



## 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 ІІІ-13  
Щербан Ярина Олегівна

## **Тема роботи:**

Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та робота з масивами та структурами.

## **Мета роботи:**

Навчитися працювати з одновимірними, двовимірними та динамічними масивами, вказівниками та посиланнями, структурами та вкладеними структурами.

## **Теоретичні відомості:**

1. Введення в Цикли та їх Види в C++:
  - Огляд видів циклів: for, while, do-while.
2. Управління Виконанням Циклів:
  - Застосування операторів break та continue.
  - Умови завершення циклів.
3. Вкладені Цикли
4. Основи Функцій у C++
5. Перевантаження Функцій та Простір Імен
6. Розширені Можливості Функцій:
  - Функції зі змінною кількістю параметрів (еліпсис)
  - 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.
7. Вбудовані Функції в C++

## **Опрацювання теоретичного матеріалу :**

1. Вивчення мови C++ за допомогою сайтів : <https://www.w3schools.com/>, <https://acode.com.ua/>
2. Робота з блок-схемами та Draw io <https://www.programiz.com/article/flowchart-programming>
3. Опрацювала відео щодо створення функцій <https://www.youtube.com/watch?v=VKQ242d-Rag>

## **Виконання роботи :**

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

### **Завдання №1 Епік 3 : Практичне завдання : Менеджмент бібліотеки**

Опис задачі : 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.

### Умови задачі :

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

## **Завдання №2 VNS Lab 2 - Task 1 – Variant 21 :**

Опис задачі : Використовуючи оператор циклу, знайти суму елементів, зазначених у конкретному варіанті. Результат надрукувати, надавши відповідний заголовок.

Умови задачі : Знайти суму 13 членів ряду, у якому 
$$a_n = \frac{\ln(n!)}{n^2}$$

## **Завдання №3 VNS Lab 3 - Task 1 – Variant 21 :**

Опис задачі : 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  $\epsilon$  ( $\epsilon=0.0001$ ). Для порівняння знайти точне значення функції.

Умови задачі :  $y = \arctg x$        $0, 1 \leq x \leq 1$        $n = 40$       
$$S = x - \frac{x^3}{3} + \dots (-1)^n \frac{x^{2n+1}}{2n+1}$$

## **Завдання №4 VNS Lab 7 - Task 1 – Variant 21 :**

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

Умови задачі : Написати функцію (або макровизначення), що знаходить кут трикутника за його сторонами. Написати функцію `angles` с змінною кількістю параметрів, що знаходить кути  $n$ -кутника за заданими сторонами. Написати викликаючу функцію `main`, що звертається до функції `angle` не менше трьох разів з кількістю параметрів 3, 9, 11.

## **Завдання №5 VNS Lab 7 - Task 2 – Variant 21 :**

Опис задачі : Написати перевантажені функції й основну програму, що їх викликає.

### Умови задачі :

- а) для масиву цілих чисел знаходить кількість від'ємних елементів;
- б) для рядка знаходить кількість слів, що закінчуються і починаються на ту ж букву

## Завдання №6 Algotester : Зуби

Опис задачі : Мале Бісеня любить гострити зуби. А Зла Тітонька любить до нього підходити і питатися: «Що, зуби гостриш?». Бісеняті таке не дуже подобається, тому воно придумало робити таке.

У Малого Бісеняти є  $n$  зубів. Кожен зуб має коефіцієнт загостреності  $a_i$ . Також існує межа загостреності  $k$ . Якщо коефіцієнт загостреності певного зуба є більшим чи рівним межі загостреності, то такий зуб вважається загостреним.

Мале Бісеня хоче наступного разу, коли Зла Тітонька його щось запитає, показати їй якнайбільше загострених зубів, що розташовані поспіль.

Допоможіть Малому Бісеняті дізнатися, скільки найбільше зубів воно зможе показати.

Умови задачі : У першому рядку задані два цілих числа  $n$  та  $k$  — кількість зубів та межа загостреності відповідно.

В другому рядку задано  $n$  цілих чисел  $a_i$  — коефіцієнти загостреності зубів.

## Завдання №7 Algotester : Спекотні дні пінгвінів

Опис задачі : Ви собі навіть уявити не можете, як же спекотно пінгвінам на Мадагаскарі. Щоб хоч трішки охолодитись, вони випивають безалкогольні коктейлі, однак і з цим проблемно — руки не пристосовані до такого способу життя. Вам потрібно допомогти визначити пінгвінам, чи зможуть вони випити коктейль, що лежить на столі.

Для простоти будемо вважати, що коктейль на столі — круг із діаметром  $l$ , в той час, як рот пінгвіна в будь-який момент часу — прямокутник із шириною  $w$ , сторони якого паралельні осям координат. В початковий момент часу рот пінгвіна закритий, тому прямокутник вироджений — його висота  $0$  (інакше кажучи, він є горизонтальним відрізком). Верхня щелепа пінгвіна може розкритись не більше ніж на  $u$  дюймів відносно початкової позиції, в той час, як нижня — на  $d$  дюймів.

Будемо вважати, що пінгвін може випити коктейль, якщо той повністю впишеться в його рот. Допоможіть пінгвіну за всіма заданими параметрами визначити, чи зможе він випити коктейль.

