

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 – 12
Лящук Соломія

Львів 2024

Epic 3

Tasks:

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

Sources:

<https://acode.com.ua/urok-66-operatory-upravlinnya-potokom-vykonannya-program/>

<https://acode.com.ua/urok-67-operatory-umovnogo-rozgaluzhennya-if-else/>

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

<https://www.educative.io/answers/how-to-use-the-ellipsis-in-cpp>

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

<https://acode.com.ua/urok-107-vbudovani-funktsiyi/>

Task 1

1. Введення в Цикли та їх Види в C++

- 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.

2. Управління Виконанням Циклів:

- Застосування операторів break та continue.
- Умови завершення циклів.
- Передчасне завершення виконання циклу.
- Приклади та вправи з управлінням циклами.

3. Вкладені Цикли:

- Поняття та важливість вкладених циклів.
- Реалізація вкладених циклів: приклади для різних сценаріїв.
- Практичні завдання на вкладені цикли.

4. Основи Функцій у C++:

- Визначення та оголошення функцій.
- Параметри функцій: передача за значенням і за посиланням.
- Параметри за замовчуванням.

- Повернення значень з функцій.
- Приклади створення та використання функцій.
- 5. Перевантаження Функцій та Простір Імен:
 - Концепція перевантаження функцій.
 - Правила та приклади перевантаження функцій.
 - Поняття та використання просторів імен.
 - Вкладені простори імен (C++ 17)
 - Роль просторів імен у організації коду.
- 6. Розширені Можливості Функцій:
 - 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.
- 7. Вбудовані Функції в C++:
 - Огляд вбудованих функцій у C++.
 - Приклади використання стандартних функцій у програмуванні.
 - Роль вбудованих функцій у спрощенні коду.
 - Практичні завдання для розуміння вбудованих функцій.

