

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 ШІ-11
Вербицький Юрій Віталійович

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:

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/operatorsy-tsylu/>

<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_8XRRIus7Y&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=49

Виконання роботи:

1)Перелік завдань:

- 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

2)Умови завдань:

Task 3:

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

$$a_n = 10^{-n} (n-1)!$$

Task4: Обчислити функцію $f(x)$, використовуючи її розклад в степеневий ряд у двох випадках:

а) для заданого n ;

б) для заданої точності ε ($\varepsilon=0.0001$).

Для порівняння знайти точне значення функції.

$y = \frac{e^x + e^{-x}}{2}$	$0,1 \leq x \leq 1$	10	$S = 1 + \frac{x^2}{2!} + \dots + \frac{x^{2n}}{(2n)!}$
------------------------------	---------------------	----	---

Task5(1): Написати функцію `prost` зі змінною кількістю параметрів, що знаходить всі прості числа з декількох інтервалів. Інтервали задаються границями `A` і `B`. Написати викликаючу функцію `main`, що звертається до функції `prost` не менше трьох разів з кількістю параметрів 3, 5, 6.

Task5(2): Написати перевантажені функції й основну програму, що їх викликає.

а) для масиву цілих чисел знаходить середнє арифметичне;

б) для рядка знаходить кількість букв, що містяться в ньому.

Task6: 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.

Task7: Зуби

Обмеження: 2 сек., 256 MiB

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 nn зубів. Кожен зуб має коефіцієнт загостреності aia_i . Також існує межа загостреності kk . 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.

Вхідні дані

У першому рядку задані два цілих числа nn та kk — кількість зубів та межа загостреності відповідно.

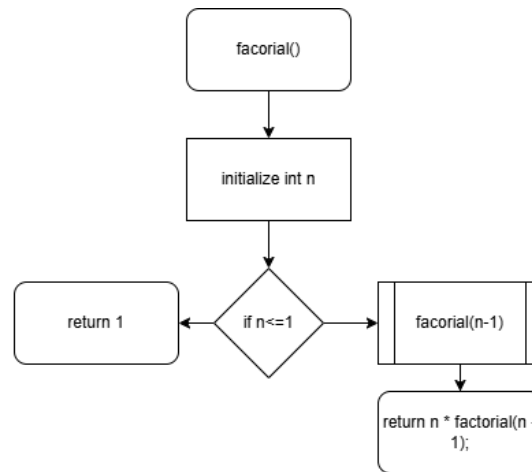
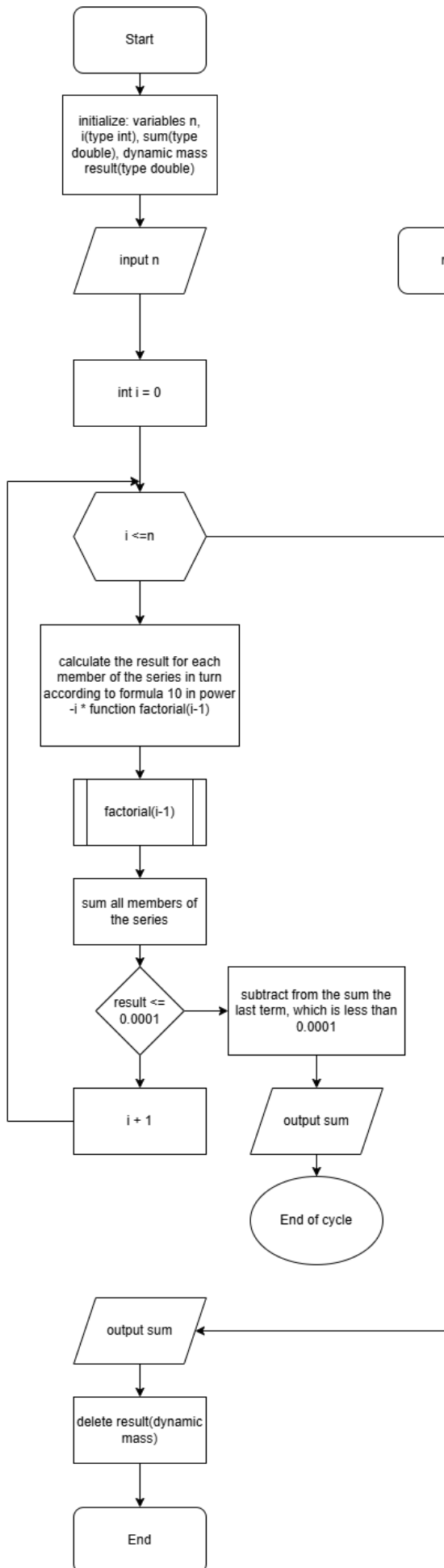
В другому рядку задано nn цілих чисел aia_i — коефіцієнти загостреності зубів.