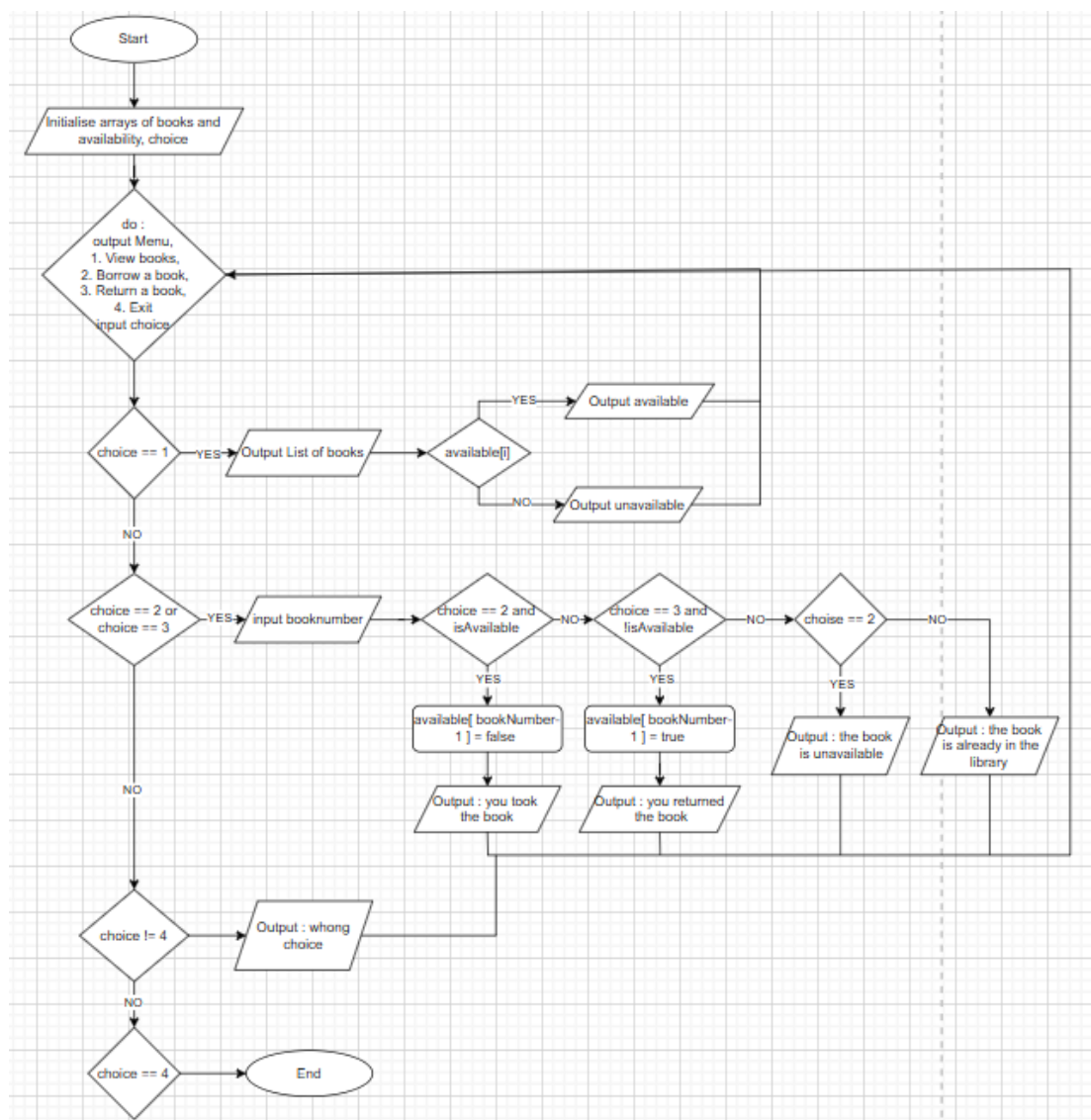
Умови задачі : У єдиному рядку задано 4 натуральні числа  $l$ ,  $w$ ,  $u$  та  $d$  — діаметр коктейлю, ширина рота і на скільки дюймів щелепи можуть розкритися відповідно.

Якщо пінгвін зможе випити коктейль, виведіть рядок `Three times Sex on the Beach, please!`, в протилежному випадку — `Forget about the cocktails, man!`.

### 2. Дизайн та планована оцінка часу виконання завдань:

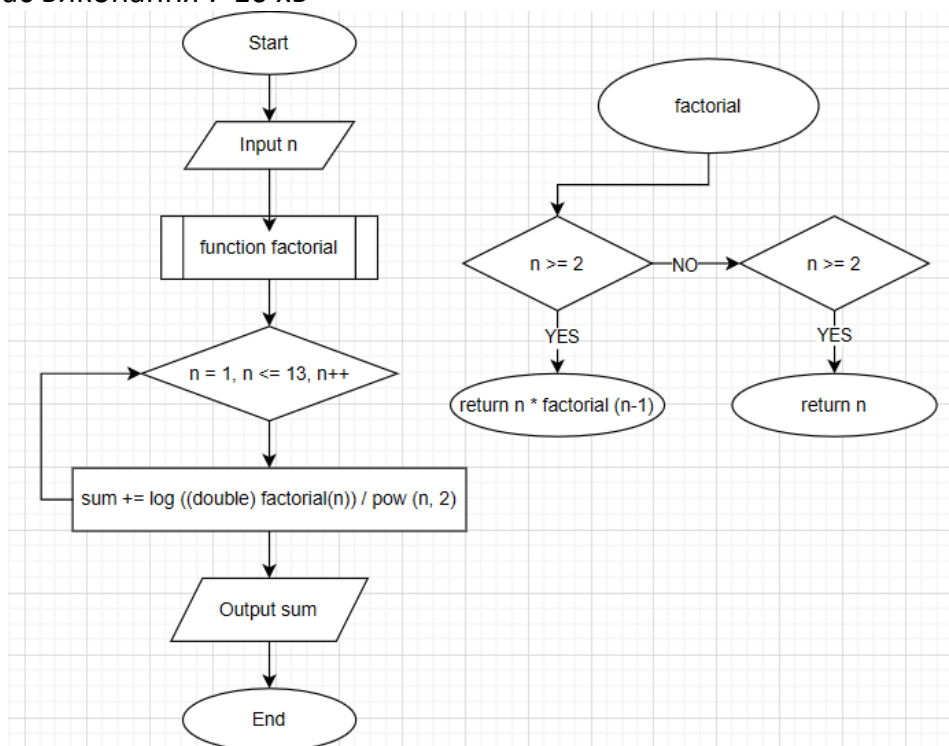
## Завдання №1 Епік 3 : Практичне завдання : Менеджмент бібліотеки

