
29     double b = M_PI;
30     int k = 10;
31     int n = 20;
32     double epsilon = 0.0001;
33     double step = (b - a) / k;
34     std::cout << std::fixed << std::setprecision(6);
35
36     for(; a<=b; a += step){
37         double x;
38         x = a;
39         double exact = exactFunction(x);
40         double seriesN = seriesFunction(x, n);
41         double seriesEpsilon = seriesFunctionWithAccuracy(x, epsilon);
42         std::cout << "X= " << x << "\tY= " << exact << "\tSN= " << seriesN << "\tSE= " << seriesEpsilon << "\n";
43     }
44     return 0;
45 }

```

Рисунок 3.3. Код до програми №2

|             |              |               |               |
|-------------|--------------|---------------|---------------|
| X= 0.628319 | Y= -0.723771 | SN= -0.722590 | SE= -0.723772 |
| X= 0.879646 | Y= -0.629023 | SN= -0.628103 | SE= -0.628794 |
| X= 1.130973 | Y= -0.502692 | SN= -0.503169 | SE= -0.502354 |
| X= 1.382301 | Y= -0.344778 | SN= -0.346309 | SE= -0.344373 |
| X= 1.633628 | Y= -0.155282 | SN= -0.156192 | SE= -0.154457 |
| X= 1.884956 | Y= 0.065797  | SN= 0.066869  | SE= 0.066053  |
| X= 2.136283 | Y= 0.318459  | SN= 0.320894  | SE= 0.317784  |
| X= 2.387610 | Y= 0.602704  | SN= 0.603857  | SE= 0.604439  |
| X= 2.638938 | Y= 0.918531  | SN= 0.915548  | SE= 0.920958  |
| X= 2.890265 | Y= 1.265941  | SN= 1.257435  | SE= 1.269705  |
| X= 3.141593 | Y= 1.644934  | SN= 1.596163  | SE= 1.635082  |

Рисунок 3.4. Приклад виконання програми №2

Фактично затрачений час 2 години.

**Посилання на файл у пул реквесті**

### Завдання №3

```
1  #include <iostream>
2  #include <string>
3  #include <cmath>
4  #include <cstdarg>
5  using namespace std;
6
7  string binaryToTernary(int binary) {
8      int decimal = 0;
9      int power = 0;
10
11     while (binary > 0) {
12         int last_digit = binary % 10;
13         decimal += last_digit * pow(2, power);
14         binary /= 10;
15         power++;
16     }
17
18     string ternary = "";
19     while (decimal > 0) {
20         int remainder = decimal % 3;
21         ternary = to_string(remainder) + ternary;
22         decimal /= 3;
23     }
24
25     if (ternary == "") {
26         return "0";
27     }
28
29     return ternary;
30 }
31
32
33 void Convert(int count, ...) {
34     va_list args;
35     va_start(args, count);
36
37     for (int i = 0; i < count; i++) {
38         int binary = va_arg(args, int);
39         string ternary = binaryToTernary(binary);
40         cout << "Binary: " << binary << " -> Ternary: " << ternary << endl;
41         if(i+1 == count){
42             cout << "End of function with " << count << " arguments\n\n";
43         }
44     }
45
46     va_end(args);
47 }
48
49
50 int main() {
51     Convert(3, 101, 110, 10110);
52     Convert(6, 1001, 1010, 1101, 1111, 10000, 10001);
53     Convert(7, 10101, 11011, 11101, 100111, 101010, 110110, 111111);
54
55     return 0;
56 }
57
```

Рисунок 3.5. Код до програми №3

```

Binary: 101 -> Ternary: 12
Binary: 110 -> Ternary: 20
Binary: 10110 -> Ternary: 211
End of function with 3 arguments

Binary: 1001 -> Ternary: 100
Binary: 1010 -> Ternary: 101
Binary: 1101 -> Ternary: 111
Binary: 1111 -> Ternary: 120
Binary: 10000 -> Ternary: 121
Binary: 10001 -> Ternary: 122
End of function with 6 arguments