Вихідні дані

Єдине ціле число — відповідь на задачу.

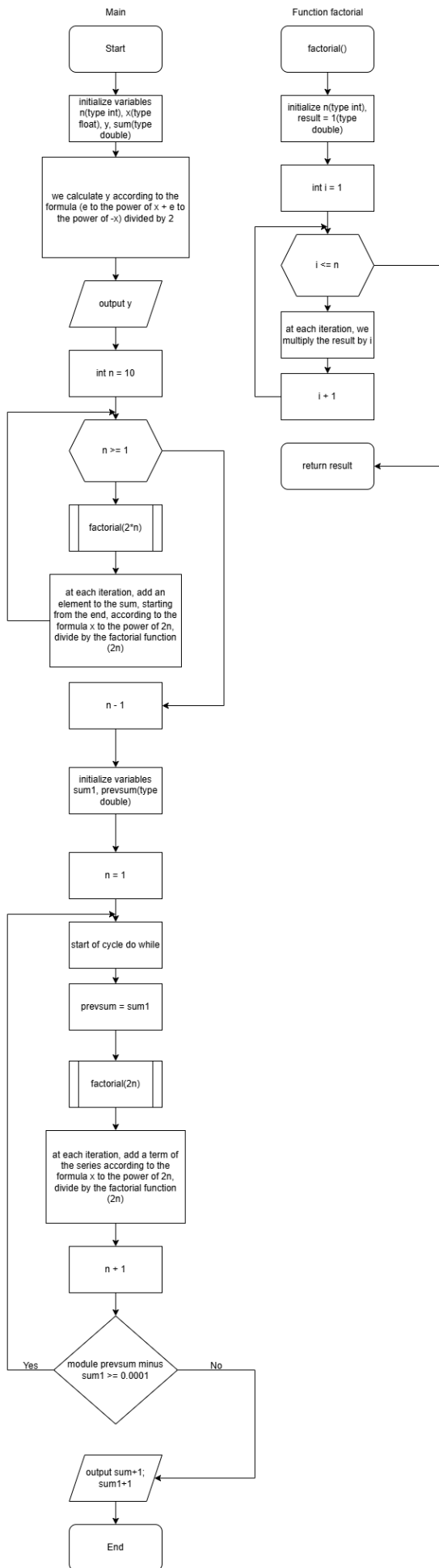
3)Дизайн та планова оцінка часу виконання завдань:

Task 3 - Lab# programming: VNS Lab 2 Варіант 17



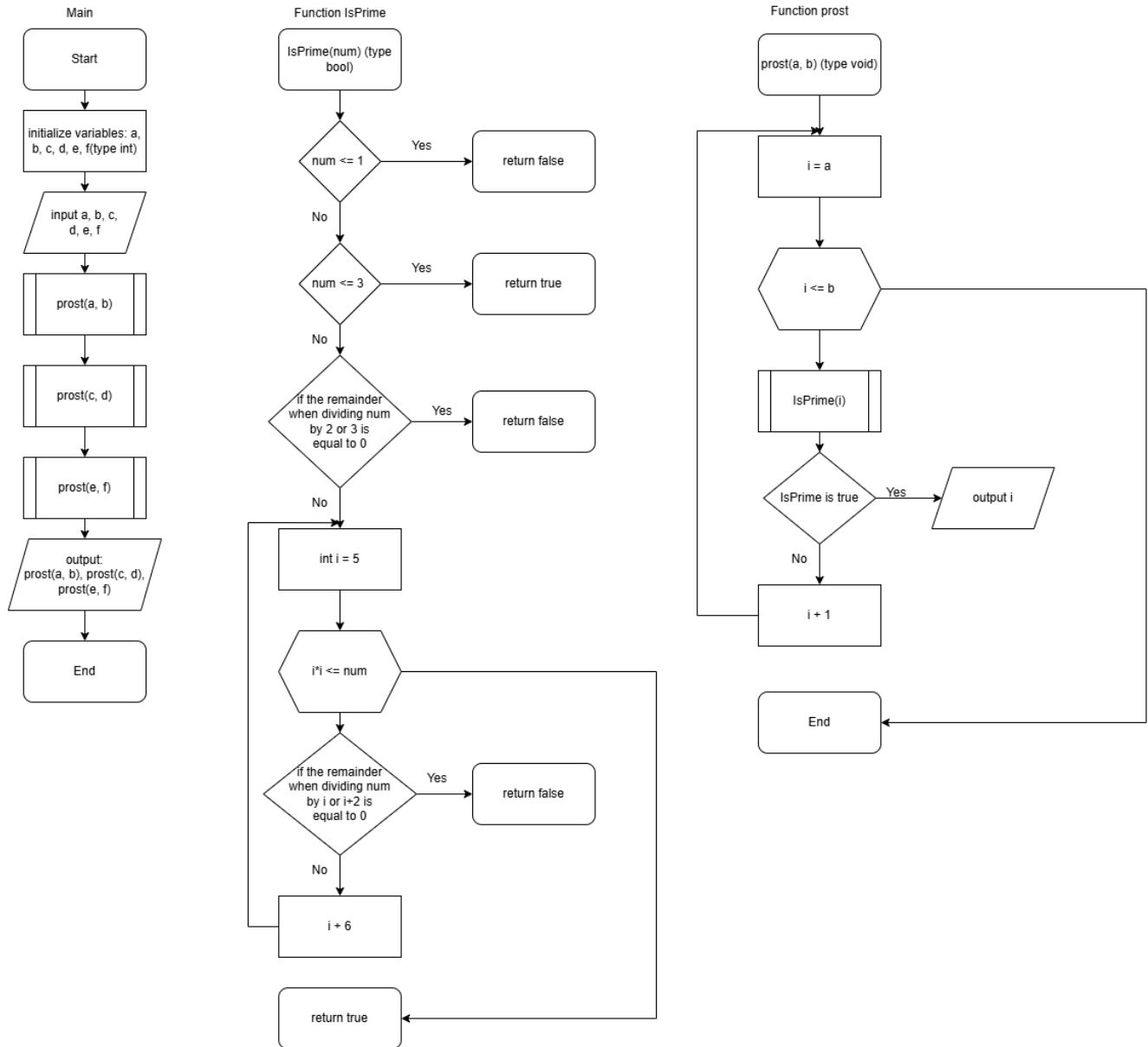
Орієнтований час виконання: 45 хв.