Запланований час виконання : 1 год



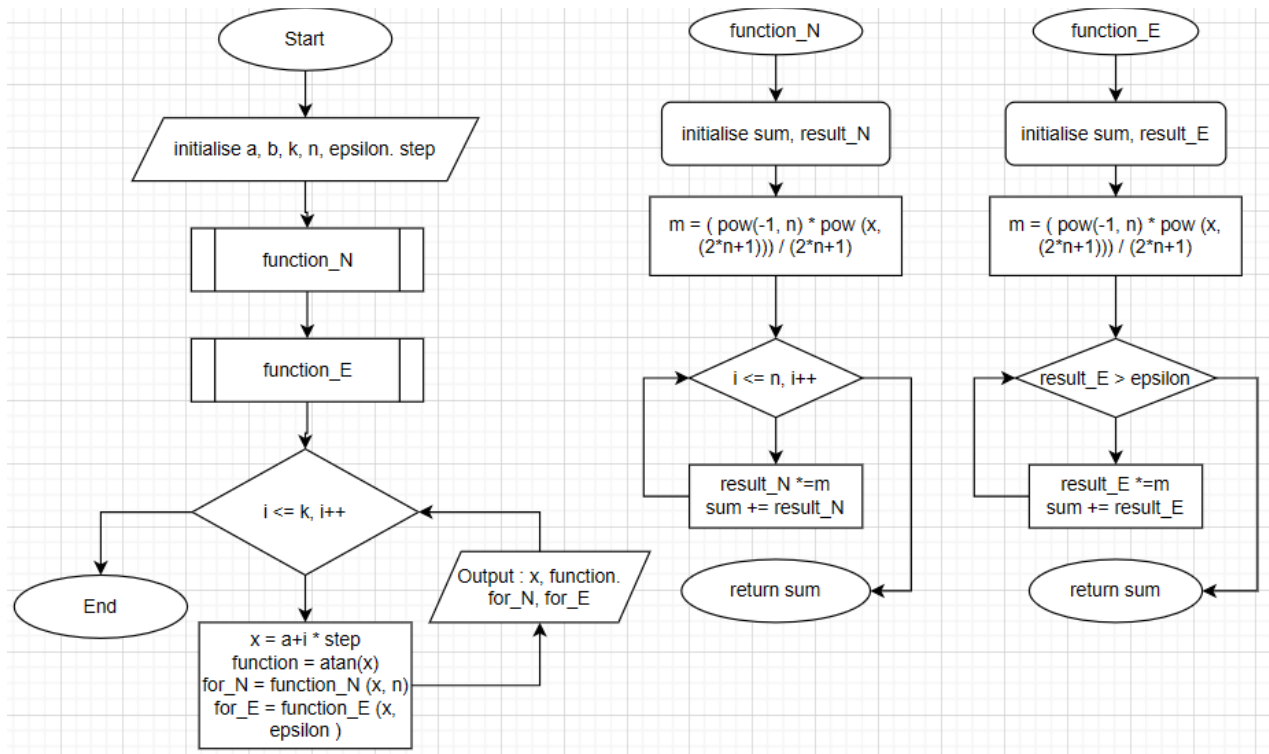
## Завдання №2 VNS Lab 2 - Task 1 – Variant 21 :

Запланований час виконання : 10 хв



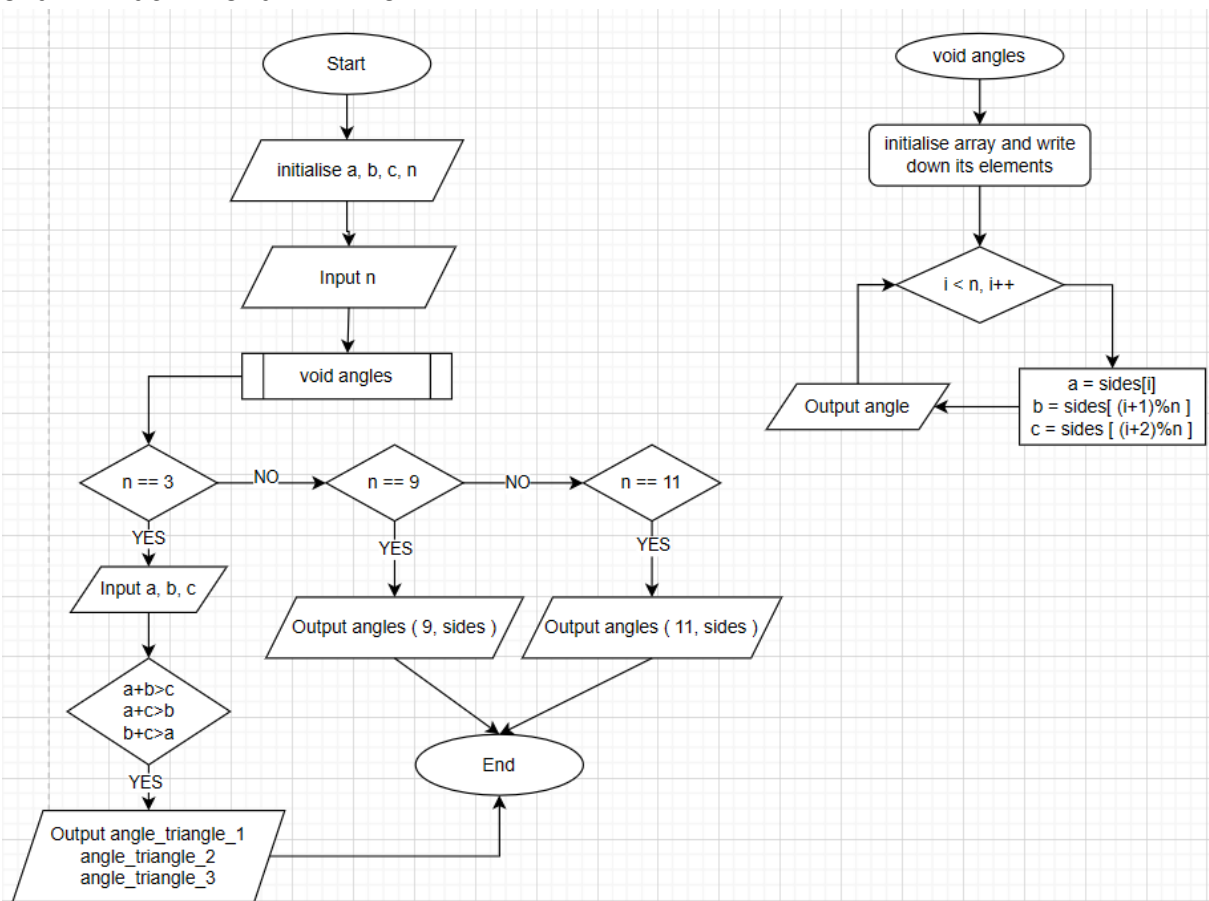
### Завдання №3 VNS Lab 3 - Task 1 – Variant 21 :