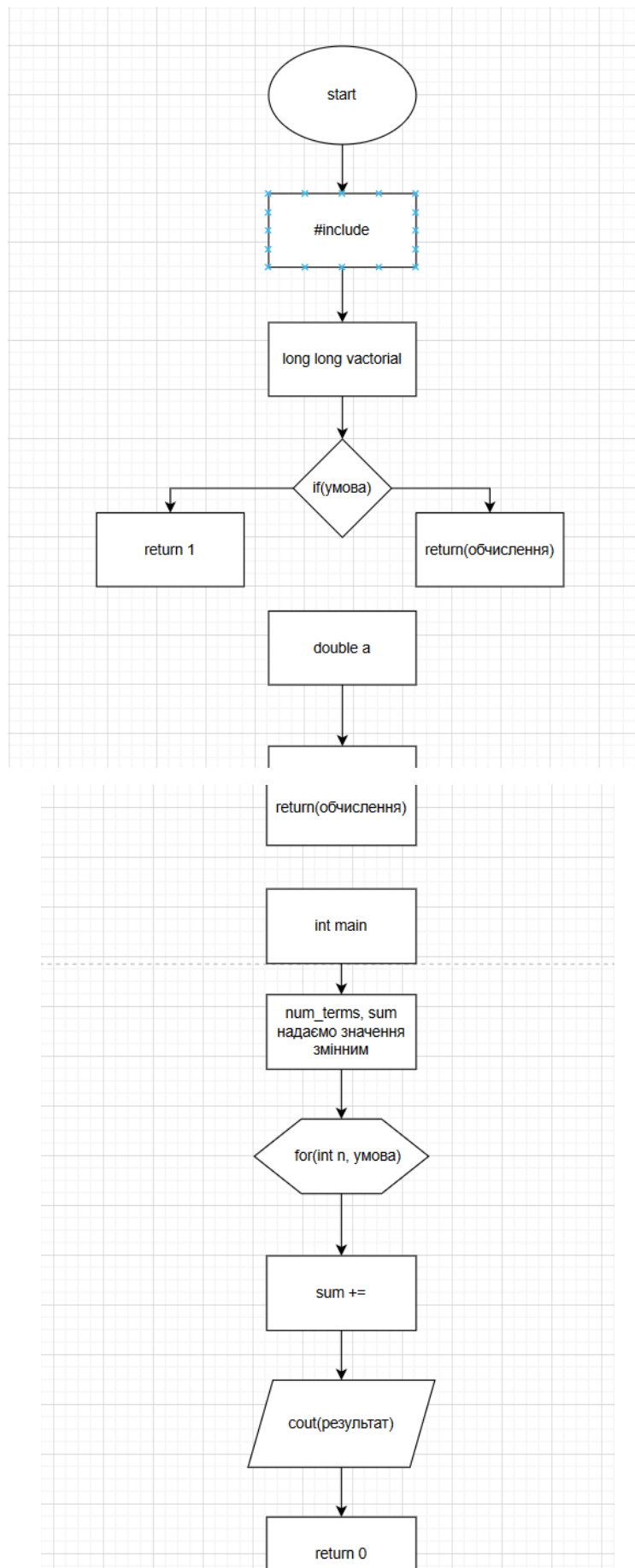
Task 2

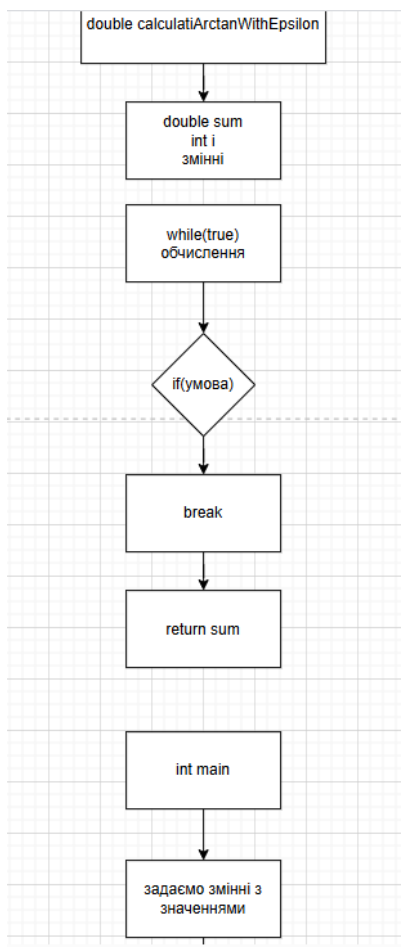
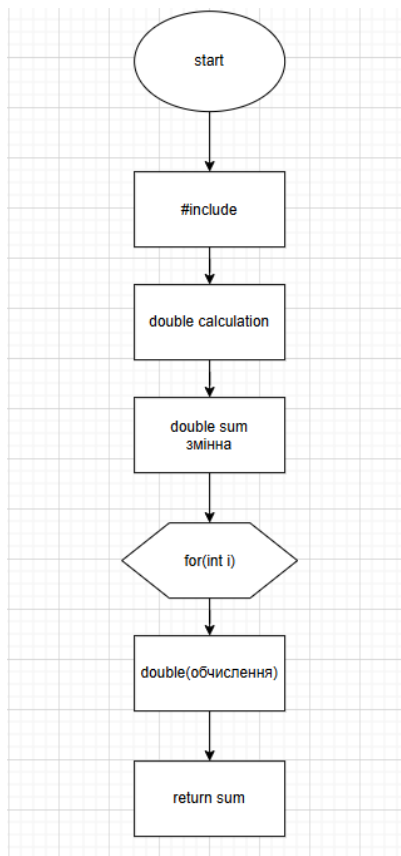
Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)

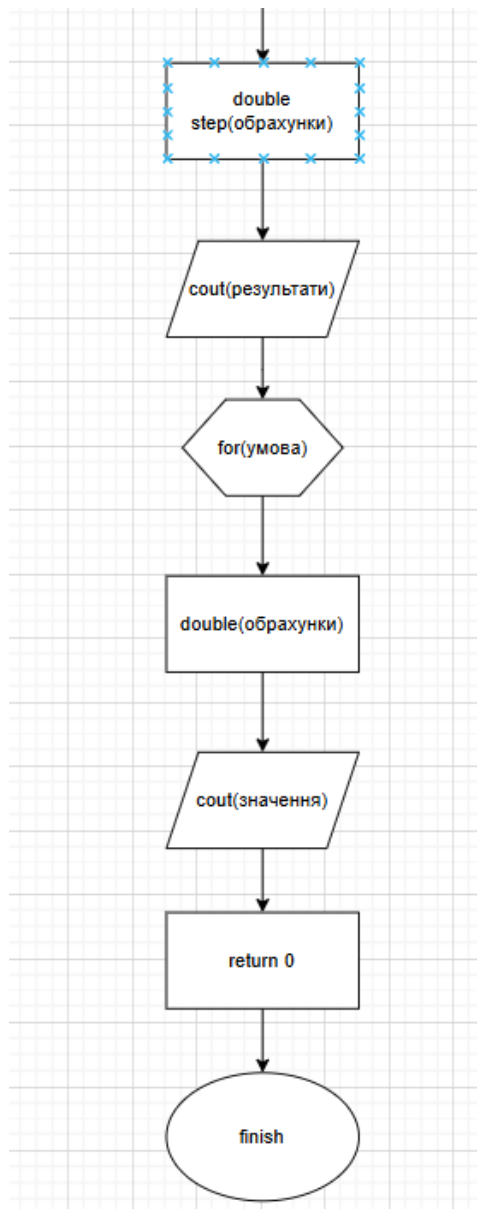
Програмні коди – номер блок-схеми:

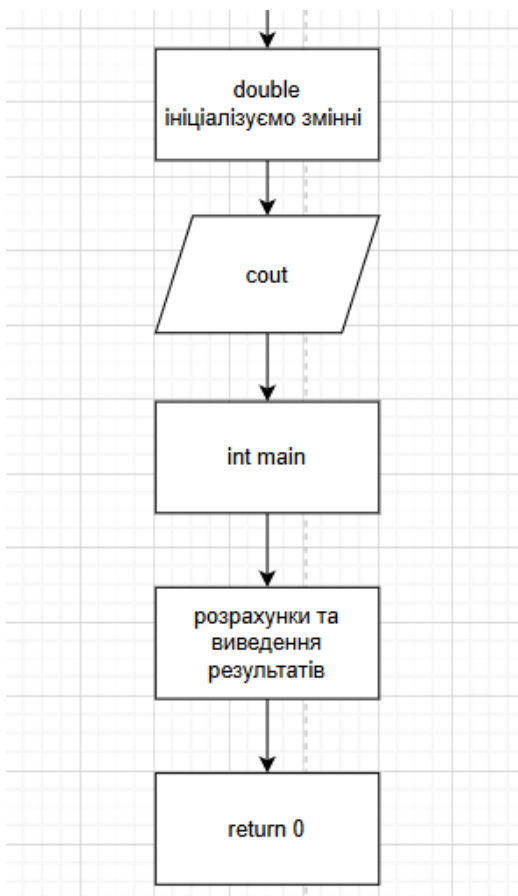
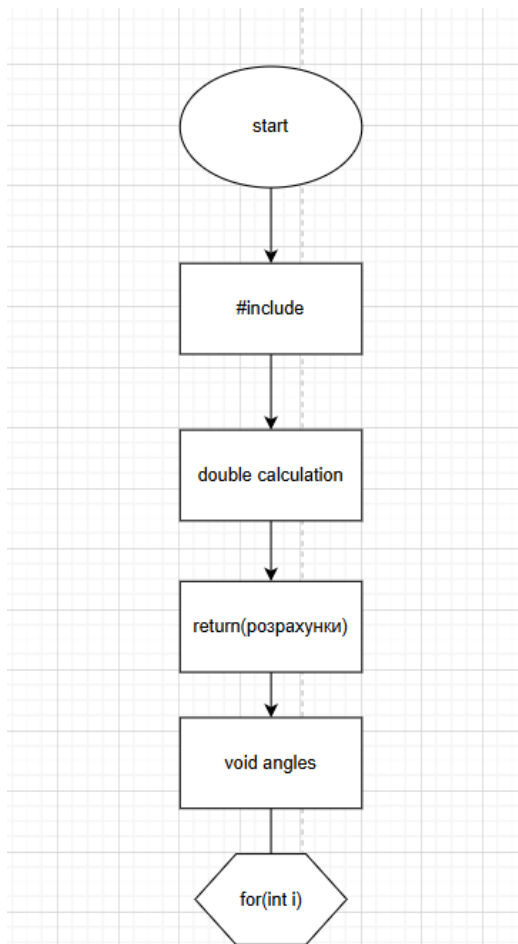
- 1. VNS Lab 2
- 2. VNS Lab 3
- 3. VNS Lab 7 (задача 1)
- 4. VNS Lab 7 (задача 2)
- 5. Class Practice Task
- 6. Self Practice Task

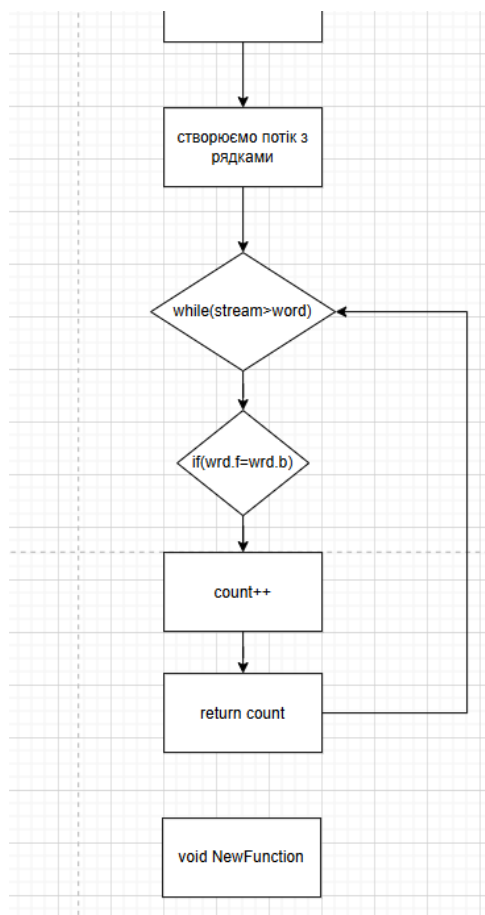
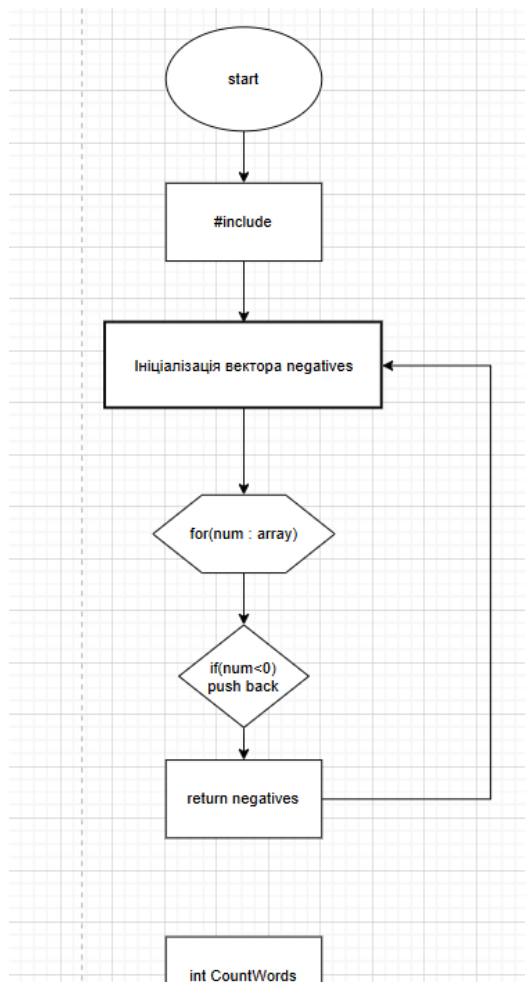
1)