Task 4 - Lab# programming: VNS Lab 3 Варіант 17



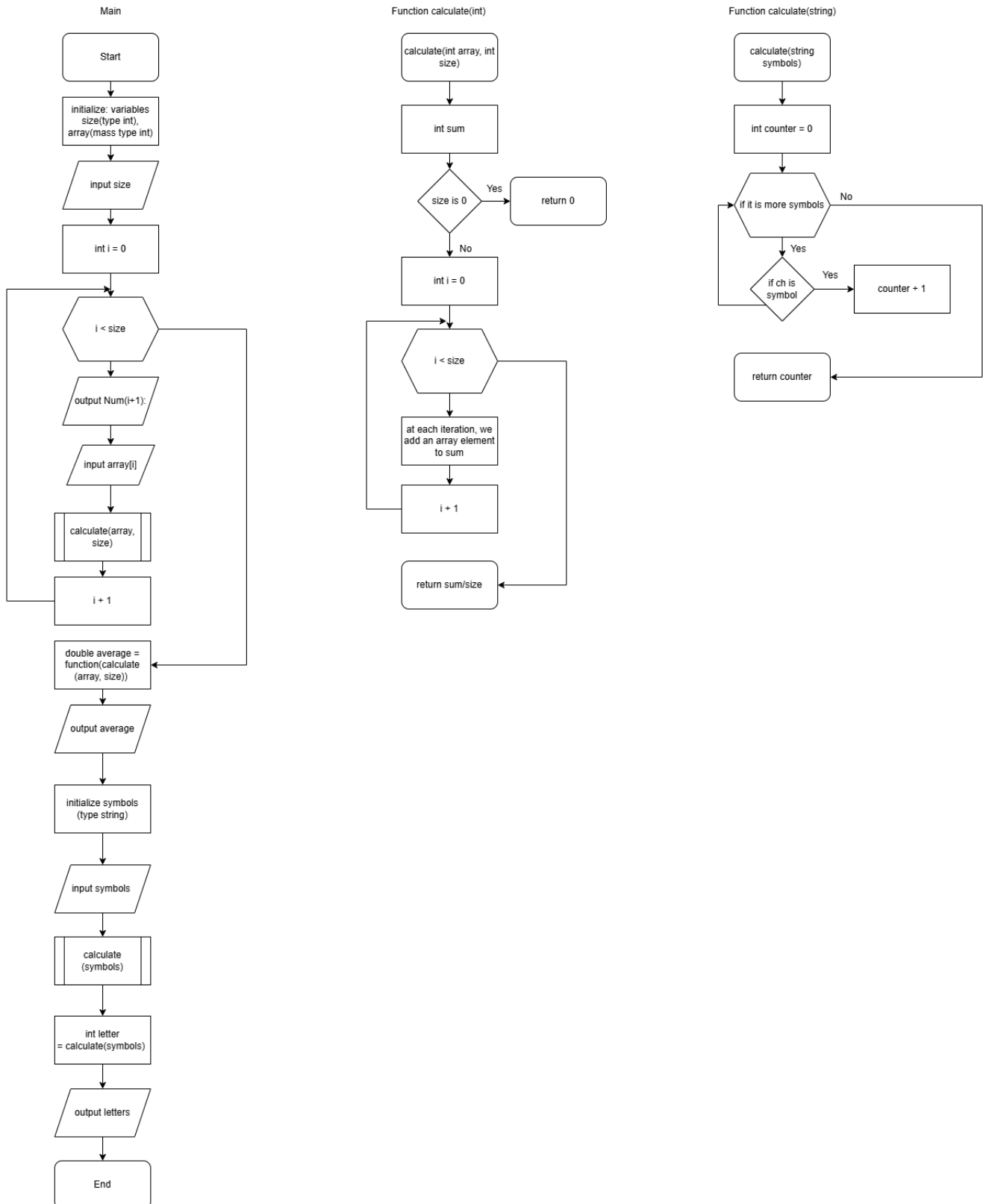
Орієнтовний час виконання: 1 година