Запланований час виконання : 30 хв



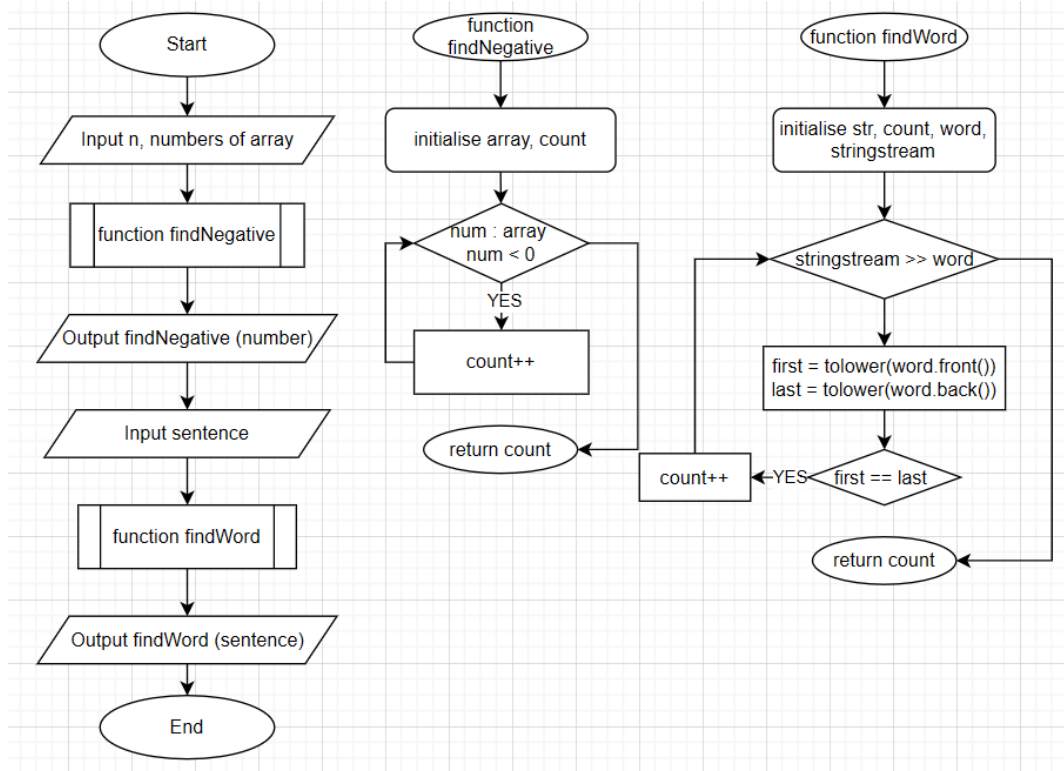
### Завдання №4 VNS Lab 7 - Task 1 – Variant 21 :

Запланований час виконання : 40 хв



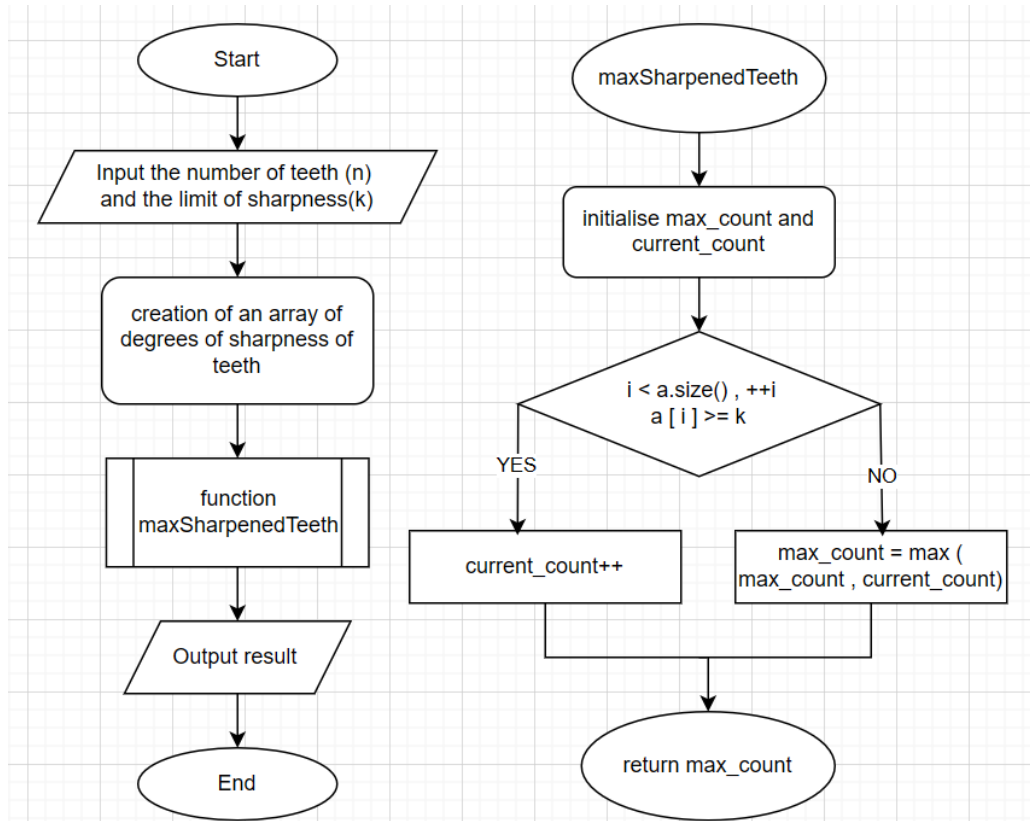
## Завдання №5 VNS Lab 7 - Task 2 – Variant 21 :

