

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. Рекурсія. Вбудовані функції.»

з дисципліни: «Основи програмування»

до:

Практичних Робіт до блоку № 3

Виконав:

Студент групи ШІ-11

Яровой Павло Олегович

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:

1)Перелік тем:

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

Тема №2 -Управління Виконанням Циклів.

Тема №3 -Вкладені Цикли.

Тема №4 -Перевантаження Функцій та Простір Імен.

Тема №5 - Основи Функцій у C++

Тема №6 - Розширені Можливості Функцій.

Тема №7 - Вбудовані Функції в C++.

2)Індивідуальний план опрацювання теорії:

Тема №1

Джерела:

<http://cpp.dp.ua/operatory-tsyklu/>

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

Тема №2

Джерела:

https://rstudio-pubs-static.s3.amazonaws.com/378108_d61a0575f6d446ab962ff588e3e39c0d.html

Тема №3

Джерела: методичні відомості до лабораторної роботи №3

<https://studfile.net/preview/5252553/page:3/>

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

Тема №4

Джерела: <https://youtu.be/kYxNioENAIo?si=ErDsnPDsHfo1DycA>

<https://acode.com.ua/urok-108-perevantazhennya-funktsij/>

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

Тема №5

Джерела:

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

Тема №6

Джерела:

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

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

Тема №7

Джерела:

https://www.youtube.com/watch?v=V_8XRRlus7Y&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=49

Виконання роботи:

1)Перелік завдань:

- Epic 3 Task 1 - Theory Education Activities
- Epic 3 Task 2 - Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)
- Epic 3 Task 3 - Lab# programming: VNS Lab 2
- Epic 3 Task 4 - Lab# programming: VNS Lab 3
- Epic 3 Task 5 - Lab# programming: VNS Lab 7
- Epic 3 Task 6 - Practice# programming: Class Practice Task
- Epic 3 Task 7 - Practice# programming: Self Practice Task
- Epic 3 Task 8 - Result Documentation Report and Outcomes Placement Activities (Docs and Programs on GitHub)
- Epic 3 Task 9 - Results Evaluation and Release

2) Умови завдань:

Task 3:

20) Знайти суму ряду з точністю $\epsilon=0.0001$, загальний член якого

$$a_n = e^n \cdot 100^{-n^2}$$

Task 4:

$$20 \quad \left| \quad y = \left(\frac{x^2}{4} + \frac{x}{2} + 1 \right) e^{\frac{x}{2}} \quad \right| \quad 0,1 \leq x \leq 1 \quad \left| \quad 30 \quad \right| \quad S = 1 + 2 \frac{x}{2} + \dots + \frac{n^2 + 1}{n!} \left(\frac{x}{2} \right)^n$$

Task 5.1:

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

Написати функцію (або макровизначення), що визначає чи можна із чисел x , y , z побудувати трикутник. Написати функцію `triangle` зі змінною кількістю параметрів, що визначає скільки трійок рядом розташованих чисел типу `int` можуть бути довжинами сторін трикутника. Написати викликаючу функцію `main`, що звертається до функції `triangle` не менше трьох разів з кількістю параметрів 3, 9, 11.

Task 5.2:

Написати перевантажені функції й основну програму, що їх викликає.

- а) для масиву цілих чисел знаходить кількість парних елементів;
- б) для рядка знаходить кількість слів, що починаються на букву «а» .

3) Дизайн та планова оцінка часу виконання завдань:

Task 6:

Задача

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

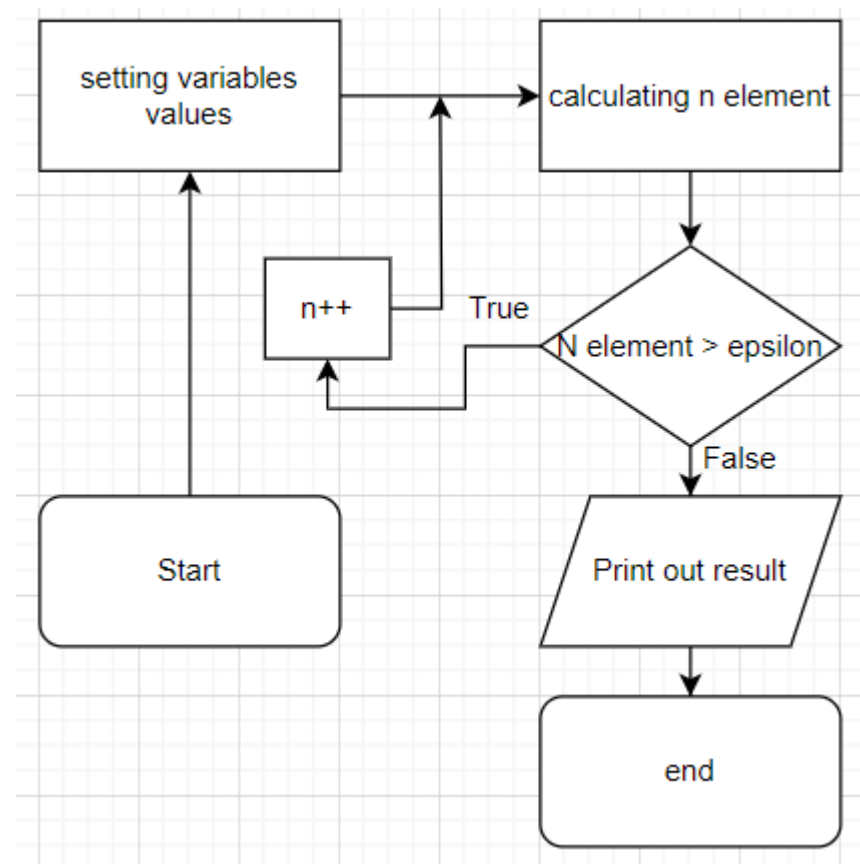
Програма повинна вміти

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

Структури даних

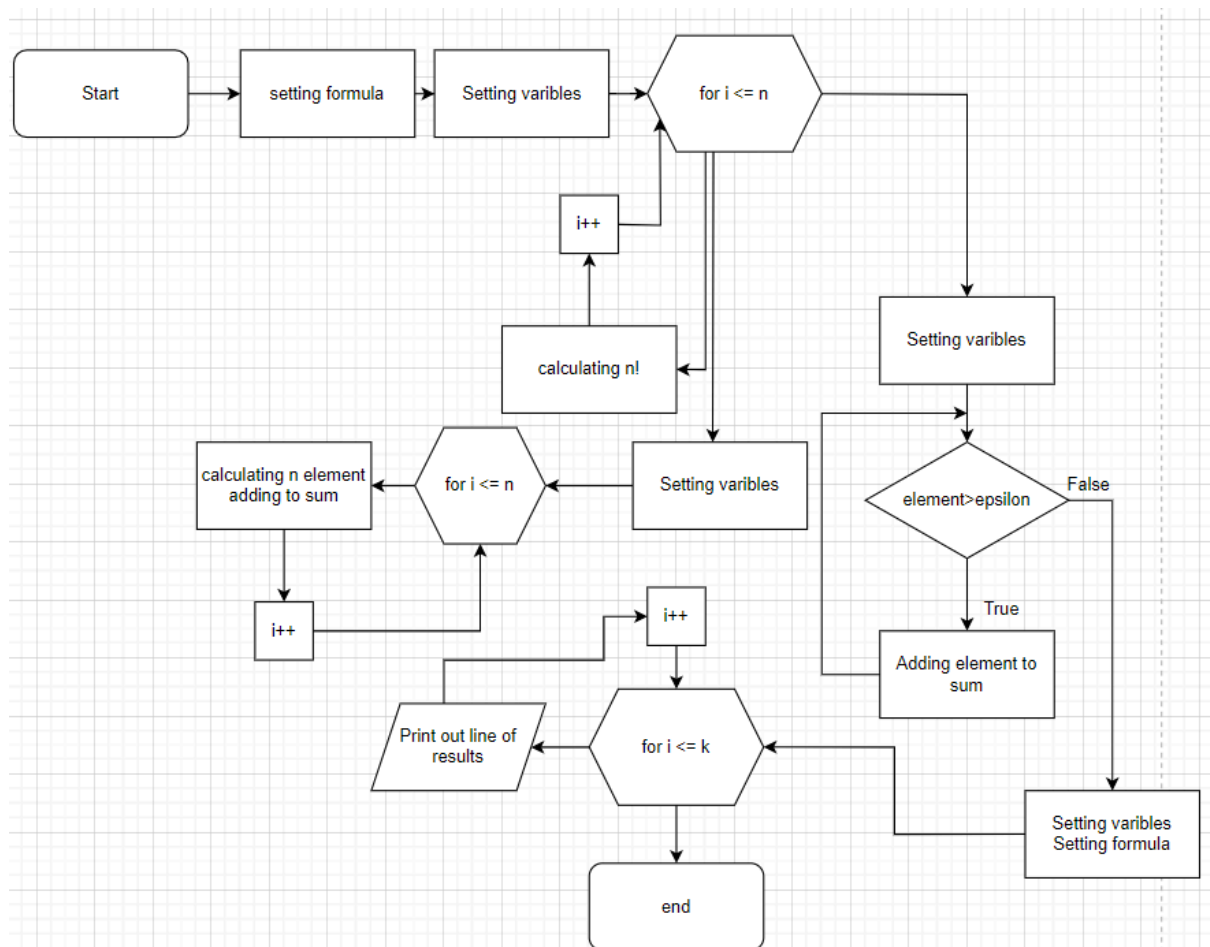
- Використовуйте масив або вектор для зберігання назв книг.
- Використовуйте інший масив або вектор для збереження стану доступності кожної книги.

