

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:
Практичних Робіт до блоку № 1

Виконала:
Студентка групи ІІІ-12
Костак Олеся Михайлівна

Львів – 2024 р.

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

Мета роботи:

Вдосконалити вміння програмувати шляхом вивчення циклів і функцій. Засвоїти різні види циклів (for, while, do-while) та навчитися керувати їх виконанням за допомогою операторів break і continue.

Навчитися працювати з функціями: оголошення, передача параметрів, перевантаження та рекурсія. Розібратися з еліпсисом та просторами імен для ефективної організації коду. Ознайомитися з використанням вбудованих функцій і функцій зі змінною кількістю параметрів.

Теоретичні відомості:

1. Теоретичні відомості з переліком важливих тем:
 - 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++.

2. Індивідуальний план опрацювання теорії:
Джерела Інформації:

1. Сайт [geeksforgeeks](https://www.geeksforgeeks.com/)
 2. Сайт [programiz.com](https://www.programiz.com/)
 3. Сайт [w3schools.com](https://www.w3schools.com/)
 4. Ютуб-канал Блоган
 5. Лекції Пшеничного
 6. Сайт [cppreference.com](https://www.cppreference.com/)
- Тема №*.1: **Введення в Цикли та їх Види в C++.**
 - *Що опрацьовано:*
 - Відео [C++ • Теорія • Урок 31 • Поняття циклу](#)
 - Відео [C++ • Теорія • Урок 32 • while](#)
 - Відео [C++ • Теорія • Урок 105 • goto](#)
 - Стаття [C++ The foreach Loop](#)
 - *Статус:* Ознайомлена з основними видами циклів for, while, do-while та їх синтаксисом.
 - *Початок опрацювання теми:* 08.09

- *Звершення опрацювання теми: 10.09*

Тема №*.2: **Управління Виконанням Циклів.**

- *Що опрацьовано:*
 - Стаття [C++ Break and Continue](#)
- *Статус:* Вмію використовувати оператори break та continue, розумію їхню різницю.
- *Початок опрацювання теми: 15.10*
- *Звершення опрацювання теми: 15.10*

Тема №*.3: **Вкладені Цикли.**

- *Що опрацьовано:*
 - Відео [C++ • Теорія • Урок 38 • Вкладені конструкції](#)
 - Стаття [C++ Nested Loop](#)
- *Статус:* ознайомлена із вкладеними циклами. Вмію використовувати їх для створення n-вимірних масивів та інших завдань.
- *Початок опрацювання теми: 16.10*
- *Звершення опрацювання теми: 17.10*

Тема №*.4: **Основи Функцій у C++.**

- *Що опрацьовано:*
 - Стаття [C++ Functions](#)
 - Стаття [C++ Function Parameters](#)
 - Відео [C++ • Теорія • Урок 46 • Параметри за замовчуванням](#)
- *Статус:* Вмію визначати та оголошувати функції, передавати параметри за значенням і за посиланням. Розумію застосування return.
- *Початок опрацювання теми: 12.10*
- *Звершення опрацювання теми: 15.10*

Тема №*.5: **Перевантаження Функцій та Простір Імен.**

- *Що опрацьовано:*
 - Стаття [C++ Function Overloading](#)
 - Відео [C++ • Теорія • Урок 48 • Переваантаження функцій](#)
 - Відео [C++ • Теорія • Урок 106 • namespace](#)
 - Стаття [Namespace in C++ | Set 1 \(Introduction\)](#)
- *Статус:* Ознайомлена з поняттям перевантаження функції та простору імен. Вмію створювати простір імен.
- *Початок опрацювання теми: 20.10*
- *Звершення опрацювання теми: 24.10*

Тема №*.6: Розширені Можливості Функцій.

- *Що опрацьовано:*
 - Стаття [Variadic function templates in C++](#)
 - Стаття [Variadic functions](#)
 - Відео [C++ • Теорія • Урок 44 • Область видимості, глобальні/локальні дані, static/extern](#)
 - Відео [C++ • Теорія • Урок 54 • Рекурсія](#)
- *Статус:* Ознайомлена з використанням функцій зі змінною кількістю параметрів, з областю видимості функції static та extern. Вмію використовувати рекурсію та передавати масиви як параметри функції.
- *Початок опрацювання теми:* 14.10
- *Звершення опрацювання теми:* 20.10

Тема №*.7: Вбудовані Функції в C++.

- *Що опрацьовано:*
 - Стаття [Inline Functions in C++](#)
- *Статус:* Ознайомлена з поняттям вбудованої функції. Розумію її переваги та недоліки і де її слід використовувати, а де – ні.
- *Початок опрацювання теми:* 05.09
- *Звершення опрацювання теми:* 06.06

Виконання роботи:

1. Опрацювання завдання та вимог до програм та середовища:

Завдання №1 VNS Lab 2 - Task 1

- *Варіант завдання:* 5
- *Деталі завдання:* Використовуючи оператор циклу, знайти суму елементів, зазначених у конкретному варіанті. Результат надрукувати, надавши відповідним заголовком.

Завдання №2 VNS Lab 3 - Task 1

- *Варіант завдання:* 5
- *Деталі завдання:* 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 ($\epsilon=0.0001$).Для порівняння знайти точне значення функції.

Завдання №3 VNS Lab 7 – Task-1

- *Варіант завдання:* 5
- *Деталі завдання:* Написати функцію sum зі змінною кількістю параметрів, що знаходить суму чисел типу int за формулою:
$$S=a_1*a_2-a_2*a_3+a_3*a_4-\dots$$

Завдання №4 VNS Lab 7 – Task-2

- *Варіант завдання:* 5
- *Деталі завдання:* Написати перевантажені функції й основну програму, що їх викликає.
 - а) для множення дійсних чисел;
 - б) для множення комплексних чисел.