Запланований час виконання : 30 хв



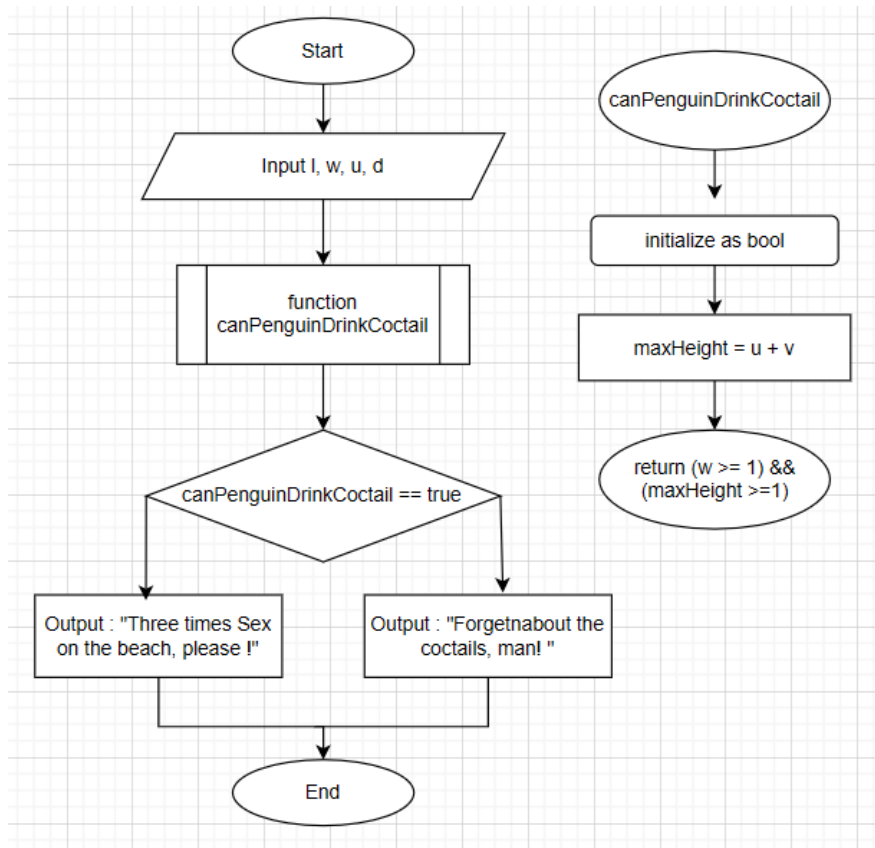
## Завдання №6 Algotester : Зуби

Запланований час виконання : 20 хв

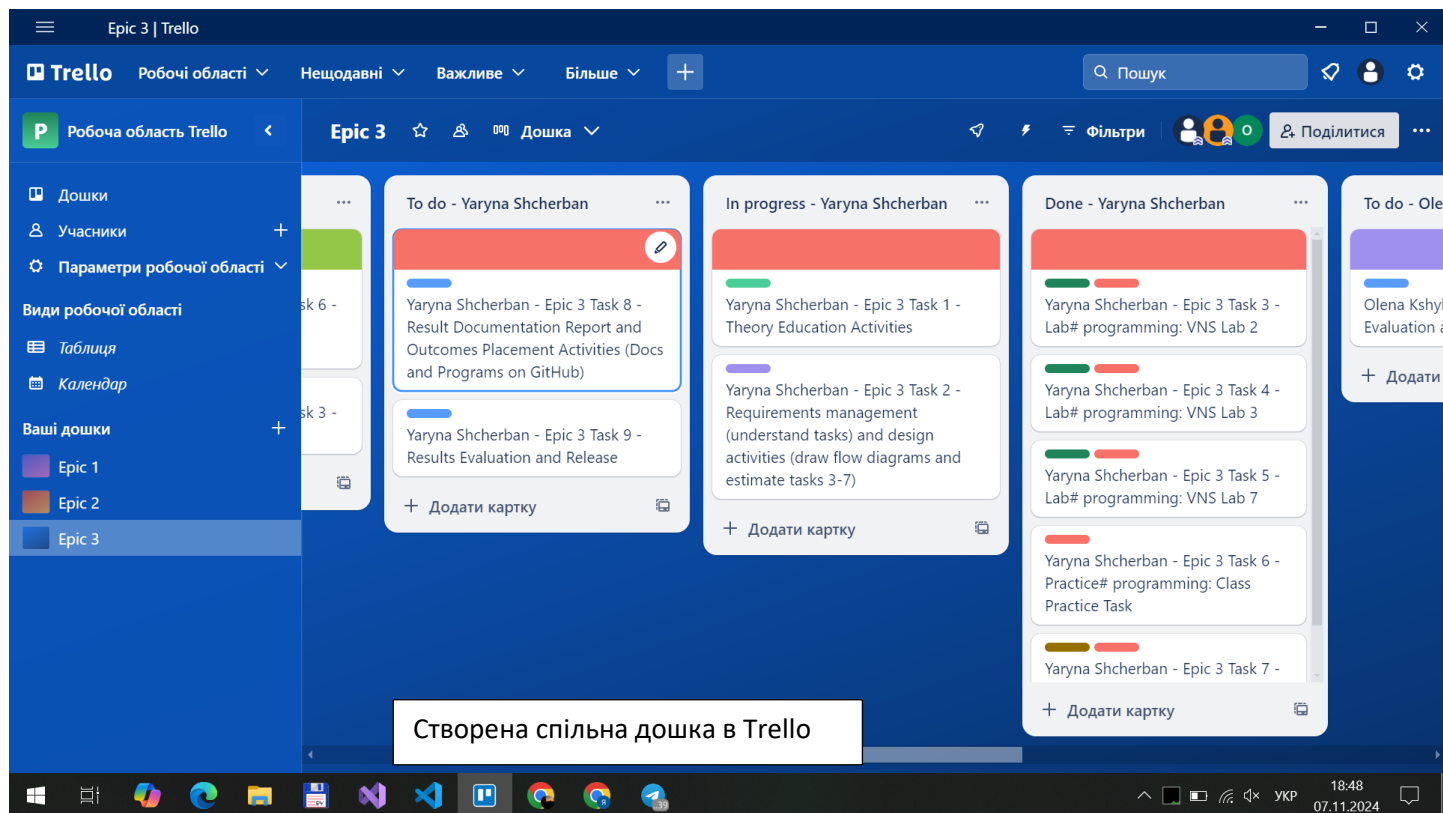


## Завдання №7 Algotester : Спекотні дні пінгвінів

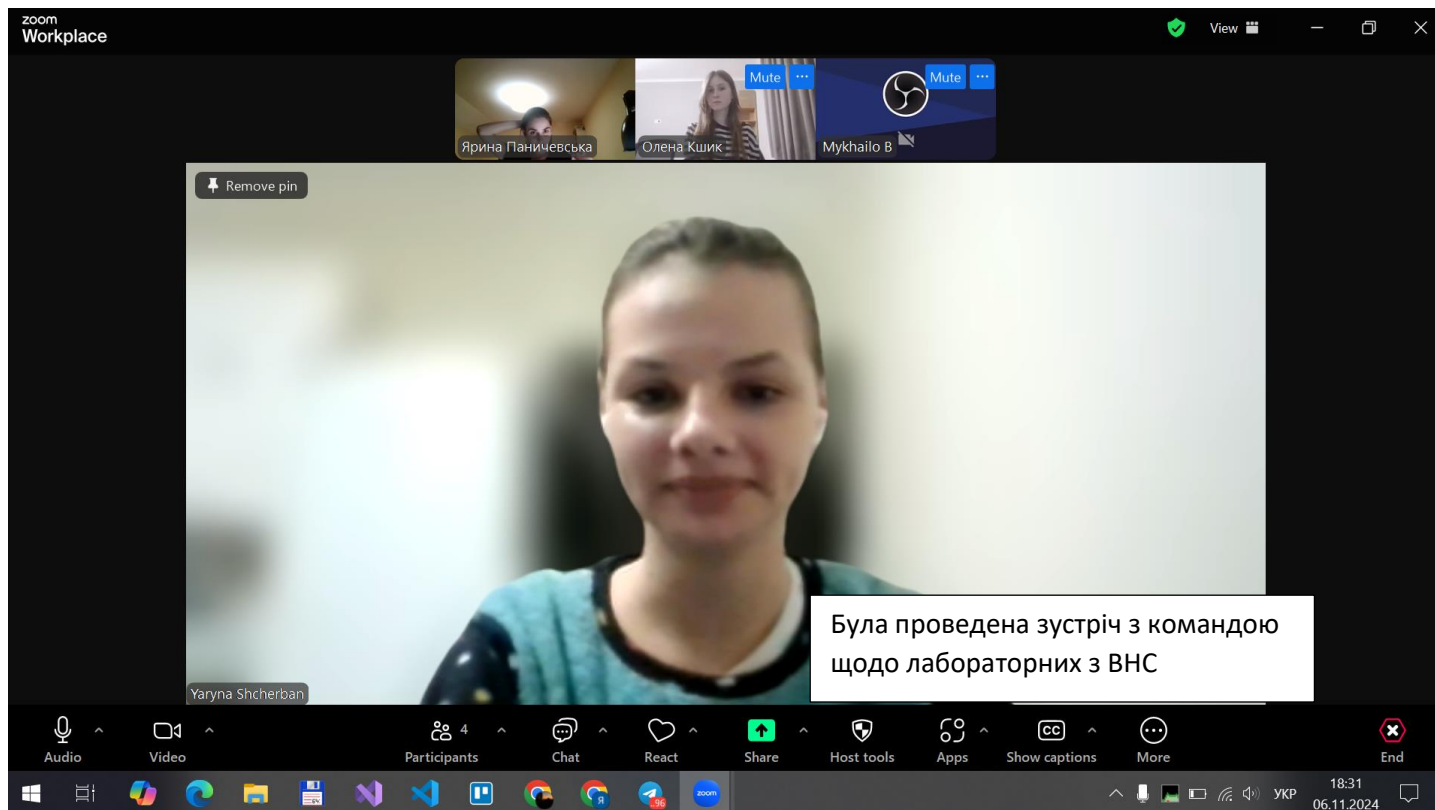
Запланований час виконання : 15 хв



### 3. Конфігурація середовища до виконання завдань:







4. Код програм з посиланням на зовнішні ресурси:

### Завдання №1 Епік 3 : Практичне завдання : Менеджмент бібліотеки

practice\_work\_team\_tasks\_yaryna\_shcherban.cpp

```
Epic 3 > practice_work_team_tasks_yaryna_shcherban.cpp > ...
1  #include <iostream>
2  #include <vector>
3  #include <string>
4
5  using namespace std;
6
7  int main() {
8
9      vector<string> books = {
10         "Divine Rivals",
11         "Once upon a broken heart",
12         "Powerless",
13         "Shatter me",
14         "Fourth wing",
15         "The Atlas Six",
16         "King of wrath",
17         "Endgame",
18         "Twisted games",
19         "The Silent Patient" };
20
21
22     vector<bool> available(books.size(), true);
23     int choice;
24
25     do {
26         cout << "\nMenu:\n";
27         cout << "1. View books\n";
28         cout << "2. Borrow a book\n";
29         cout << "3. Return a book\n";
30         cout << "4. Exit\n";
```

```

31     cout << "Choose an option: ";
32     cin >> choice;
33
34     if (choice == 1) {
35         cout << "List of books:\n";
36         for (size_t i = 0; i < books.size(); ++i) {
37             cout << i + 1 << ". " << books[i];
38             if (available[i]) {
39                 cout << " (Available) ";
40             } else {
41                 cout << " (Unavailable) ";
42             }
43             cout << "\n";
44         }
45
46     } else if (choice == 2 || choice == 3) {
47