Binary: 10101 -> Ternary: 210
Binary: 11011 -> Ternary: 1000
Binary: 11101 -> Ternary: 1002
Binary: 100111 -> Ternary: 1110
Binary: 101010 -> Ternary: 1120
Binary: 110110 -> Ternary: 2000
Binary: 111111 -> Ternary: 2100
End of function with 7 arguments

```

Рисунок 3.6. Приклад виконання програми №3

Фактично затрачений час 2 години.

**Посилання на файл у пулл реквесті**

**Завдання №4**

```

vns_lab_7_task_2_variant_14_mykola_fedoryshyn.cpp > main()
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5
6  double calculateLn(double x) {
7      return log(x);
8  }
9  double calculateLog(double x) {
10     return log10(x);
11 }
12
13 int main() {
14     double number;
15
16     cout << "Введіть число для обчислення натурального логарифму: ";
17     cin >> number;
18     cout << "Натуральний логарифм числа " << number << " дорівнює: " << calculateLn(number) << endl;
19
20     cout << "\nВведіть число для обчислення десяткового логарифму: ";
21     cin >> number;
22     cout << "Десятковий логарифм числа " << number << " дорівнює: " << calculateLog(number) << endl;
23
24     return 0;
25 }
26

```

Рисунок 3.7. Код до програми №4

```

Введіть число для обчислення натурального логарифму: 1
Натуральний логарифм числа 1 дорівнює: 0

Введіть число для обчислення десяткового логарифму: 10000
Десятковий логарифм числа 10000 дорівнює: 4

```

Рисунок 3.8. Приклад виконання програми №4

**Посилання на файл у пулл реквесті**

## Завдання №5

```
practice_work_task_1_mykola_fedoryshyn.cpp > main()
1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5  int main(){
6      vector<string> library = {"Переслідування Аделіни", "Інститут", "Не озирайся ні мовчи", "Ветролом"};
7      vector<bool> library_status = {};
8      int choice, choice2;
9      for(string k: library){
10         library_status.push_back(1);
11     }
12
13
14     while(1){
15         cout << "Бібліотека книг:\n";
16         for(int i = 0; i<library_status.size(); i++){
17             if(library_status[i] == 1){
18                 cout << "\t" << i+1 << " " << library[i] << "\n";
19             }
20         }
21         cout << "Ваші книги з цієї бібліотеки:\n";
22         for(int i = 0; i<library_status.size(); i++){ // for(bool k: library_status)
23             if(library_status[i] == 0){
24                 cout << "\t" << i+1 << " " << library[i] << "\n";
25             }
26         }
27         cout
28         << "\nФункціонал: \n"
29         << "\t1 Взяти книгу\n"
30         << "\t2 Повернути книгу\n"
31         << "\t3 Вийти з програми\n";
32         cin >> choice;
33
34         if (choice == 1){
35             do{
36                 cout << "Введіть номер книги яку хочете взяти: ";
37                 cin >> choice2;
38                 if (library_status[choice2-1] == 1){
39                     library_status[choice2-1] = 0;
40                     cout << "#####" << endl;
41                     break;
42                 }
43                 else if (choice2 > library.size()){
44                     cout << "Такої книги тут нема\n" << endl;
45                 }
46                 else{
47                     cout << "Цю книгу вже взяли\n" << endl;
48                 }
49             }while(1);
50         }
51
52         else if (choice == 2){
```

```

51
52     else if (choice == 2){
53         do{
54             cout << "Введіть номер книги яку хочете повернути: ";
55             cin >> choice2;
56
57             if (choice2 > library.size()){
58                 cout << "Такої книги тут не було\n" << endl;
59             }
60
61             else if (library_status[choice2-1] == 0){
62                 library_status[choice2-1] = 1;
63                 cout << "#####" << endl;
64                 break;
65             }
66             else{
67                 cout << "Ця книга вже є\n" << endl;
68             }
69         }while (1);
70     }
71 }
72 else if (choice == 3){
73     break;
74 }
75
76 }
77
78 }
79

```

Рисунок 3.9. Код до програми №5

```

Бібліотека книг:
1 Переслідування Аделіни;
2 Інститут;
3 Не озирайся і мовчи;
4 Ветролом;
Ваші книги з цієї бібліотеки:

Функціонал:
1 Взяти книгу
2 Повернути книгу
3 Вийти з програми

1
Введіть номер книги яку хочете взяти: 2
#####
Бібліотека книг:
1 Переслідування Аделіни;
3 Не озирайся і мовчи;
4 Ветролом;
Ваші книги з цієї бібліотеки:
2 Інститут;

Функціонал:
1 Взяти книгу
2 Повернути книгу
3 Вийти з програми

2
Введіть номер книги яку хочете повернути: 2
#####
Бібліотека книг:
1 Переслідування Аделіни;
2 Інститут;
3 Не озирайся і мовчи;
4 Ветролом;
Ваші книги з цієї бібліотеки:

Функціонал:
1 Взяти книгу
2 Повернути книгу
3 Вийти з програми

3

```

Рисунок 3.10. Приклад виконання програми №5

Фактично затрачений час 2 години.

**Посилання на файл у пул реквесті**



## Завдання №6

```
self_practice_work_algotester_task_1_mykola_fedoryshyn.cpp > main()
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4  int main(){
5      int size;
6      std::cin >> size;
7      int list[size];
8
9
10     for(int i = 0; i < size; i++){
11         std::cin >> list[i];
12     }
13
14
15     for(int i = 0; i < (size-1); i++){
16         if(abs(list[i]-list[i+1]) < 2){
17             cout << "NO";
18             return 0;
19         }
20     }
21     cout << "YES";
22
23 }
```

Рисунок 3.11. Код до програми №6

```
3
2
5
8
YES
```

Рисунок 3.12. Приклад виконання програми №6

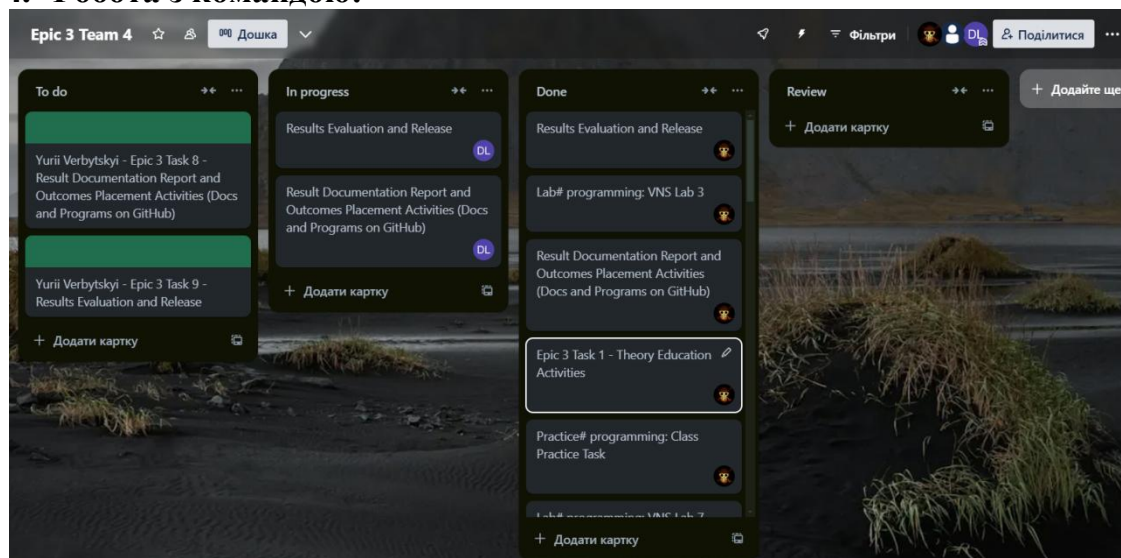
|                      |        |            |       |       |                          |
|----------------------|--------|------------|-------|-------|--------------------------|
| декілька секунд тому | C++ 20 | Зараховано | 0.007 | 1.043 | <a href="#">Перегляд</a> |
|----------------------|--------|------------|-------|-------|--------------------------|

Рисунок 3.13. Зарахування на Алготестері

Фактично затрачений час 10хв.

Посилання на файл у пулл реквесті

## 4. Робота з командою:



**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.