Завдання №5 Class Practice Work

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

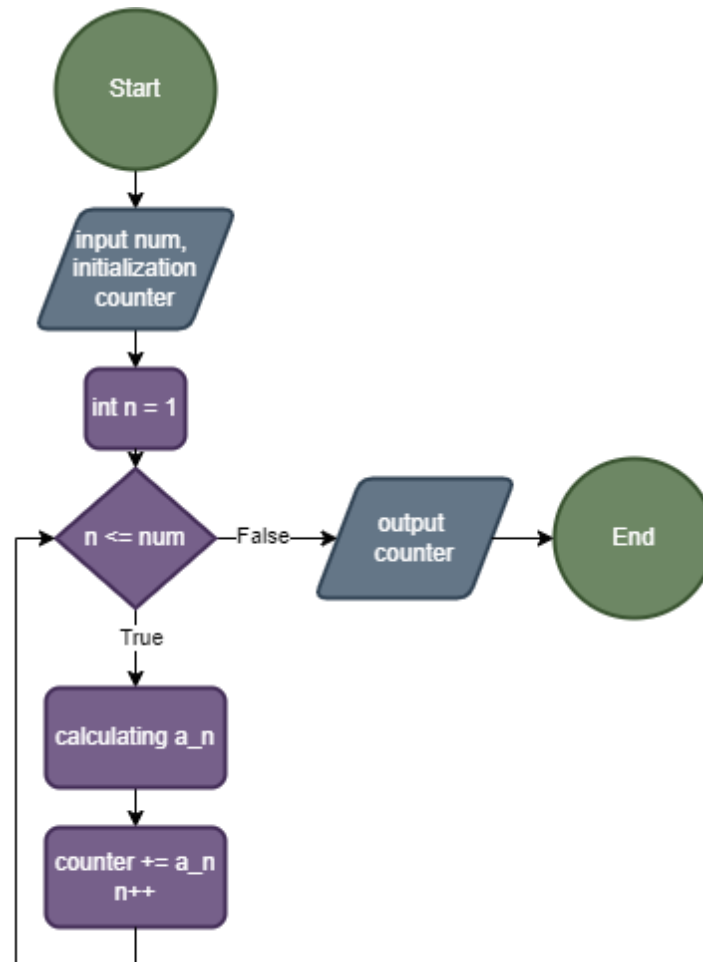
Завдання №6 Self Practice Work ([Шифрування корупційних схем](#))

- *Деталі завдання:* в офісі Зеника працює один татарин. Щомісяця він придумує нову геніальну корупційну схему й охоче ділиться нею із Зеником у повідомленні. Цього місяця знову не обійшлося без нової хитрої схеми. Однак Татарин — не дурний. Він знає, що повідомлення зі схемою для Зеника можуть перехопити правоохоронці. Тому він шифрує його алгоритмом кодування довжин серій.

2. Дизайн та планована оцінка часу виконання завдань:

Програма №1 VNS Lab 2 - Task 1 - 5

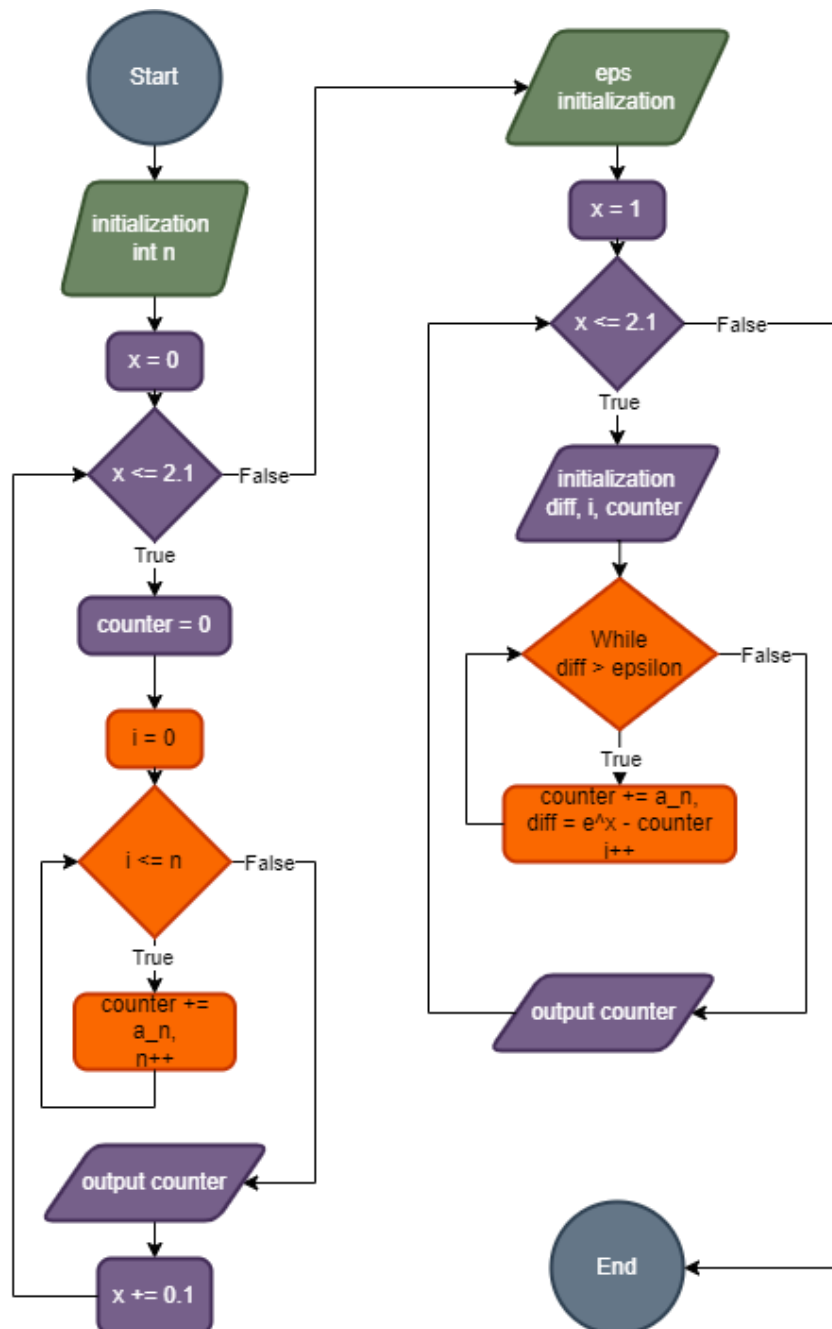
Блок-схема:



- Планований час на реалізацію: 10хв

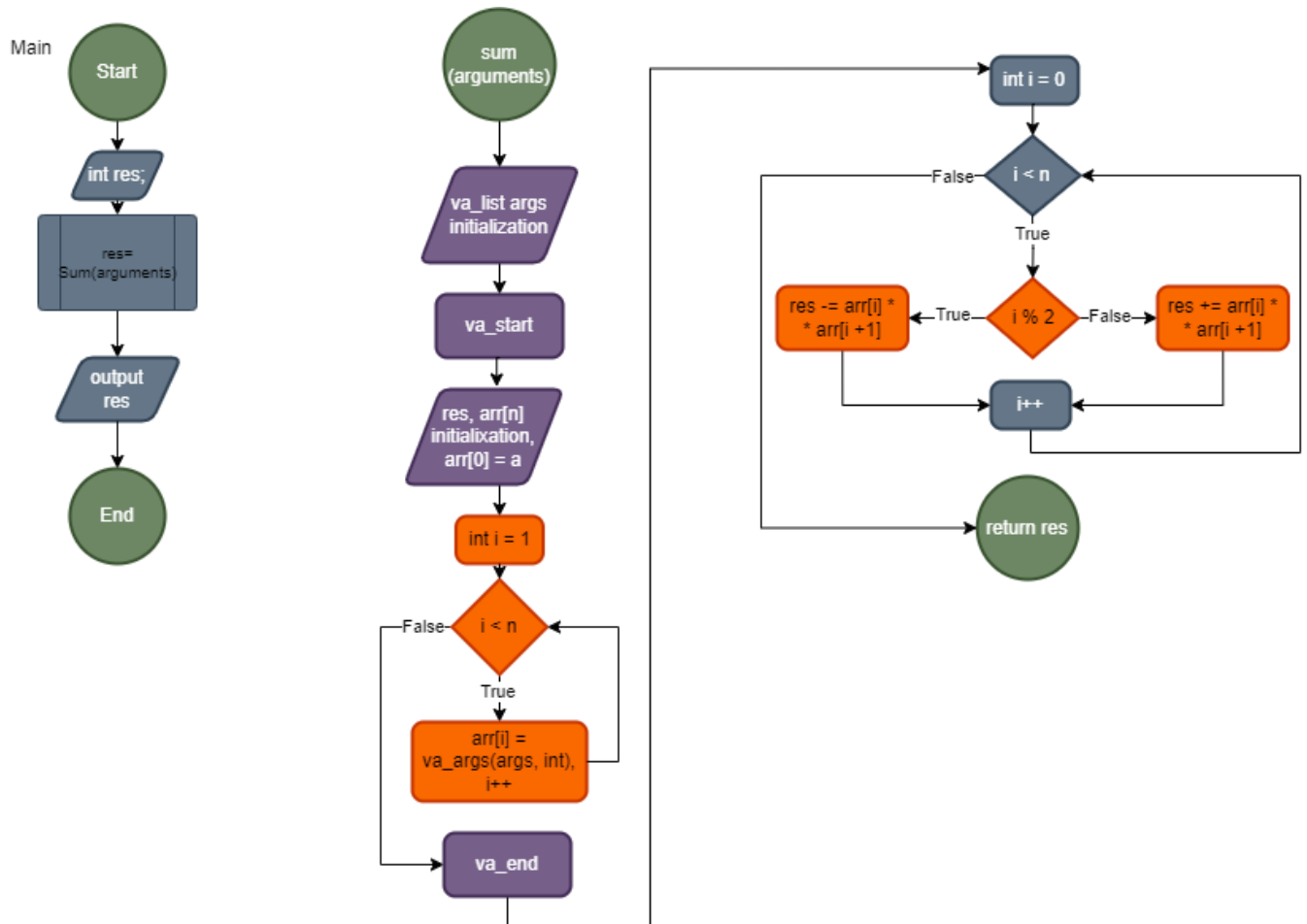
Програма №2 VNS Lab 3 - Task 1

- Блок-схема:



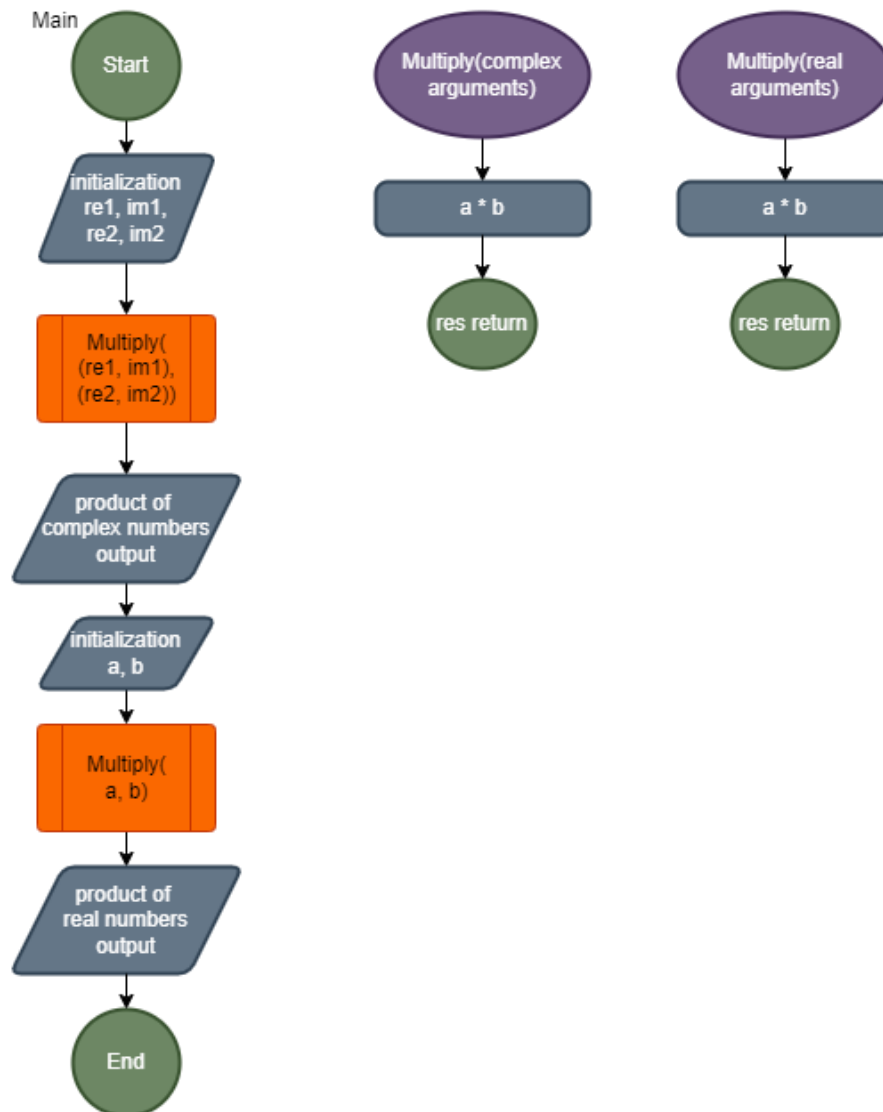
- Планований час на реалізацію: 50 хв

- Планований час на реалізацію: 5хв
- Завдання №3 VNS Lab 7 – Task-1
- Блок-схема:



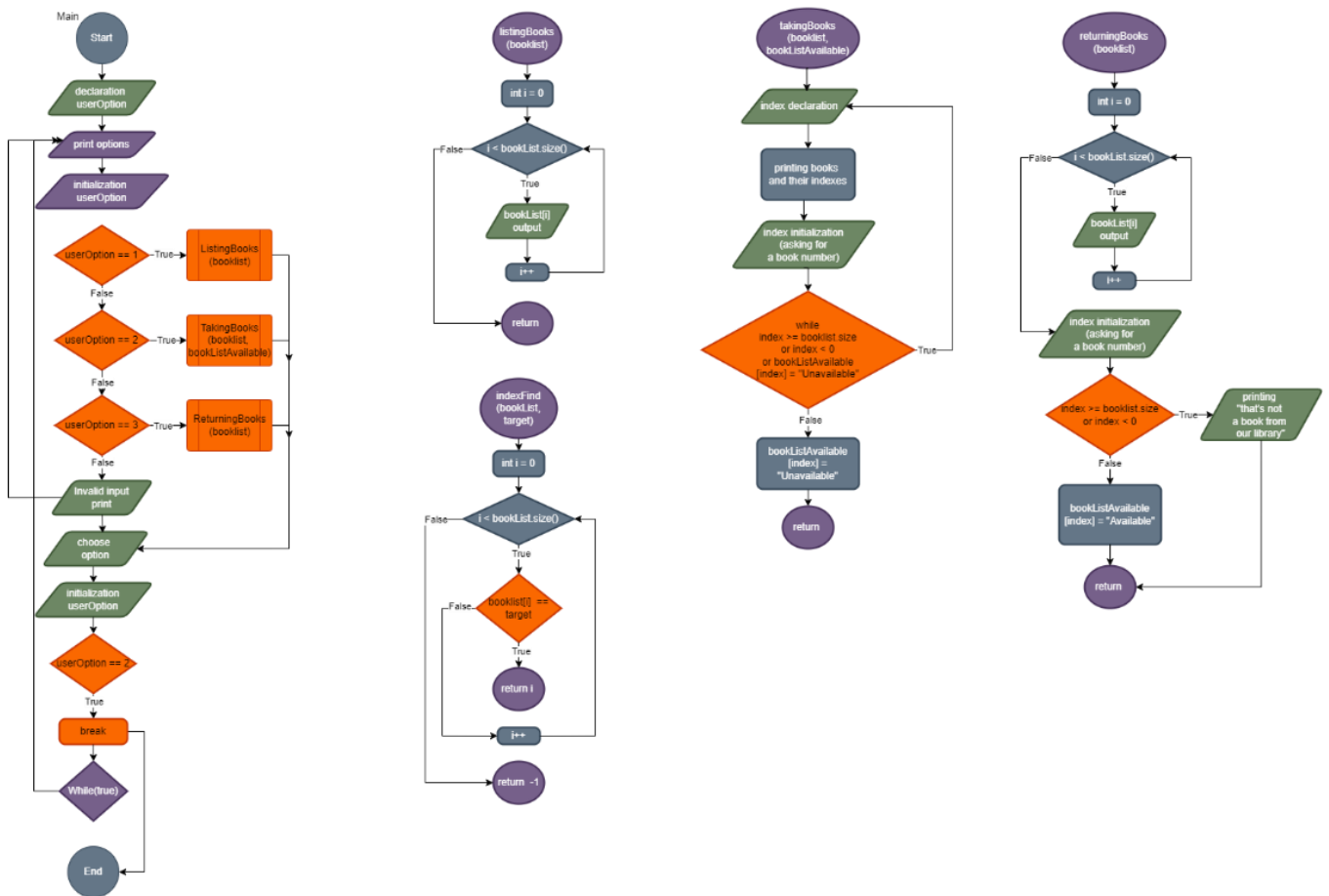
- Планований час на реалізацію: 30хв

- Завдання №4 VNS Lab 7 – Task-2
- Блок-схема:



- Планований час на реалізацію: 30хв

- Завдання №5 Class Practice Work
- Блок-схема:



- Планований час на реалізацію: 3 год

№6 Self Practice Work ([Шифрування корупційних схем](#))

- Блок-схема: -
- Планований час на реалізацію: 20 хв

3. Код програм з посиланням на зовнішні ресурси:

Завдання №1 VNS Lab 2 - Task 1

[Посилання](#) на файл програми у пул-запиті GitHub

```

vns_lab_2_task_john_black.cpp > main()
1  #include <stdio.h>
2  #include <cmath>
3
4  int main()
5  {
6      //float epc = 0.0001;
7      int num;
8      printf("Input num to calculate series num");
9      scanf("%d", &num);
10     float counter = 0;
11
12     for(int n = 1; n <= num; n++)
13     {
14         float a_n = pow(-1, n - 1) / pow(n, n);
15         counter += a_n;
16     }
17     printf("Final result of calculating: %.4f", counter);
18
19     return 0;
20
21 }

```

Завдання №2 VNS Lab 3 - Task

[Посилання](#) на файл програми у пул-запиті GitHub

```

vns_lab_3_task_olesia_kostak.cpp > main()
1  #include <iostream>
2  #include <cmath>
3
4
5  long long factorial(int num);
6  int main()
7  {
8      // a) for given n
9      int n = 15;
10     for (float x = 1; x <= 2.1; x += 0.1)
11     {
12         long double counter = 0;
13         for (int i = 0; i <= n; i++)
14             counter += (pow(x, i) / factorial(i));
15         std::cout << "e in the " << x << " power: " << counter << std::endl;
16     }
17
18     std::cout << std::endl << " " << std::endl;
19
20     // b) for given epsilon
21     float eps = 0.0001;
22     for (float x = 1; x <= 2.1; x += 0.1)
23     {
24         double diff = pow(M_E, x);
25         int i = 0;
26         double counter = 0;
27         while (diff > eps)
28         {
29             counter += pow(x, i) / factorial(i);
30             diff = fabs(pow(M_E, x) - counter);
31             i++;
32         }
33         std::cout << "e in the " << x << " power: " << counter << std::endl;
34     }
35
36     return 0;
37 }
38
39
40 long long factorial(int num)
41 {
42     if (num > 0)
43         return num * factorial(num - 1);
44     else
45         return 1;
46 }

```

Завдання №3 VNS Lab 7 – Task-1

[Посилання](#) на файл програми у пул-запиті GitHub

```
100, 4 days ago | 1 author (100)
1  #include <iostream>
2  #include <cstdarg>
3
4  int sum(int n, int a, ...);
5
6  int main()
7  {
8      std::cout << sum(5, 3, 100, 10, 90, 3) << std::endl;
9      std::cout << sum(10, 8, 1, 1, 9, 69, 4, 10, 13, 3, 7) << std::endl;
10     std::cout << sum(12, 5, 5, 50, 9, 6, 4, 70, 1, 6, 7, 3, 1) << std::endl;
11     You, 4 days ago • First commit
12     return 0;
13 }
14
15 int sum(int n, int a, ...)
16 {
17     va_list args;
18     va_start (args, a);
19     int res = 0;
20     int arr[n];
21
22     arr[0] = a;
23     for(int i = 1; i < n; i++)
24         arr[i] = va_arg(args, int);
25
26     va_end (args);
27
28     for(int i = 0; i < n - 1; i++)
29     {
30         if(i % 2)
31             res -= arr[i] * arr[i+1];
32         else
33             res += arr[i] * arr[i + 1];
34     }
35
36     return res;
37 }
```

Завдання №4 VNS Lab 7 – Task-2

[Посилання](#) на файл програми у пул-запиті GitHub

```
1  v #include <iostream>          You, 4 days ago • First commit
2  #include <complex>
3
4  v long double Multiply(long double a, long double b)
5  {
6      return a * b;
7  }
8
9  v std::complex<double> Multiply(std::complex<double> a, std::complex<double> b)
10 {
11     return a * b;
12 }
13
14
15 v int main()
16 {
17     std::complex<double> num1, num2;
18     double re1, im1, re2, im2;
19     std::cout << "Enter two complex numbers:" << std::endl;
20     std::cin >> re1 >> im1 >> re2 >> im2;
21     num1 = std::complex<double>(re1, im1);
22     num2 = std::complex<double>(re2, im2);
23     std::cout << "Multiple: " << Multiply(num1, num2) << std::endl;
24
25     long double a, b;
26     std::cout << "Enter two real numbers:" << std::endl;
27     std::cin >> a >> b;
28     std::cout << "Multiple: " << Multiply(a, b) << std::endl;
29
30     return 0;
31 }
```