48         int bookNumber;
49         cout << "Enter the book number: ";
50         cin >> bookNumber;
51
52         if (bookNumber < 1 || bookNumber > books.size()) {
53             cout << "Incorrect book number. Try again.\n";
54             continue;
55         }
56

```

```

57         bool isAvailable = available[bookNumber - 1];
58
59         if (choice == 2 && isAvailable) {
60
61             available[bookNumber - 1] = false;
62             cout << "You took the book: " << books[bookNumber - 1] << "\n";
63         }
64         else if (choice == 3 && !isAvailable) {
65
66             available[bookNumber - 1] = true;
67             cout << "You returned the book: " << books[bookNumber - 1] << "\n";
68         }
69         else {
70             if (choice == 2) {
71                 cout << "The book is unavailable.\n";
72             } else {
73                 cout << "The book is already in the library.\n";
74             }
75         }
76
77     } else if (choice != 4) {
78         cout << "Wrong choice. Try again.\n";
79     }
80
81 } while (choice != 4);
82

```

```

83     cout << "The program is complete.\n";
84     return 0;
85 }

```

## Завдання №2 VNS Lab 2 - Task 1 – Variant 21 :

vns\_lab\_2\_task\_variant\_21\_yaryna\_shcherban.cpp

Epic 3 > vns\_lab\_2\_task\_variant\_21\_yaryna\_shcherban.cpp > ...

```
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  unsigned long long factorial(int n) {
7      if ( n >= 2 ){
8          return n * factorial ( n - 1 );
9      } else if (n == 1){
10         return n;
11     }
12     return 1;
13 }
14
15 int main() {
16     int n;
17     double sum = 0;
18
19     for ( n = 1 ; n <= 13 ; n++) {
20         sum += log ((double) factorial (n)) / pow (n , 2);
21     }
22     cout << "The sum of the series is : " << sum << endl;
23     return 0;
24 }
```