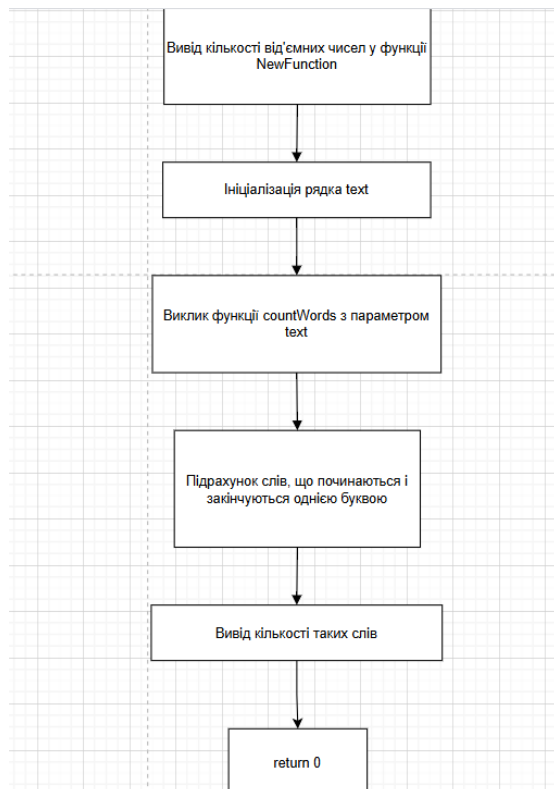
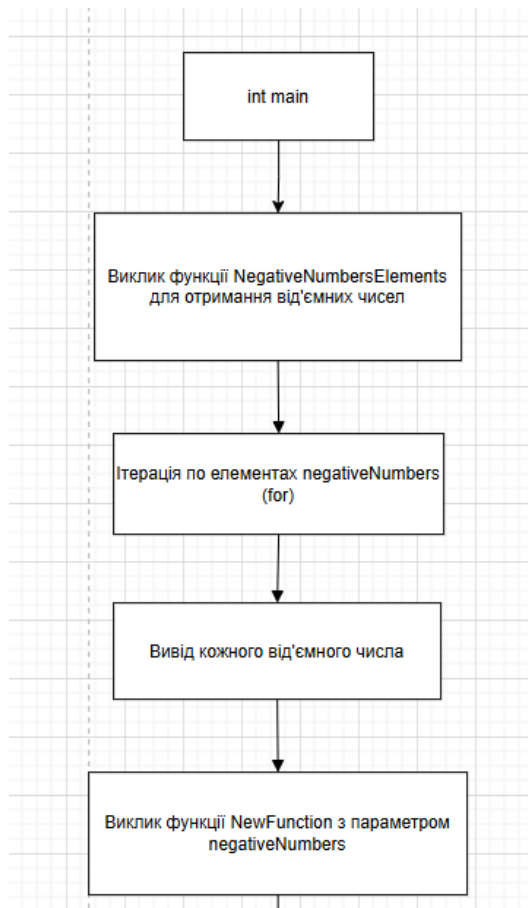