Task 3 - Lab# programming: VNS Lab 2 Task 1 Варіант 20



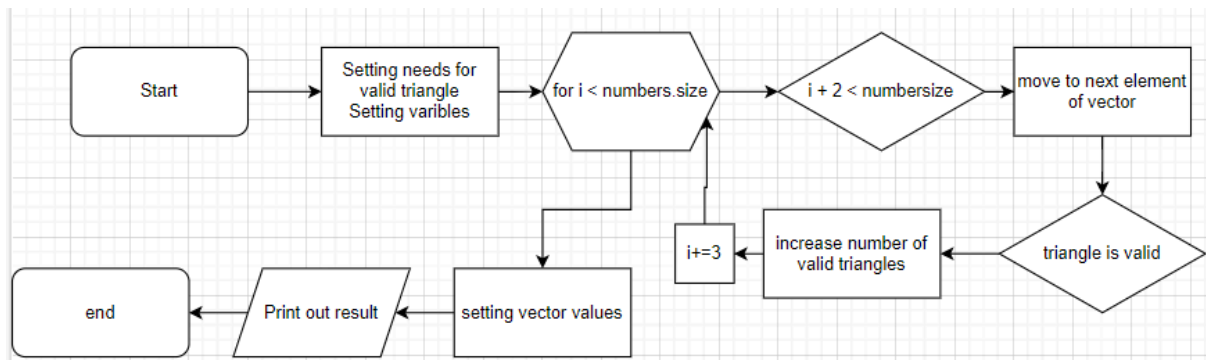
Орієнтований час виконання: 25 хв