Task 5(1) - Lab# programming: VNS Lab 7 task 1 Варіант 17



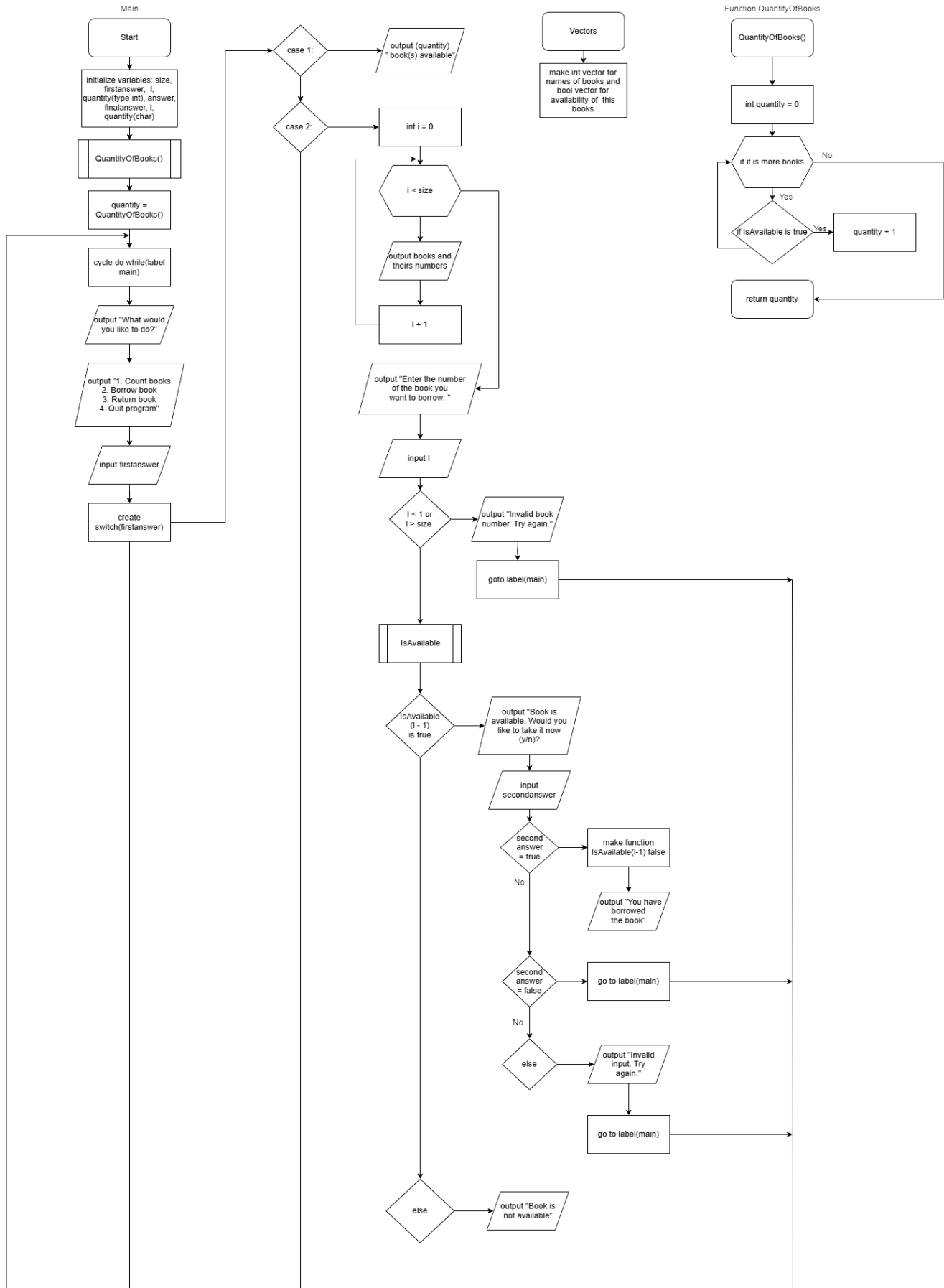