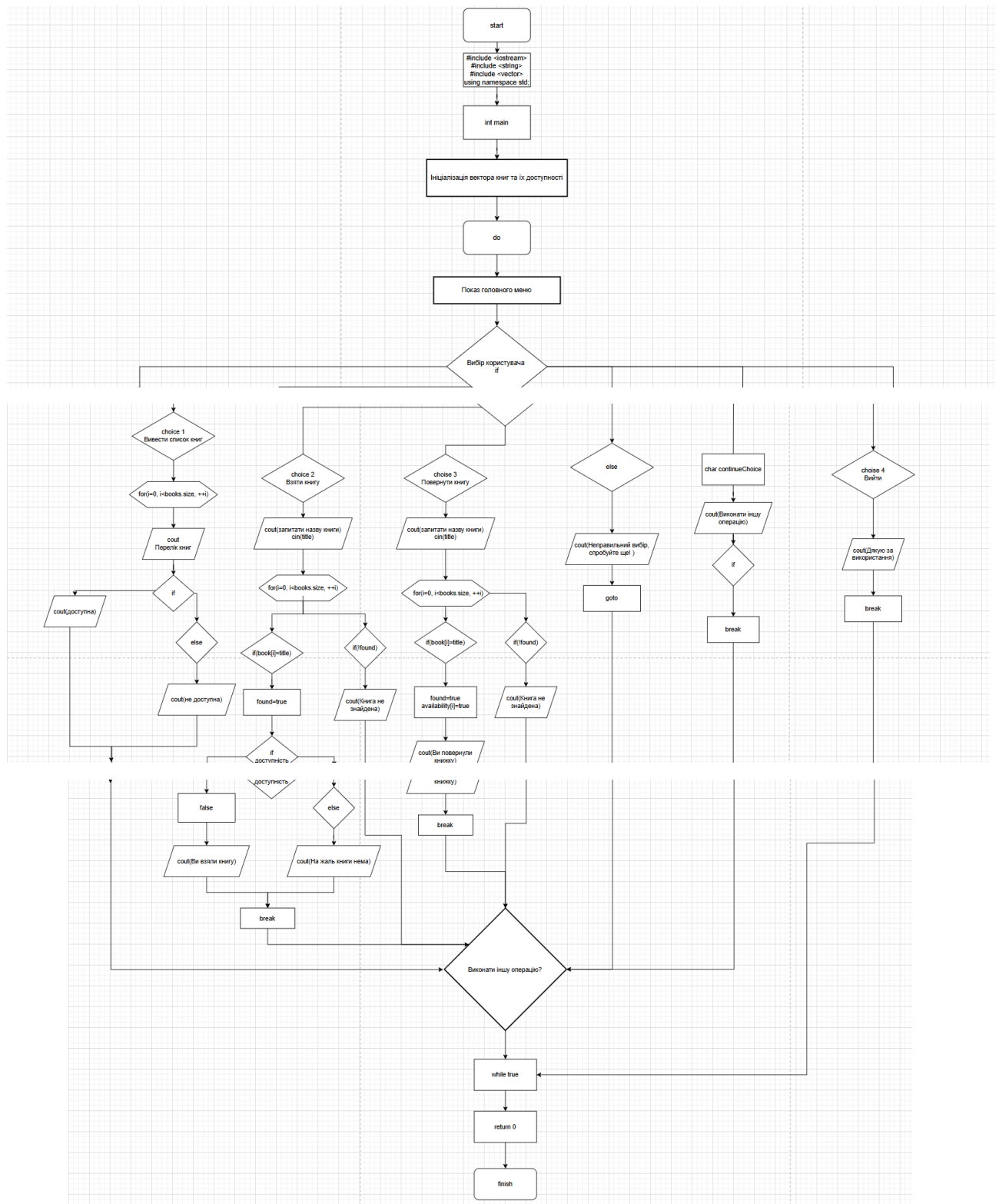




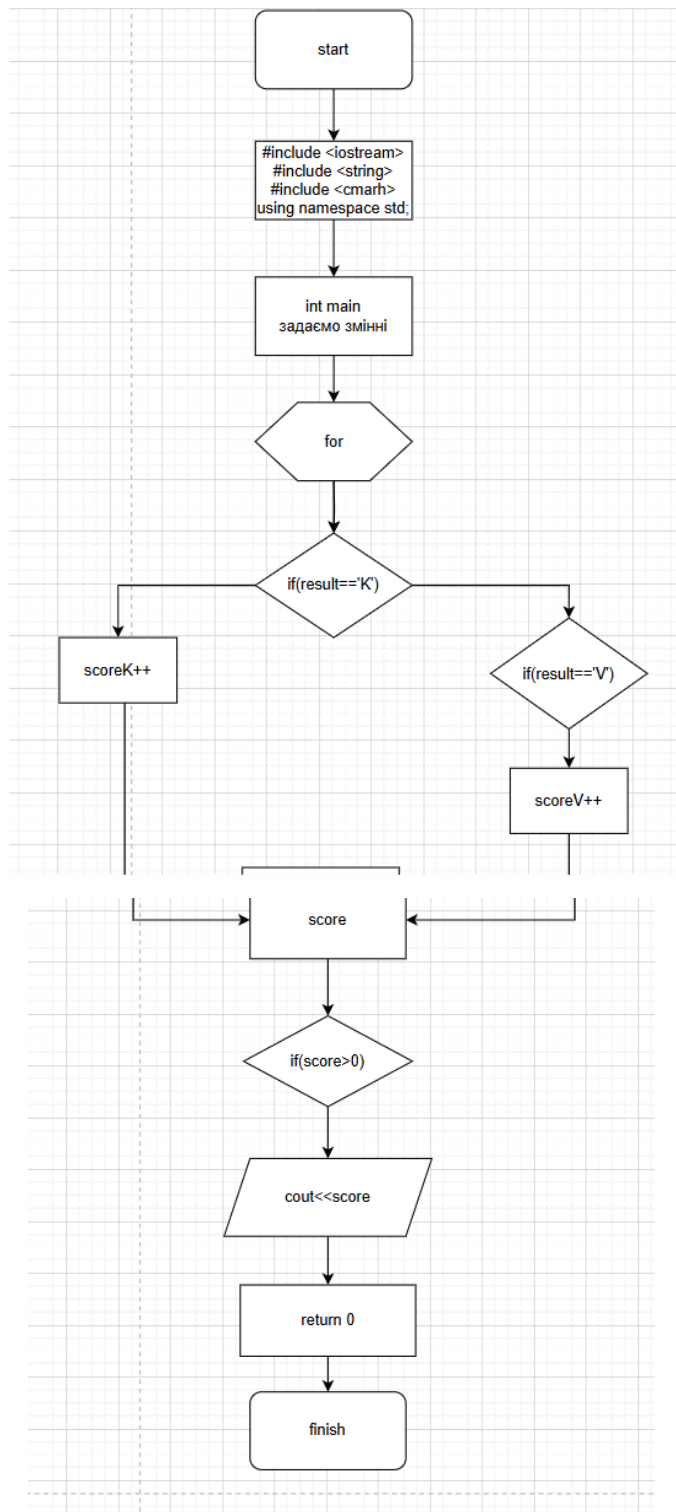




5)



6)



Task 3

VNS Lab 2

V – 21

$a(n) = \ln(n!)/n^2$ (знайти суму перших 13 членів).

В цьому завданні я використала математичні формулу для розрахунків та цикли.

Expected time: 40 min

Time: 60 min

```
vns_lab_2_task_solomia_liashchuk.cpp X
C: > cpp > vns_lab_2_task_solomia_liashchuk.cpp > main()
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  long long factorial(int n)
6  {
7      long long result = 1;
8      if(n == 0 || n == 1)
9      {
10         return 1;
11     }
12     else
13     {
14         return n * factorial(n - 1);
15     }
16 }
17
18 double a(int n)
19 {
20     return log(factorial(n) / pow(n, 2));
21 }
22
23 int main()
24 {
25     const int num_terms = 13;
26     double sum = 0.0;
27     for(int n = 1; n <= num_terms; ++n)
28     {
29         sum += a(n);
30     }
31
32     cout << "Сума перших " << num_terms << " членів ряду дорівнює: " << sum << endl;
33
34     return 0;
35 }
36
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
● PS C:\Users\olesi> & 'c:\Users\olesi\.vscode\extension
engine-In-srextbo.1mz' '--stdout=Microsoft-MIEngine-Out-
'--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter
Сума перших 13 членів ряду дорівнює: 79.0031
○ PS C:\Users\olesi>
```

Task 4

VNS Lab 3

vns_lab_3_task_solomia_liashchuk.cpp X

C: > cpp > vns_lab_3_task_solomia_liashchuk.cpp > calculateArctanWithEpsilon(double, double)

```
1  #include <iostream>
2  #include <cmath>
3  #include <iomanip>