Task 4 - Lab# programming: VNS Lab 3 Task 1 Варіант 20



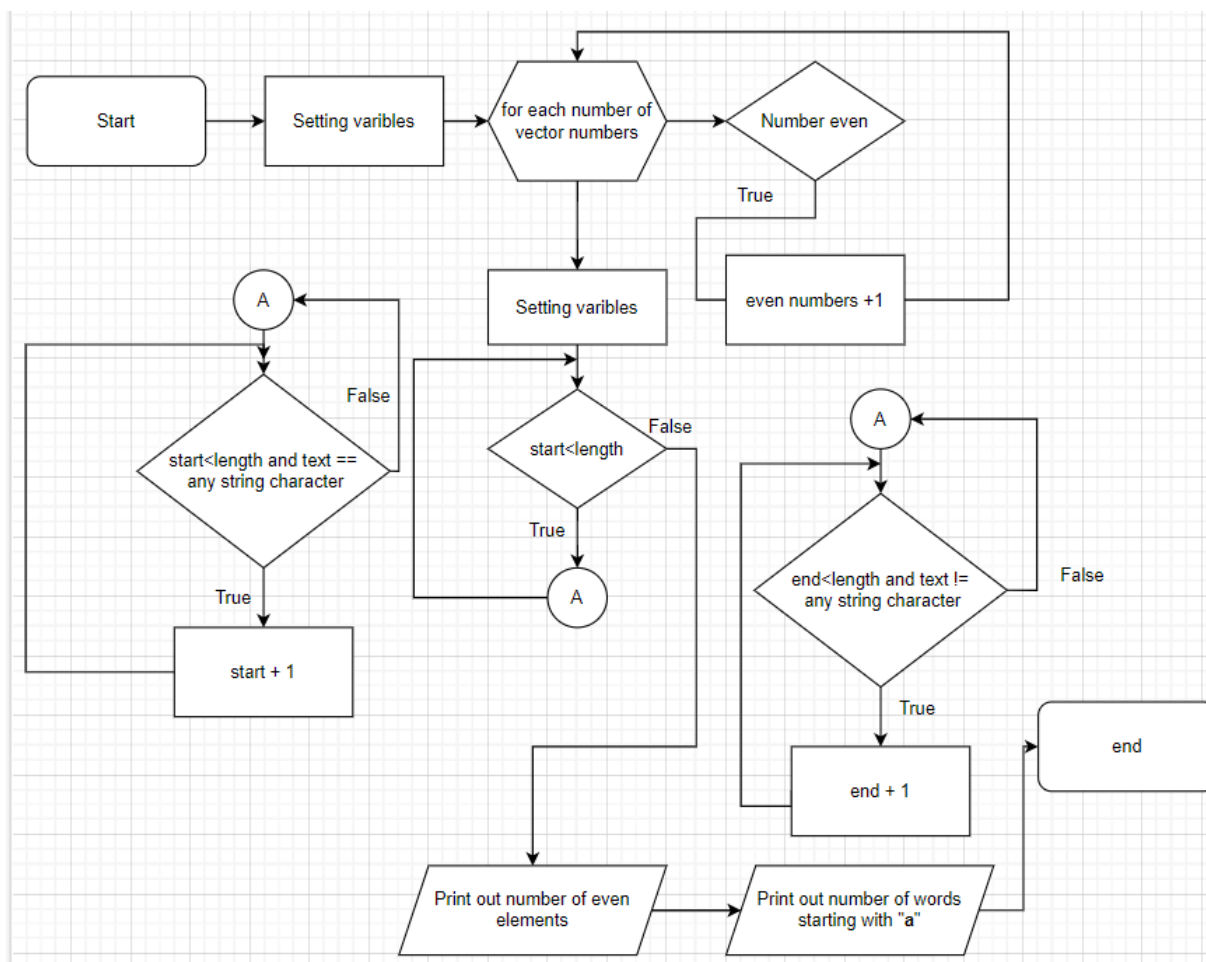
Орієнтовний час виконання: 10хв

Task 5.1 - Lab# programming: VNS Lab 7 Task 1 варіант 20



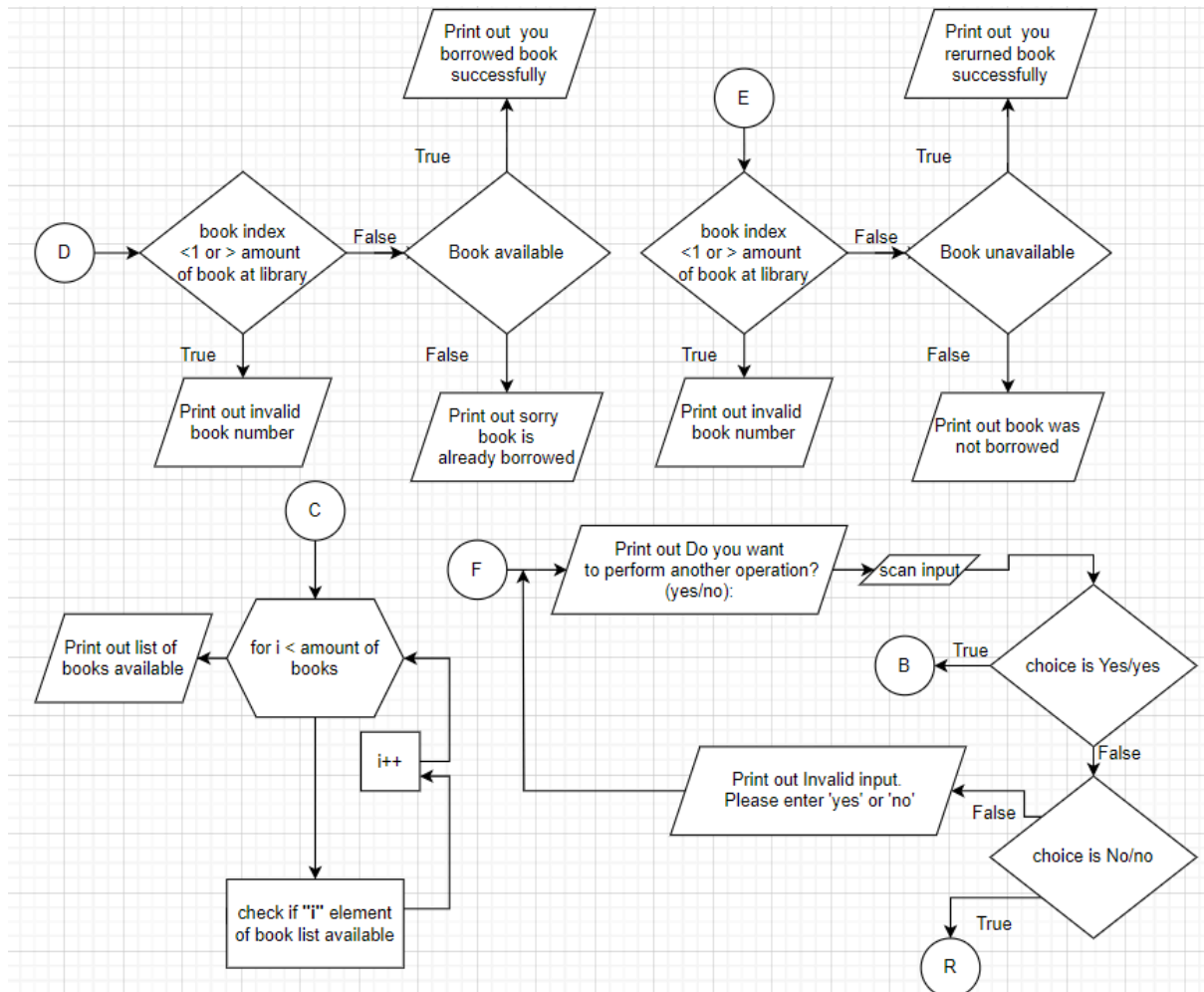
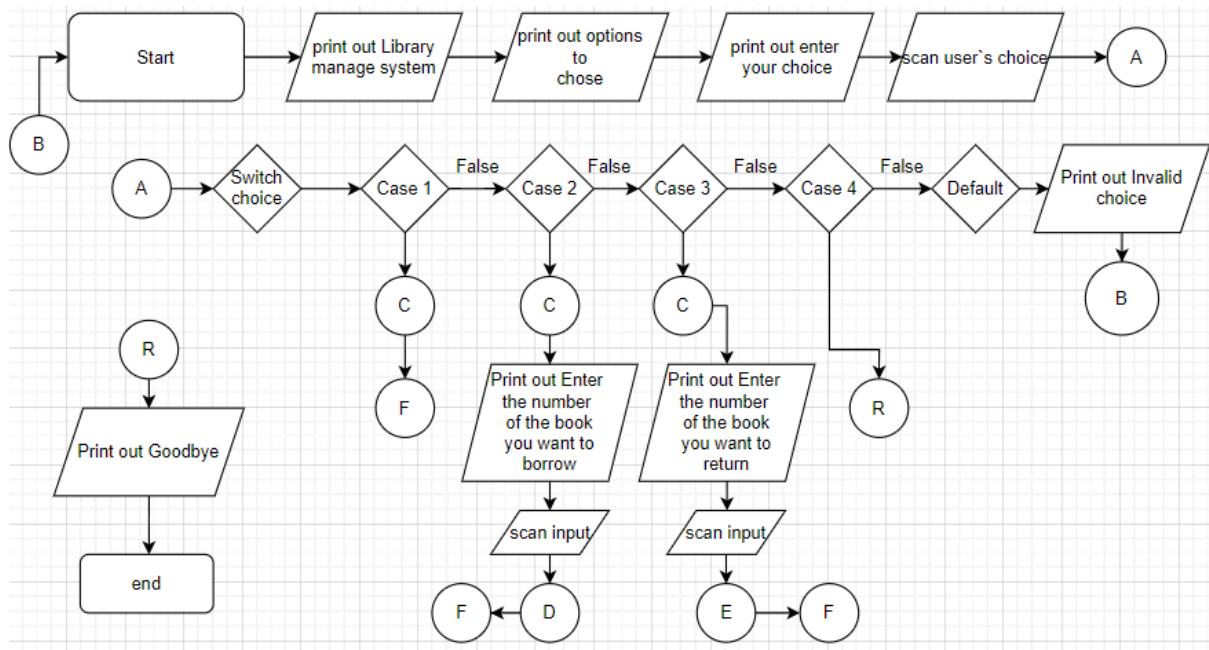
Орієнтований час виконання: 30 хв