Орієтований час виконання: 1 год

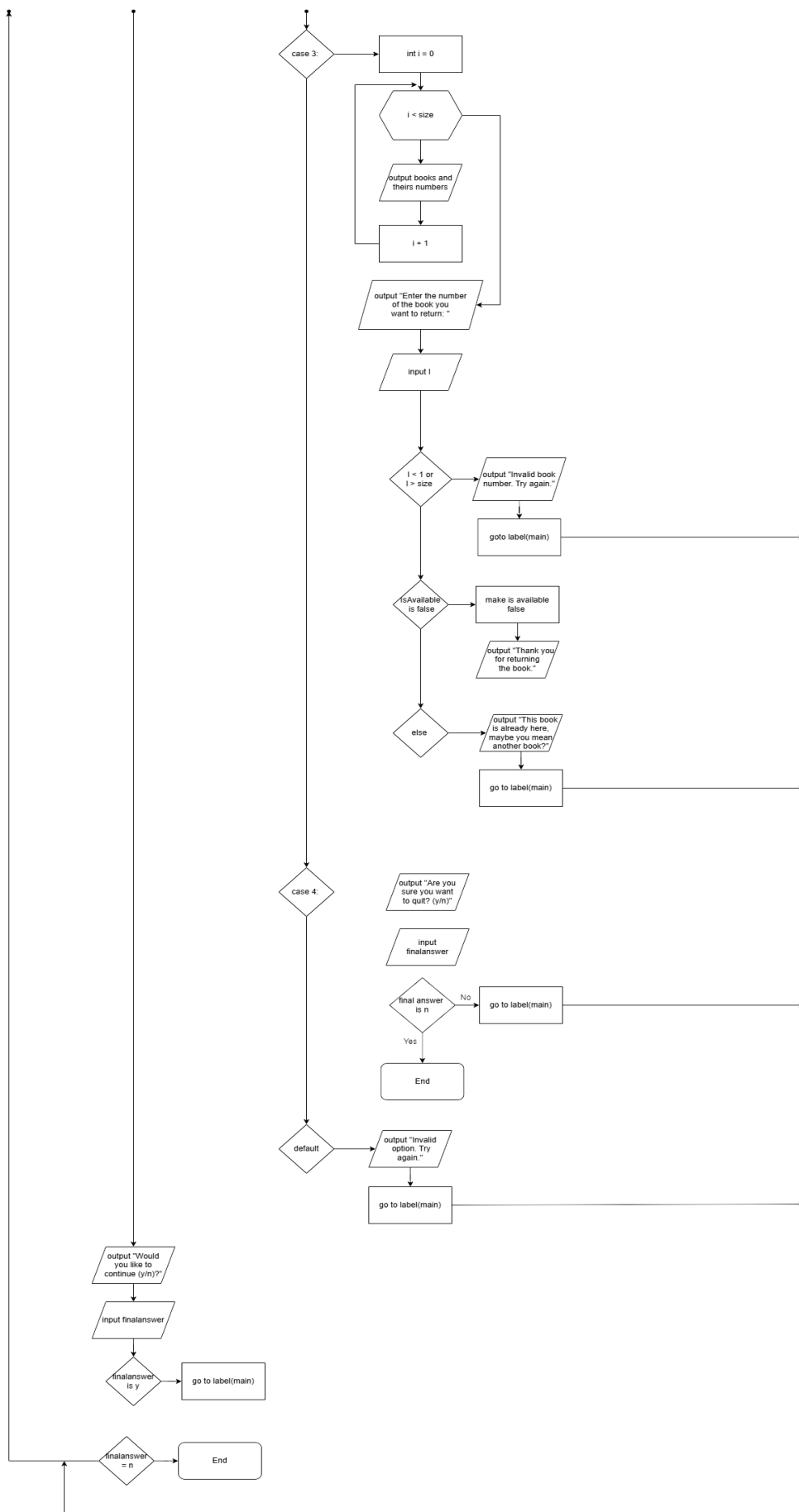
Task 5(1) - Lab# programming: VNS Lab 7 task 2 Варіант 17