Завдання №5 Class Practice Work

[Посилання](#) на файл програми у пул-запиті GitHub

```

1 #include <iostream>
2 #include <vector>
3 #include <string>
4 #include <limits>
5
6
7 extern std::vector<std::string> booklist;
8 extern std::vector<std::string> booklistAvailable;
9 int indexFind(std::vector<std::string> booklist, std::string target);
10 void takingBooks (std::vector<std::string>& booklist, std::vector<std::string>& booklistAvailable);
11 void listingBooks(std::vector<std::string> booklist);
12 void returningBooks(std::vector<std::string>& booklist);
13
14
15 int main()
16 {
17     std::cout << "Congratulations! You're in online library." << std::endl;
18     std::string userOption;
19     do
20     {
21         point:
22         std::cout << "Available options:" << std::endl <<
23         "1 - List all books" << std::endl <<
24         "2 - Take a book (if available)" << std::endl <<
25         "3 - Return a book" << std::endl <<
26         "Enter a Number of the option: " << std::endl;
27         std::cin >> userOption;
28
29         if (userOption == "1")
30             listingBooks(booklist);
31
32         else if (userOption == "2")
33             takingBooks(booklist, booklistAvailable);
34
35         else if (userOption == "3")
36             returningBooks(booklist);
37         else
38         {
39             std::cout << "You have enter not a correct option. You'll be moved to the main menu." << std::endl;
40             goto point;
41         }
42
43         std::cout << "Choose option:" << std::endl <<
44         "1. Return to the main menu." << std::endl <<
45         "2. End session" << std::endl;
46         std::cin >> userOption;
47         if (userOption == "2")
48             break;
49     }
50     while(true);
51
52     return 0;

```

```

53
54
55
56
57 std::vector<std::string> booklist = {
58     "Dune",
59     "Neuromancer",
60     "Snow Crash",
61     "The Hitchhiker's Guide to the Galaxy",
62     "Brave New World",
63     "The War of the Worlds",
64     "Fahrenheit 451",
65     "The Left Hand of Darkness",
66     "The Expanse: Leviathan Wakes",
67     "The Three-Body Problem",
68     "Do Androids Dream of Electric Sheep?",
69     "Altered Carbon",
70     "The Man in the High Castle",
71     "The Time Machine",
72     "Ringworld",
73     "The Moon is a Harsh Mistress",
74     "The Forever War"
75 };
76 std::vector<std::string> booklistAvailable (booklist.size(), "Available");
77
78 void listingBooks(std::vector<std::string> booklist)
79 {
80     std::cout << "All the books in our library:" << std::endl;
81     std::cout << "-----" << std::endl;
82     for(int i = 0; i < booklist.size(); i++)
83         std::cout << ". " << booklist[i] << std::endl;
84     std::cout << "-----" << std::endl;
85
86     return;
87 }
88
89 void takingBooks (std::vector<std::string>& booklist, std::vector<std::string>& booklistAvailable)
90 {
91     int index;
92     std::cout << "All books status: " << std::endl;
93     for (std::string book: booklist)
94     {
95         index = indexFind(booklist, book);
96         std::cout << index << ". " << book << std::endl << "Status: " << booklistAvailable[index] << std::endl;
97     }
98     do
99     {
100         std::cout << "What book would you like to take. Enter the number of it: ";
101         std::cin >> index;
102         if(index >= booklist.size() || index < 0)
103             std::cout << "You have entered not a correct number. Please try again" << std::endl;

```

```

104 | else if(bookListAvailable[index] == "Unavailable") You, 4 days ago • First commit
105 |     std::cout << "This book is unavailable. Please choose another option." << std::endl;
106 | }
107 while(index >= bookList.size() || index < 0 || bookListAvailable[index] == "Unavailable");
108 bookListAvailable[index] = "Unavailable";
109 std::cout << "You have taken a book. Good reading!" << std::endl;
110 return;
111 }
112 }
113
114 int indexFind(std::vector<std::string> bookList, std::string target)
115 {
116     for (size_t i = 0; i < bookList.size(); i++)
117         if (bookList[i] == target)
118             return i;
119     return -1;
120 }
121
122 void returningBooks(std::vector<std::string>& bookList)
123 {
124     for (int i = 0; i < bookList.size(); i++)
125         std::cout << i << ". " << bookList[i] << std::endl;
126     std::cout << "What book would you like to return? Enter the index" << std::endl;
127     int index;
128     std::cin >> index;
129     if(index >= bookList.size() || index < 0)
130         std::cout << "That's not a book from our library" << std::endl;
131     else
132     {
133         bookListAvailable[index] = "Available";
134         std::cout << "You have successfully returned a book" << std::endl;
135     }
136     return;
137 }

```

Завдання №6 Self Practice Work ([Шифрування корупційних схем](#))

[Посилання](#) на файл програми у пул-запиті GitHub

```

1  #include <iostream>
2  #include <string>
3
4
5  int main()
6  {
7      std::string str;
8      std::cin >> str;
9      int counter = 1;
10     for (int i = 0; i < str.length(); i++)
11     {
12         if (str[i] == str[i + 1])
13             counter++;
14         else
15         {
16             std::cout << counter << str[i];
17             counter = 1;
18         }
19     }
20     return 0;
21 }
22