Task 5.2 - Lab# programming: VNS Lab 7 Task 2 варіант 20



Орієнтований час виконання: 30 хв

Task 6 - Practice# programming: Class Practice Task



Орієнтований час виконання 1,5 год

4) Код програм з посиланням на зовнішні ресурси:

Task 3 - Lab# programming: VNS Lab 2 Task 1 Вариант 20

```
#include <iostream>
#include <cmath>

using namespace std;

int main() {
    const double epsilon = 0.0001;
    double sum = 0.0;
    double elementN;
    int n = 0;

    do {
        elementN = exp(n) * pow(100, -(n * n));
        sum += elementN;
        n++;
    } while (elementN > epsilon);

    cout << "\n Sum of n elements:" << sum << endl;
    return 0;
}
```

Task 4 - Lab# programming: VNS Lab 3 Task 1 Вариант 20

```

1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  double Function(double x) {
7      return ((x * x) / 4.0) + (x / 2.0) + 1 * exp(x / 2.0);
8  }
9
10 double factorial(int n) {
11     double fact = 1;
12     for (int i = 2; i <= n; ++i) {
13         fact *= i;
14     }
15     return fact;
16 }
17
18 double SeriesForN(double x, int n) {
19     double sum = 1.0;
20     double element = 1.0;
21     for (int i = 1; i <= n; ++i) {
22         element = ((x / 2.0) * element) / i;
23         sum += ((i * i + 1) / factorial(i)) * element;
24     }
25     return sum;
26 }
27
28 double SeriesForEpsilon(double x, double epsilon) {
29     double sum = 1.0;
30     double element = 1.0;
31     int i = 1;
32
33     while (abs((i * i + 1) * element / factorial(i)) >= epsilon) {
34         element = ((x / 2.0) * element) / i;
35         sum += ((i * i + 1) / factorial(i)) * element;
36         ++i;
37     }
38     return sum;
39 }
40
41 int main() {
42     double a = 0.1, b = 1.0;
43     int k = 10;
44     int n = 30;

```

```

45     const double epsilon = 0.0001;
46
47     double step = (b - a) / k;
48
49     cout << "Calculation results: " << endl;
50     for (int i = 0; i <= k; ++i) {
51         double x = a + i * step;
52
53         double just = Function(x);
54
55         double seriesN = SeriesForN(x, n);
56
57         double seriesEpsilon = SeriesForEpsilon(x, epsilon);
58
59         cout << "X=" << x << " SN=" << seriesN << " SE=" << seriesEpsilon << " Y=" << just << endl;
60     }
61
62     return 0;
63 }

```

Task 5.1 - Lab# programming: VNS Lab 7 Task 1 вариант 20

```
1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5
6  #define NeedsForTriangle(x, y, z) ((x + y > z) && (y + z > x) && (z + x > y))
7
8  bool ValidTriangle(int x, int y, int z) {
9      return NeedsForTriangle(x, y, z);
10 }
11
12 int triangle(const vector<int>& numbers) {
13     int validTriangles = 0;
14     for (size_t i = 0; i < numbers.size(); i += 3) {
15         if (i + 2 < numbers.size()) {
16             int x = numbers[i];
17             int y = numbers[i + 1];
18             int z = numbers[i + 2];
19             if (ValidTriangle(x, y, z)) {
20                 validTriangles++;
21             }
22         }
23     }
24     return validTriangles;
25 }
26
27 int main() {
28     vector<int> numbers1 = {3, 4, 5};
29     vector<int> numbers2 = {3, 4, 5, 5, 5, 5, 7, 8, 10};
30     vector<int> numbers3 = {3, 4, 5, 6, 7, 8, 10, 10, 10, 12, 14};
31
32     cout << "Number of possible triangles for 3 parameters: " << triangle(numbers1) << endl;
33     cout << "Number of possible triangles for 9 parameters: " << triangle(numbers2) << endl;
34     cout << "Number of possible triangles for 11 parameters: " << triangle(numbers3) << endl;
35
36     cout << "Can we form a triangle with (3, 4, 5): " << (ValidTriangle(3, 4, 5) ? "Yes" : "No") << endl;
37     cout << "Can we form a triangle with (5, 5, 5): " << (ValidTriangle(5, 5, 5) ? "Yes" : "No") << endl;
38
39     return 0;
40 }
```