### Завдання №3 VNS Lab 3 - Task 1 – Variant 21 :

vns\_lab\_3\_task\_variant\_21\_yaryna\_shcherban.cpp

Epic 3 > vns\_lab\_3\_task\_variant\_21\_yaryna\_shcherban.cpp > ...

```
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  double function_N (double x, int n);
7  double function_E (double x, double epsilon);
8
9  double function_N(double x, int n) {
10     double sum = 1.0;
11     double result_N = 1.0;
12     double m = (pow (-1, n) * pow (x, (2*n+1))) / (2*n+1) ;
13     for (int i = 1; i <= n; i++) {
14         result_N *= m;
15         sum += result_N;
16     }
17     return sum;
18 }
19
20 double function_E (double x, double epsilon) {
21     double sum = 1.0;
22     double result_E = 1.0;
23     int n;
24     double m = (pow (-1, n) * pow (x, (2*n+1))) / (2*n+1) ;
25     int i = 1;
26     while (result_E > epsilon) {
27         result_E *= m;
28         sum += result_E;
29         ++i;
30     }
```

```

31     return sum;
32 }
33
34 int main() {
35
36     double a = 0.1;
37     double b = 1.0;
38     int k = 10;
39     int n = 40;
40     double epsilon = 0.0001;
41     double step = (b - a) / k;
42
43
44     for ( int i = 0; i <= k; i++ ) {
45         double x = a + i * step;
46         double function = atan (x);
47         double for_N = function_N (x, n);
48         double for_E = function_E (x, epsilon);
49         cout << "x = " << x << endl;
50         cout << "result = " << function << endl;
51         cout << "result in n = " << for_N << endl;
52         cout << "result in epsilon = " << for_E << endl;
53     }
54 }

```

## Завдання №4 VNS Lab 7 - Task 1 – Variant 21 :

vns\_lab\_7\_task\_1\_variant\_21\_yaryna\_shcherban.cpp

Epic 3 > vns\_lab\_7\_task\_1\_variant\_21\_yaryna\_shcherban.cpp > ...

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <math.h>
4  #include <stdarg.h>
5
6  #define pi 3.1415
7  #define angle_triangle_1(a, b, c) acos(-0.5 * (b * b - a * a - c * c) / (a * c)) * (180 / pi)
8  #define angle_triangle_2(a, b, c) acos(-(c * c - a * a - b * b) / (2.0 * a * b)) * (180 / pi)
9  #define angle_triangle_3(a, b, c) acos(-(a * a - b * b - c * c) / (2.0 * b * c)) * (180 / pi)
10
11 int a, b, c, n;
12
13 void angles (int n, ...) {
14     if ( n < 3 ) {
15         printf("Enter another one number of sides of the polygon! \n");
16         return;
17     }
18     va_list args;
19     va_start(args, n);
20
21     int sides[n];
22     for (int i = 0; i < n; i++) {
23         sides[i] = va_arg(args, double);
24     }
25     va_end(args);
26
27     int i = 0;
28     printf("Angles for %d-gon:\n", n);
29     for (int i = 0; i < n; i++) {
30         float a = sides[i];

```

```

31     float b = sides[(i + 1) % n];
32     float c = sides[(i + 2) % n];
33     printf("Angle %d: %f \n" , i + 1 , angle_triangle_1(a, b, c) );
34 }
35
36 }
37
38
39 int main() {
40
41     printf("Enter the number of sides of the polygon : ");
42     scanf("%d", &n);
43
44     if ( n == 3 ) {
45
46         printf("Input a: ");
47         scanf("%d", &a);
48         printf("Input b: ");
49         scanf("%d", &b);
50         printf("Input c: ");
51         scanf("%d", &c);
52
53         if ((a+b>c)&&(a+c>b)&&(b+c>a)){
54
55             printf("Angle_1 = %f\n", angle_triangle_1(a, b, c));
56             printf("Angle_2 = %f\n", angle_triangle_2(a, b, c));
57             printf("Angle_3 = %f\n", angle_triangle_3(a, b, c));
58
59         } else {
60             printf("\nInput another size of sides!\n");
61         }
62
63     } else if ( n == 9 ) {
64
65         angles(9, 3.0, 5.0, 7.0, 9.0, 11.0, 13.0, 15.0, 17.0, 19.0);
66
67     } else if ( n == 11) {
68
69         angles(11, 3.0, 4.0, 6.0, 8.0, 10.0, 12.0, 14.0, 16.0, 18.0, 20.0, 22.0);
70     }
71
72     return 0;
73 }

```

## Завдання №5 VNS Lab 7 - Task 2 – Variant 21 :

vns\_lab\_7\_task\_2\_variant\_21\_yaryna\_shcherban.cpp

Epic 3 > vns\_lab\_7\_task\_2\_variant\_21\_yaryna\_shcherban.cpp > ...

```
1  #include <iostream>
2  #include <string>
3  #include <sstream>
4  #include <vector>
5  #include <cctype>
6
7  using namespace std;
8
9  int findNegative (const vector<int>& array) {
10     int count = 0;
11     for (int num : array) {
12         if (num < 0) {
13             count++;
14         }
15     }
16     return count;
17 }
18
19 int findWord(const string& str) {
20     int count = 0;
21     stringstream ss(str);
22     string word;
23
24     while (ss >> word) {
25         char first = tolower(word.front());
26         char last = tolower(word.back());
27
28         if (first == last) {
29             count++;
30         }
31     }
32     return count;
33 }
34
35 int main() {
36     vector<int> numbers;
37     int n;
38     cout << "Enter the size of the array : ";
39     cin >> n;
40     cout << "Enter elements for array : ";
41     for (int i = 0; i < n; i++) {
42         int Number;
43         cin >> Number;
44         numbers.push_back(Number);
45     }
46     cout << "Number of negative elements in the array : " << findNegative (numbers) << endl;
47
48     string sentence ;
49     cout << "Enter the sentence: ";
50     cin >> sentence;
51     cout << "Number of words beginning and ending with the same letter : " << findWord(sentence) << endl;
52
53     return 0;
54 }
55
56 }
```

## Завдання №6 Algotester : Зуби

self\_practice\_work\_algotester\_task\_1\_yaryna\_shcherban.cpp

```
Epic 3 > self_practice_work_algotester_task_1_yaryna_shcherban.cpp > ...
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  int maxSharpenedTeeth(const vector<int>& a, int k) {
6      int max_count = 0;
7      int current_count = 0;
8      for (int i = 0; i < a.size(); ++i) {
9          if (a[i] >= k) {
10             current_count++;
11         } else {
12             max_count = max(max_count, current_count);
13             current_count = 0;
14         }
15     }
16     max_count = max(max_count, current_count);
17     return max_count;
18 }
19
20 int main() {
21
22     int n, k;
23     cin >> n;
24     cin >> k;
25
26     vector<int> a(n);
27     for (int i = 0; i < n; ++i) {
28         cin >> a[i];
29     }
30     int result = maxSharpenedTeeth(a, k);
31
32     cout << result << endl;
33     return 0;
34 }
```

## Завдання №7 Algotester : Спекотні дні пінгвінів

self\_practice\_work\_algotester\_task\_2\_yaryna\_shcherban.cpp