4  using namespace std;
5
6  double calculateArctan(double x, int terms)
7  {
8      double sum = 0.0;
9      for(int i = 0; i < terms; i++)
10     {
11         double term = pow(-1, i) * pow(x, 2 * i + 1) / (2 * i + 1);
12         sum += term;
13     }
14     return sum;
15 }
16
17 double calculateArctanWithEpsilon(double x, double epsilon)
18 {
19     double sum = 0.0;
20     int i = 0;
21     while (true)
22     {
23         double term = pow(-1, i) * pow(x, 2 * i + 1) / (2 * i + 1);
24         sum += term;
25         if (abs(term) < epsilon) {
26             break;
27         }
28         i++;
29     }
30     return sum;
31 }
32 int main()
```

```

32  int main()
33  {
34      double start = 0.1;
35      double end = 1.0;
36      int intervals = 10;
37      int terms = 40;
38      double epsilon = 0.0001;
39
40      double step = (end - start) / intervals;
41
42      cout << fixed << setprecision(8);
43      cout << "Calculation results:\n";
44      for (int i = 0; i <= intervals; i++)
45      {
46
47          double x = start + i * step;
48
49          double fixedN = calculateArctan(x, terms);
50
51          double untilEpsilon = calculateArctanWithEpsilon(x, epsilon);
52
53          double exact = atan(x);
54
55
56          cout << "X=" << x
57              << " FixedN=" << fixedN
58              << " UntilEpsilon=" << untilEpsilon
59              << " Exact=" << exact << endl;
60      }
61
62      return 0;
63  }

```

Task 5

VNS Lab 7 (завдання 1)

V – 21

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.