Task 5.2 - Lab# programming: VNS Lab 7 Task 2 вариант 20

```
1  #include <iostream>
2  #include <vector>
3  #include <string>
4
5  using namespace std;
6
7  int countElements(const vector<int>& numbers) {
8      int count = 0;
9      for (int number : numbers) {
10         if (number % 2 == 0) {
11             count++;
12         }
13     }
14     return count;
15 }
16 int countElements(const string& text) {
17     int count = 0;
18     size_t start = 0;
19     size_t length = text.size();
20
21     while (start < length) {
22         while (start < length && text[start] == ' ') {
23             start++;
24         }
25         size_t end = start;
26         while (end < length && text[end] != ' ') {
27             end++;
28         }
29
30         if (end > start && (text[start] == 'a' || text[start] == 'A')) {
31             count++;
32         }
33
34         start = end + 1;
35     }
36
37     return count;
38 }
39 int main() {
40     vector<int> numbers = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
41     cout << "Number of even elements: " << countElements(numbers) << endl;
42     string text = "apple army marble avocado action mystery";
43     cout << "Number of words starting with 'a': " << countElements(text) << endl;
44     return 0;
45 }
```

Task 6 - Practice# programming: Class Practice Task

```
1  #include <iostream>
2  #include <vector>
3  #include <string>
4
5  using namespace std;
6
7  string displayBooks(const vector<string>& books, const vector<bool>& availability) {
8      string result = "\nList of Books:\n";
9      for (size_t i = 0; i < books.size(); ++i) {
10         result += to_string(i + 1) + ". " + books[i] + " - " +
11             (availability[i] ? "Available" : "Not Available") + "\n";
12     }
13     return result;
14 }
15
16 string borrowBook(vector<bool>& availability, int bookNumber) {
17     if (bookNumber < 1 || bookNumber > static_cast<int>(availability.size())) {
18         return "Invalid book number.\n";
19     }
20     if (availability[bookNumber - 1]) {
21         availability[bookNumber - 1] = false;
22         return "You borrowed the book successfully.\n";
23     }
24     return "Sorry, the book is already borrowed.\n";
25 }
26
27 string returnBook(vector<bool>& availability, int bookNumber) {
28     if (bookNumber < 1 || bookNumber > static_cast<int>(availability.size())) {
29         return "Invalid book number.\n";
30     }
31     if (!availability[bookNumber - 1]) {
32         availability[bookNumber - 1] = true;
33         return "You returned the book successfully.\n";
34     }
35     return "The book was not borrowed.\n";
36 }
37
38 int main() {
39     vector<string> books = {
40         "1984",
41         "The Great Gatsby",
42         "Neuromancer",
43         "Brave New World",
44         "Faust",
45         "The Master and Margarita"
```

```

46     };
47     vector<bool> availability(books.size(), true);
48
49     int choice;
50     string continueChoice;
51
52     start:
53     cout << "\nLibrary Management System\n";
54     cout << "1. List all books\n";
55     cout << "2. Borrow a book\n";
56     cout << "3. Return a book\n";
57     cout << "4. Exit\n";
58     cout << "Enter your choice: ";
59     cin >> choice;
60
61     switch (choice) {
62     case 1:
63         cout << displayBooks(books, availability);
64         break;
65     case 2: {
66         cout << displayBooks(books, availability);
67         cout << "\nEnter the number of the book you want to borrow: ";
68         int bookNumber;
69         cin >> bookNumber;
70         cout << borrowBook(availability, bookNumber);
71         break;
72     }
73     case 3: {
74         cout << displayBooks(books, availability);
75         cout << "\nEnter the number of the book you want to return: ";
76         int bookNumber;
77         cin >> bookNumber;
78         cout << returnBook(availability, bookNumber);
79         break;
80     }
81     case 4:
82         cout << "Exiting the system. Goodbye!\n";
83         return 0;
84     default:
85         cout << "Invalid choice. Please try again.\n";
86         goto start;
87     }
88