```

Epic 3 > G+ self_practice_work_algotester_task_2_yaryna_shcherban.cpp > ...
1  #include <iostream>
2  using namespace std;
3
4  bool canPenguinDrinkCocktail(int l, int w, int u, int d) {
5      int maxHeight = u + d;
6      return (w >= 1) && (maxHeight >= 1);
7  }
8
9  int main() {
10     int l, w, u, d;
11     cin >> l >> w >> u >> d;
12
13     if (canPenguinDrinkCocktail(l, w, u, d) == true ) {
14         cout << "Three times Sex on the Beach, please!" << endl;
15     } else {
16         cout << "Forget about the cocktails, man!" << endl;
17     }
18
19     return 0;
20 }

```

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

## Завдання №1 Епік 3 : Практичне завдання : Менеджмент бібліотеки

Фактично затрачений час – 1 год

|   |   |   |
|---|---|---|
| <pre> Menu: 1. View books 2. Borrow a book 3. Return a book 4. Exit Choose an option: 1 List of books: 1. Divine Rivals (Available) 2. Once upon a broken heart (Available) 3. Powerless (Available) 4. Shatter me (Available) 5. Fourth wing (Available) 6. The Atlas Six (Available) 7. King of wrath (Available) 8. Endgame (Available) 9. Twisted games (Available) 10. The Silent Patient (Available) </pre> | <pre> Menu: 1. View books 2. Borrow a book 3. Return a book 4. Exit Choose an option: 2 Enter the book number: 4 You took the book: Shatter me </pre> | <pre> Menu: 1. View books 2. Borrow a book 3. Return a book 4. Exit Choose an option: 3 Enter the book number: 4 You returned the book: Shatter me </pre> |
| <pre> Menu: 1. View books 2. Borrow a book 3. Return a book 4. Exit Choose an option: 5 Wrong choice. Try again.  Menu: 1. View books 2. Borrow a book 3. Return a book 4. Exit Choose an option: 4 The program is complete. PS C:\Yaryna\University_1\Programming\Epic 3&gt; </pre>  |   |   |



## Завдання №2 VNS Lab 2 - Task 1 – Variant 21 :

Фактично затрачений час – 10 хв

```
' --dbgExe=c:\msys64\ucrt64\bin\gdb.exe' '--inter
The sum of the series is : 2.01092
PS C:\Yaryna\University_1\Programming\Epic 3> █
```

## Завдання №3 VNS Lab 3 - Task 1 – Variant 21 :

Фактично затрачений час – 40 хв

|  |   |
|--|---|
| x = 0.1<br>result = 0.0996687<br>result in n = 1<br>result in epsilon = 1<br>x = 0.19<br>result = 0.187762<br>result in n = 1<br>result in epsilon = 1.23456<br>x = 0.28<br>result = 0.273009<br>result in n = 1<br>result in epsilon = 1.38887<br>x = 0.37<br>result = 0.35438<br>result in n = 1<br>result in epsilon = 1.58727<br>x = 0.46<br>result = 0.431139<br>result in n = 1<br>result in epsilon = 1.85178 | x = 0.55<br>result = 0.502843<br>result in n = 1<br>result in epsilon = 2.22214<br>x = 0.64<br>result = 0.569313<br>result in n = 1<br>result in epsilon = 2.77763<br>x = 0.73<br>result = 0.630578<br>result in n = 1<br>result in epsilon = 3.70349<br>x = 0.82<br>result = 0.686818<br>result in n = 1<br>result in epsilon = 5.55515<br>x = 0.91<br>result = 0.738313<br>result in n = 1.00001<br>result in epsilon = 11.1101 |
|--|---|

## Завдання №4 VNS Lab 7 - Task 1 – Variant 21 :

Фактично затрачений час – 1 год

|   |  |
|---|--|
| Enter the number of sides of the polygon : 3<br>Input a: 3<br>Input b: 4<br>Input c: 5<br>Angle_1 = 53.131669<br>Angle_2 = 90.002654<br>Angle_3 = 36.870985 | Enter the number of sides of the polygon : 9<br>Angles for 9-gon:<br>Angle 1: 38.214338<br>Angle 2: 50.705015<br>Angle 3: 54.697088<br>Angle 4: 56.546656<br>Angle 5: 57.565261<br>Angle 6: 58.188669<br>Angle 7: 58.598842<br>Angle 8: 128.148293 |
|---|--|

## Завдання №5 VNS Lab 7 - Task 2 – Variant 21 :

Фактично затрачений час – 30 хв

```
Enter the size of the array : 18
Enter elements for array : 2 3 -5 2 6 8 -5 -7 -1 3 5 -7 -3 -7 4 2 6 9
Number of negative elements in the array :7
Enter the sentence: Anna loves her car
Number of words beginning and ending with the same letter :1
PS C:\Yaryna\University_1\Programming\Epic 3> █
```

## Завдання №6 Algotester : Зуби

Фактично затрачений час – 10 хв

|           |             |        |            |       |       |         |
|-----------|-------------|--------|------------|-------|-------|---------|
| день тому | 0182 - Зуби | C++ 23 | Зараховано | 0.046 | 1.469 | 1859689 |
|-----------|-------------|--------|------------|-------|-------|---------|

```
7 4
7 1 4 7 6 3 4
3
PS C:\Yaryna\University_1\Programming\Epic 3>
```

## Завдання №7 Algotester : Спекотні дні пінгвінів

Фактично затрачений час – 15 хв

|               |                               |        |            |       |       |         |
|---------------|-------------------------------|--------|------------|-------|-------|---------|
| 2 години тому | 0163 - Спекотні дні пінгвінів | C++ 23 | Зараховано | 0.003 | 1.047 | 1860357 |
|---------------|-------------------------------|--------|------------|-------|-------|---------|

```
7 11 4 4
Three times Sex on the Beach, please!
PS C:\Yaryna\University_1\Programming\Epic 3>

10 8 7 4
Forget about the cocktails, man!
PS C:\Yaryna\University_1\Programming\Epic 3>
```

**Висновок :** Під час виконання лабораторної роботи було успішно засвоєно основні принципи роботи з одновимірними та двовимірними масивами, динамічними масивами, вказівниками та посиланнями. Розглянуто основні методи створення та обробки масивів, а також алгоритми їх обробки для вирішення різних завдань. Навички роботи з динамічними масивами і використання вказівників дозволяють ефективно керувати пам'яттю під час виконання програм.

Посилання на Pull Request : [https://github.com/artificial-intelligence-department/ai\\_programming\\_playground\\_2024/pull/250](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/250)