```

5. Результати виконання завдань, тестування та фактично затрачений час:

Завдання №1 VNS Lab 2

```

Input num to calculate series num: 3
Final result of calculating: 0.7870

```

Час затрачений на виконання завдання: 15хв

Завдання №2 VNS Lab 3

```
e in the 1 power: 2.71828
e in the 1.1 power: 3.00417
e in the 1.2 power: 3.32012
e in the 1.3 power: 3.6693
e in the 1.4 power: 4.0552
e in the 1.5 power: 4.48169
e in the 1.6 power: 4.95303
e in the 1.7 power: 5.47395
e in the 1.8 power: 6.04965
e in the 1.9 power: 6.6859
e in the 2 power: 7.38906
```

```
-----
e in the 1 power: 2.71825
e in the 1.1 power: 3.00411
e in the 1.2 power: 3.3201
e in the 1.3 power: 3.66926
e in the 1.4 power: 4.05513
e in the 1.5 power: 4.48167
e in the 1.6 power: 4.953
e in the 1.7 power: 5.47388
e in the 1.8 power: 6.04963
e in the 1.9 power: 6.68586
e in the 2 power: 7.389
```

Час затрачений на виконання завдання: 2 год

Завдання №3 VNS Lab 7 – Task-1

Час затрачений на виконання завдання: 25хв

```
-70
-257
3
```

Завдання №4 VNS Lab 7 – Task-2

```
Enter two complex numbers:
5
4
1
2
Multiple: (-3,14)
Enter two real numbers:
67
12
Multiple: 804
```

Час затрачений на виконання завдання: 30 хв

Завдання Завдання №5 Class Practice Work

```
Congratulations! You're in online library.
```

```
Available options:
```

- 1 - List all books
- 2 - Take a book (if available)
- 3 - Return a book

```
Enter a Number of the option:
```

```
1
```

```
All the books in our library:
```

- ```

```
- Dune
  - Neuromancer
  - Snow Crash
  - The Hitchhiker's Guide to the Galaxy
  - Brave New World
  - The War of the Worlds
  - Fahrenheit 451
  - The Left Hand of Darkness
  - The Expanse: Leviathan Wakes
  - The Three-Body Problem
  - Do Androids Dream of Electric Sheep?
  - Altered Carbon
  - The Man in the High Castle
  - The Time Machine
  - Ringworld
  - The Moon is a Harsh Mistress
  - The Forever War
- ```
-----
```

```
Choose option:
```

- 1. Return to the main menu.
- 2. End session

```
1
```

```
Available options:
```

- 1 - List all books
- 2 - Take a book (if available)
- 3 - Return a book

```
Enter a Number of the option:
```

```
2
```

```
All books status:
```

- ```
0. Dune
Status: Available
1. Neuromancer
Status: Available
2. Snow Crash
Status: Available
3. The Hitchhiker's Guide to the Galaxy
Status: Available
4. Brave New World
Status: Available
5. The War of the Worlds
Status: Available
6. Fahrenheit 451
```

```
Status: Available
```

```
10. Do Androids Dream of Electric Sheep?
```

```
Status: Available
```

```
11. Altered Carbon
```

```
Status: Available
```

```
12. The Man in the High Castle
```

```
Status: Available
```

```
13. The Time Machine
```

```
Status: Available
```

```
14. Ringworld
```

```
Status: Available
```

```
15. The Moon is a Harsh Mistress
```

```
Status: Available
```

```
16. The Forever War
```

```
Status: Available
```

```
What book would you like to take, Enter the number of it: 11
```

```
You have taken a book. Good reading!
```

```
Choose option:
```

- 1. Return to the main menu.
- 2. End session

```
1
```

```
Available options:
```

- 1 - List all books
- 2 - Take a book (if available)
- 3 - Return a book

```
Enter a Number of the option:
```

```
3
```

```
0. Dune
```

```
1. Neuromancer
```

```
2. Snow Crash
```

```
3. The Hitchhiker's Guide to the Galaxy
```

```
4. Brave New World
```

```
5. The War of the Worlds
```

```
6. Fahrenheit 451
```

```
7. The Left Hand of Darkness
```

```
8. The Expanse: Leviathan Wakes
```

```
9. The Three-Body Problem
```

```
10. Do Androids Dream of Electric Sheep?
```

```
11. Altered Carbon
```

```
12. The Man in the High Castle
```

```
13. The Time Machine
```

```
14. Ringworld
```

```
15. The Moon is a Harsh Mistress
```

```
16. The Forever War
```

```
What book would you like to return? Enter the index
```

```
11
```

```
You have successfully returned a book
```

```
Choose option:
```