```



```

89     do {
90         cout << "\nDo you want to perform another operation? (yes/no): ";
91         cin >> continueChoice;
92
93         if (continueChoice == "yes" || continueChoice == "Yes") {
94             goto start;
95         } else if (continueChoice == "no" || continueChoice == "No") {
96             cout << "Exiting the system. Goodbye!\n";
97             return 0;
98         } else {
99             cout << "Invalid input. Please enter 'yes' or 'no'. \n";
100        }
101    } while (true);
102
103    return 0;
104 }

```

5) Результати виконання завдань та фактично затрачений час

Task 3 - Lab# programming: VNS Lab 2 Task 1 Варіант 20

```
Sum of n elements:1.02718
```

Фактично затрачений час: 15 хв

Task 4 - Lab# programming: VNS Lab 3 Task 1 Варіант 20

```

Calculation results:
X=0.1 SN=1.10316 SE=1.10316 Y=1.10646
X=0.19 SN=1.20152 SE=1.20152 Y=1.21405
X=0.28 SN=1.30527 SE=1.30527 Y=1.33386
X=0.37 SN=1.41457 SE=1.41457 Y=1.46699
X=0.46 SN=1.52959 SE=1.52959 Y=1.61466
X=0.55 SN=1.65048 SE=1.65048 Y=1.77814
X=0.64 SN=1.77742 SE=1.77741 Y=1.95883
X=0.73 SN=1.91057 SE=1.91057 Y=2.15821
X=0.82 SN=2.05012 SE=2.05012 Y=2.37791
X=0.91 SN=2.19625 SE=2.19625 Y=2.61964
X=1 SN=2.34912 SE=2.34912 Y=2.88526

```

Фактичний час затрачений на виконання: 20хв

Task 5.2 - Lab# programming: VNS Lab 7 Task 2 варіант 20

```
Number of possible triangles for 3 parameters: 1  
Number of possible triangles for 9 parameters: 3  
Number of possible triangles for 11 parameters: 3  
Can we form a triangle with (3, 4, 5): Yes  
Can we form a triangle with (5, 5, 5): Yes
```

Фактичний час затратчений на виконання: 30хв

Task 5.2 - Lab# programming: VNS Lab 7 Task 2 варіант 20

```
Number of even elements: 5  
Number of words starting with 'a': 4
```

Фактичний час затратчений на виконання: 30хв

Task 6 - Practice# programming: Class Practice Task

Library Management System

1. List all books
2. Borrow a book
3. Return a book
4. Exit

Enter your choice: 2

List of Books:

1. 1984 - Available
2. The Great Gatsby - Available
3. Neuromancer - Available
4. Brave New World - Available
5. Faust - Available
6. The Master and Margarita - Available

Enter the number of the book you want to borrow: 5
You borrowed the book successfully.

Do you want to perform another operation? (yes/no): yes

Library Management System

1. List all books
2. Borrow a book
3. Return a book
4. Exit

Enter your choice: 3

List of Books:

1. 1984 - Available
2. The Great Gatsby - Available
3. Neuromancer - Available
4. Brave New World - Available
5. Faust - Not Available
6. The Master and Margarita - Available

Enter the number of the book you want to return: 5
You returned the book successfully.

Do you want to perform another operation? (yes/no): yes

Library Management System

1. List all books
2. Borrow a book
3. Return a book
4. Exit

Enter your choice: 1

```
List of Books:
1. 1984 - Available
2. The Great Gatsby - Available
3. Neuromancer - Available
4. Brave New World - Available
5. Faust - Available
6. The Master and Margarita - Available

Do you want to perform another operation? (yes/no): no
Exiting the system. Goodbye!
```

Фактичний час затрачений на виконання: 2 год

6) Робота з комадою

Відео-зустріч:



Висновок: У межах практичних та лабораторних робіт блоку №3 я вивчив низку нових понять, таких як: функція, перевантажена функція, функція зі

змінною кількістю параметрів, цикли, вкладені цикли, простір імен, завершення виконання циклів, рекурсія.