В цьому кодї я використала математичні формули та масиви для запису значень кутів

```

C: > cpp > vns_lab_7_task_1_variant_1_solomia_liashchuk.cpp > angles(int, double [])
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  double calculateAngele(double a, double b, double c)
6  {
7      return acos((pow(a, 2) + pow(b, 2) - pow(c, 2)) / (2 * a * b));
8  }
9
10 void angles(int n, double sides[])
11 {
12     for (int i = 0; i < n; i++)
13     {
14         double a = sides[i];
15         double b = sides[(i + 1) % n];
16         double c = sides[(i + 2) % n];
17         double angle = calculateAngele(a, b, c);
18         cout << "Angle " << (i + 1) << ": " << angle << " degrees" << endl;
19     }
20 }
21
22 int main()
23 {
24     double triangleSides[3] = {3.0, 4.0, 5.0};
25     cout << "Angles of triangle:" << endl;
26     angles(3, triangleSides);
27     double polygon9Sides[9] = {3.0, 4.0, 5.0, 2.0, 6.0, 7.0, 8.0, 9.0, 10.0};
28     cout << "\nAngles of polygon (9 sides):" << endl;
29     angles(9, polygon9Sides);
30
31     double polygon11Sides[11] = {3.0, 4.0, 5.0, 2.0, 6.0, 7.0, 8.0, 9.0, 10.0, 11.0, 12.0};
32     cout << "\nAngles of polygon (11 sides):" << endl;
33     angles(11, polygon11Sides);
34
35     return 0;
36 }

```

```

Angles of triangle:
Angle 1: 1.5708 degrees
Angle 2: 0.643501 degrees
Angle 3: 0.927295 degrees

Angles of polygon (9 sides):
Angle 1: 1.5708 degrees
Angle 2: 0.389761 degrees
Angle 3: 1.92837 degrees
Angle 4: 1.95519 degrees
Angle 5: 1.31812 degrees
Angle 6: 1.28104 degrees
Angle 7: 1.25297 degrees
Angle 8: 0.299258 degrees
Angle 9: -nan degrees

Angles of polygon (11 sides):
Angle 1: 1.5708 degrees
Angle 2: 0.389761 degrees
Angle 3: 1.92837 degrees
Angle 4: 1.95519 degrees
Angle 5: 1.31812 degrees
Angle 6: 1.28104 degrees
Angle 7: 1.25297 degrees
Angle 8: 1.23096 degrees
Angle 9: 1.21323 degrees
Angle 10: 0.246809 degrees
Angle 11: -nan degrees

```

Task 5

VNS Lab 7 (завдання 2)

V – 21

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

```
C:\> gcc vns_lab_7_task_2_variant_1_solomia_liashchuk.cpp > main()
1 #include <iostream>
2 #include <cmath>
3 #include <vector>
4 #include <sstream>
5 #include "stdio.h"
6 using namespace std;
7
8 vector<int> NegativeNumbersElements(const vector<int>& array)
9 {
10     vector<int> negatives;
11     for (int num : array)
12     {
13         if (num < 0)
14         {
15             negatives.push_back(num);
16         }
17     }
18     return negatives;
19 }
20
21 int countWords(const string& str)
22 {
23     int count = 0;
24     istringstream stream(str);
25     string word;
26     while (stream >> word)
27     {
28         if (word.front() == word.back())
```



```

28         if (word.front() == word.back())
29             count++;
30     }
31     }
32 }
33
34 return count;
35 }
36
37 void NewFunction(const vector<int>& negativeNumbers);
38
39 int main()
40 {
41
42     vector<int> numbers = {1, 5, 7, 8, 16, 10, -15, -8, -32};
43     vector<int> negativeNumbers = NegativeNumbersElements(numbers);
44     for(int num : negativeNumbers)
45     {
46         cout << "Negative numders: " << num << endl;
47     }
48
49     NewFunction(negativeNumbers);
50
51     string text = "Львівська Політехніка один з найкращих ВНЗ України";
52     int result = countWords(text);
53     cout << "Words that start and finish with the same letter: " << result << endl;
54
55
56     return 0;
57 }
58
59 void NewFunction(const vector<int>& negativeNumbers)
60 {
61     cout << "Count of negative numbers: " << negativeNumbers.size() << endl;
62 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

Negative numders: -8
Negative numders: -32
Count of negative numbers: 3
Words that start and finish with the same letter: 0
PS C:\Users\olesi>

```

Task 6

Class Practice Task

```

C: > cpp > practice_work_task_1_solomia_liashchuk.cpp > main()
1  #include <iostream>
2  #include <vector>
3  #include <string>
4
5  using namespace std;
6
7  int main()
8  {
9      vector<string> books = {"Гаррі Поттер", "1984", "Інферно", "Сяйво", "Біблія", "Кобзар", "Колонія", "Зазирни в мої сни"};
10     vector<bool> availability(books.size(), true);
11     int choice;
12
13     do
14     {
15         main_menu:
16         cout << "\n--- Бібліотека ---\n";
17         cout << "1. Перерахувати всі книги\n";
18         cout << "2. Взяти книгу\n";
19         cout << "3. Повернути книгу\n";
20         cout << "4. Вийти\n";
21         cout << "Ваш вибір: ";
22         cin >> choice;
23
24         if (choice == 1)
25         {
26             cout << "Список книг в бібліотеці:\n";
27             for (size_t i = 0; i < books.size(); ++i)
28             {
29                 cout << i + 1 << ". " << books[i] << " - ";
30                 if (availability[i])
31                 {
32                     cout << "доступна\n";
33                 }
34                 else
35                 {
36                     cout << "взята\n";