- 1. Return to the main menu.
- 2. End session

```
1
```

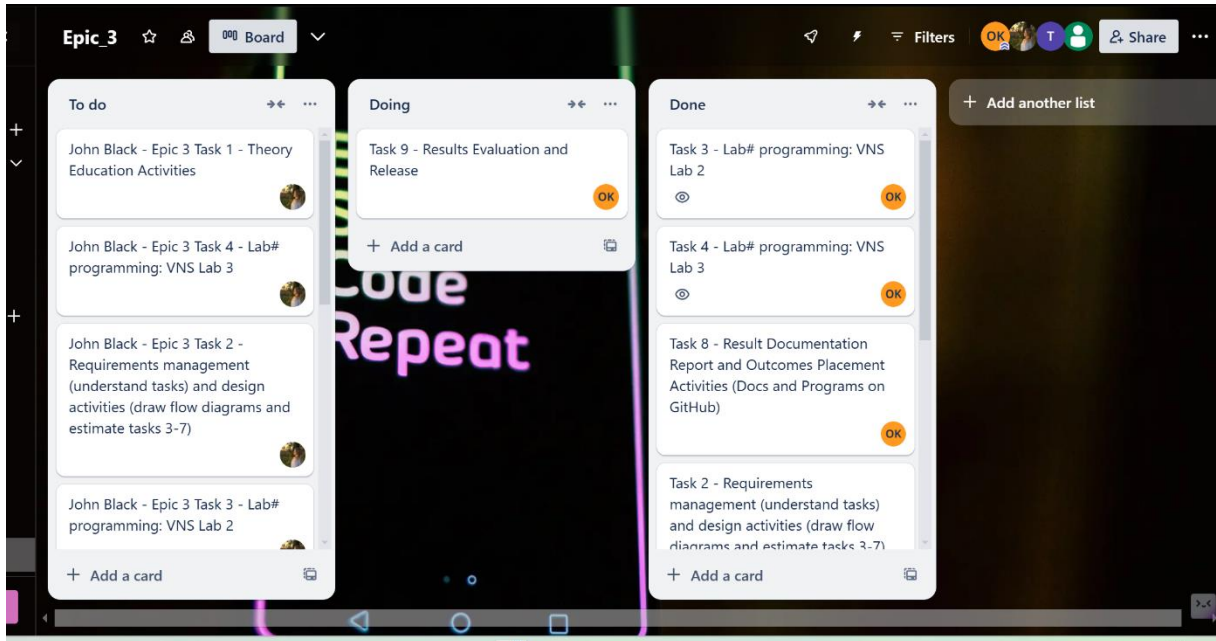
Час затрачений на виконання завдання: 3.5 години

## Завдання №6 Self Practice Work (Шифрування корупційних схем)

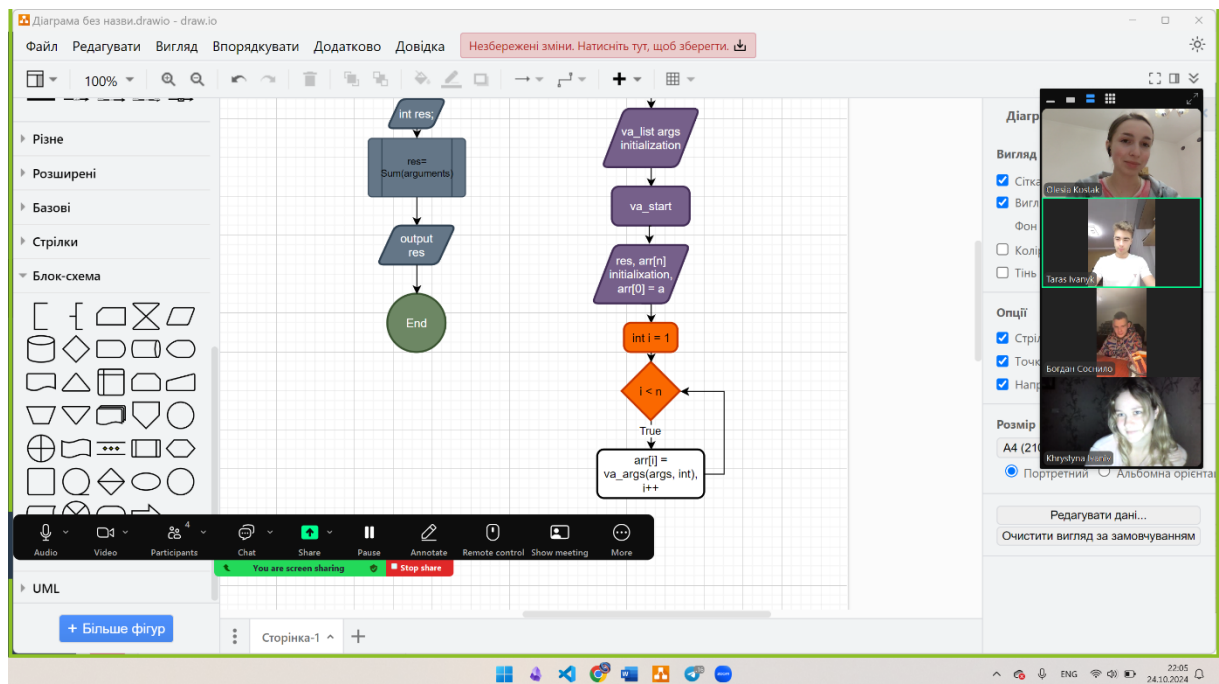
| Створено             | Компілятор | Результат  | Час (сек.) | Пам'ять (МіБ) | Дії                      |
|----------------------|------------|------------|------------|---------------|--------------------------|
| декілька секунд тому | C++ 23     | Зараховано | 0.005      | 1.105         | <a href="#">Перегляд</a> |

Час затрачений на виконання завдання: 10хв

### 6. Кооперація з командою:



- Спільна борда в трелло по 3 епіку



### **Висновки:**

По завершенню завдань Епіка №3, я вдосконалити вміння програмувати шляхом вивчення циклів і функцій. Засвоїла різні види циклів (for, while, do-while) та навчилася керувати їх виконанням за допомогою операторів break і continue. Навчилася працювати з функціями: оголошення, передача параметрів, перевантаження та рекурсія. Розібралася з еліпсисом та просторами імен для ефективної організації коду. Ознайомила з використанням вбудованих функцій і функцій зі змінною кількістю параметрів.