```

```

36                     cout << "взята\n";
37                 }
38             }
39         }
40     }
41     else if (choice == 2)
42     {
43         string title;
44         bool found = false;
45         cout << "Введіть назву книги (без пробілів), яку хочете взяти: ";
46         cin >> title;
47
48         for (size_t i = 0; i < books.size(); ++i)
49         {
50             if (books[i] == title)
51             {
52                 found = true;
53                 if (availability[i])
54                 {
55                     availability[i] = false;
56                     cout << "Ви взяли книгу: " << title << endl;
57                 }
58                 else
59                 {
60                     cout << "На жаль, книга вже зайнята.\n";
61                 }
62                 break;
63             }
64         }
65         if (!found)
66         {
67             cout << "Книга не знайдена.\n";

```

```

67         cout << "Книга не знайдена.\n";
68     }
69
70 }
71 else if (choice == 3)
72 {
73     string title;
74     bool found = false;
75     cout << "Введіть назву книги (без пробілів), яку хочете повернути: ";
76     cin >> title;
77
78     for (size_t i = 0; i < books.size(); ++i)
79     {
80         if (books[i] == title)
81         {
82             found = true;
83             availability[i] = true;
84             cout << "Ви повернули книгу: " << title << endl;
85             break;
86         }
87     }
88     if (!found)
89     {
90         cout << "Книга не знайдена.\n";
91     }
92 }
93
94 else if (choice == 4)
95 {
96     cout << "Дякую за використання програми!\n";
97     break;
98 }
99 else

```

```

89     {
90         cout << "Книга не знайдена.\n";
91     }
92
93 }
94 else if (choice == 4)
95 {
96     cout << "Дякую за використання програми!\n";
97     break;
98 }
99 else
100 {
101     cout << "Неправильний вибір! Спробуйте ще раз.\n";
102     goto main_menu;
103 }
104
105 char continueChoice;
106 cout << "Виконати іншу операцію? (y/n): ";
107 cin >> continueChoice;
108 if (continueChoice != 'y') break;
109
110 }
111
112 while (true);
113
114 return 0;
115 }

```

Task 7

Self Practice Task

Я обрала задачу “Коля, Вася і теніс”, умова завдання була в тому щоб підрахувати кількість очок і підсумувати результати.

```
: > cpp > practice_work_self_algotester_tasks_solomia_liashchuk.cpp > ...
1  #include <iostream>
2  #include <string>
3  #include <cmath>
4  using namespace std;
5
6  int main()
7  {
8      int n;
9      cin >> n;
10     string c;
11     cin >> c;
12
13     int gamescoreK = 0;
14     int gamescoreV = 0;
15     int scoreK = 0;
16     int scoreV = 0;
17
18     for(char result : c)
19     {
20         if(result == 'K')
21         {
22             scoreK++;
23         }
24         else if(result == 'V')
25         {
26             scoreV++;
27         }
28
29         if ((scoreK >= 11 || scoreV >= 11) && abs(scoreK - scoreV) >= 2)
30         {
31             if (scoreK > scoreV)
32             {
33                 gamescoreK++;
34             }
35             else
```

```

35         else
36         {
37             gamescoreV++;
38         }
39
40         scoreK = 0;
41         scoreV = 0;
42     }
43 }
44
45 cout << gamescoreK << ":" << gamescoreV << endl;
46 if (scoreK > 0 || scoreV > 0)
47 {
48     cout << scoreK << ":" << scoreV << endl;
49 }
50
51 return 0;
52 }

```

Робота в команді:

Epic 3 ☆ 📁 Дошка

Фільтри S N & Поділитися

Done IVANNA	to-do Solomia	in progress Solomia	done Solomia
Ivanna Smachylo - 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 8 - Result Documentation Report and Outcomes Placement Activities (Docs and Programs on GitHub)	Epic 3 Task 3 - Lab# programming: VNS Lab 2
Ivanna Smachylo - Epic 3 Task 3 - Lab# programming: VNS Lab 2	Epic 3 Task 6 - Practice# programming: Class Practice Task	Epic 3 Task 1 - Theory Education Activities	Epic 3 Task 4 - Lab# programming: VNS Lab 3
Ivanna Smachylo - Epic 3 Task 4 - Lab# programming: VNS Lab 3	Epic 3 Task 9 - Results Evaluation and Release	+ Додати картку	Epic 3 Task 5 - Lab# programming: VNS Lab 7
Ivanna Smachylo - Epic 3 Task 8 - Result Documentation Report and Outcomes Placement Activities (Docs and Programs on GitHub)	+ Додати картку		Epic 3 Task 7 - Practice# programming: Self Practice Task
+ Додати картку			+ Додати картку

Смачило Іванна Ростиславівна

Лячук Софія Володимирівна

Висновок: в цьому епіку я дізналася багато нового і ще раз закріпила свої знання з різних тем. Набула нової практики в програмуванні та попрацювала в команді. Ці знання були для мене дуже корисні адже це базові навички для майбутнього в програмуванні.