Орієнтований час виконання: 1.5 год

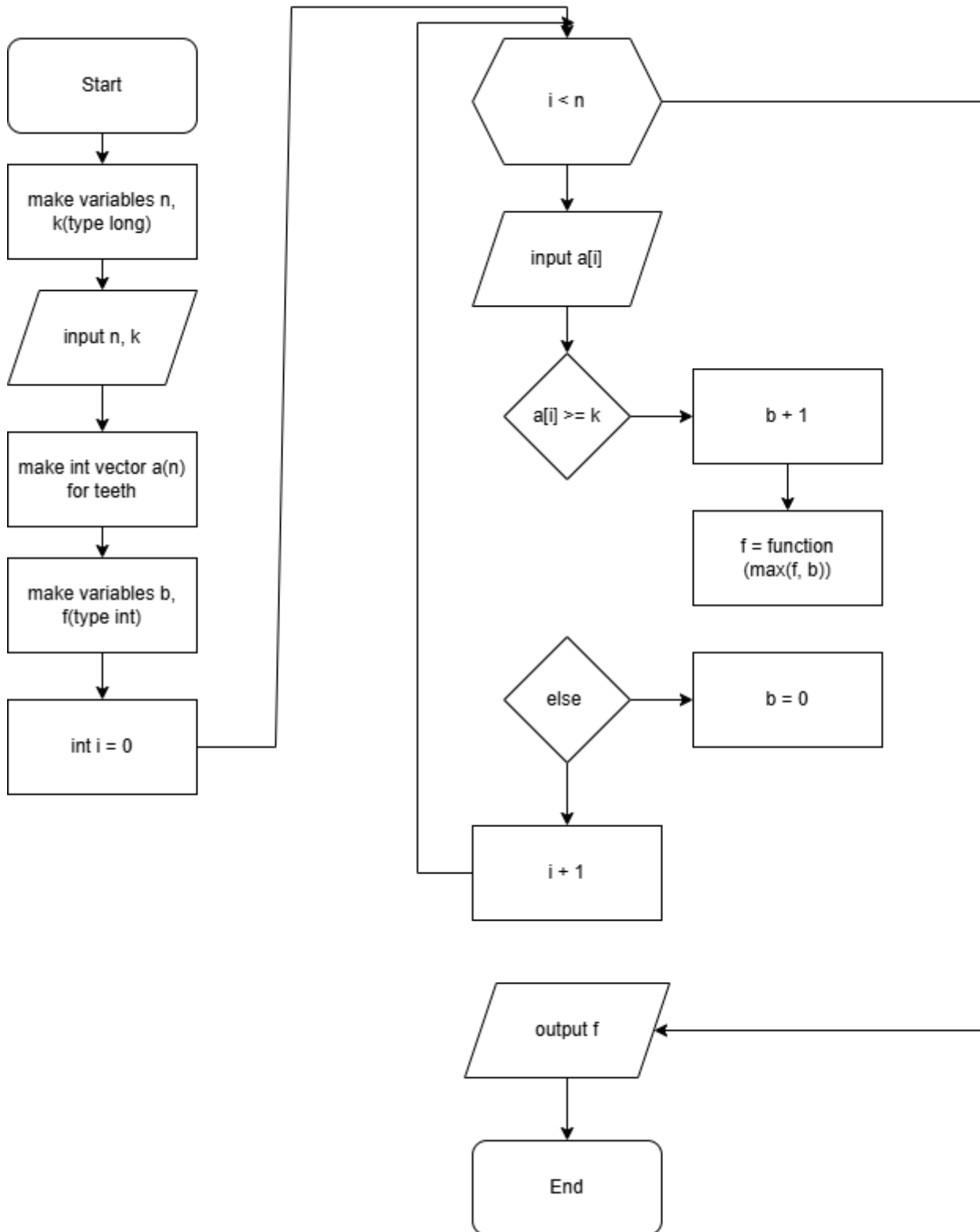
Task 6 - Practice# programming: Class Practice Task





Орієнтований час виконання 2.5 год

Task 7 - Practice# programming: Self Practice Task



Орієнтований час виконання 30 хв

4)Код програм:

Task 3 - Lab# programming: VNS Lab 2 Варіант 17

Код програми:

```
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6
7
8  double factorial(int n) {
9      if (n <= 1) {
10         return 1;
11     } else {
12         return n * factorial(n - 1);
13     }
14 }
15
16
17 int main() {
18
19     int n;
20     cin >> n;
21     int i;
22     double *result = new double[i];
23     double sum;
24
25     for(int i = 1; i <= n; i++){
26         result[i] = pow(10, -i) * factorial(i-1);
27         sum += result[i];
28
29         if(result[i] <= 0.0001){
30             sum -= result[i];
31             cout << sum<< endl;
32             break;
33         }
34     }
35
36     cout << sum;
37     delete[] result;
38     return 0;
39 }
```

Task 4 - Lab# programming: VNS Lab 3 Варіант 17

Код програми:

```
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  double factorial(int n){
7      double result = 1;
8      for(int i = 1; i <= n; i++){
9          result *= i;
10     }
11     return result;
12 }
13
14 int main(){
15     int n = 10;
16     float x = 0.9;
17     double y;
18     double sum;
19
20     y = (exp(x) + exp(-x))/2;
21     cout << y << endl;
22
23     for(int n = 10; n >= 1; n--){          // при n = 10
24         sum += (pow(x,2*n))/factorial(2*n);
25     }
26
27     double sum1;
28     double prevsum;
29     n = 1;
30     do{
31         prevsum = sum1;
32         sum1 += (pow(x,2*n))/factorial(2*n);
33         n++;
34     } while(abs(prevsum - sum1) >= 0.0001);
35
36
37     cout << sum+1 << endl;
38     cout << sum1+1 << endl;
39     return 0;
40 }
```

Task 5(1) - Lab# programming: VNS Lab 7 Task 1 Варіант 17

Код програми:


```

1  #include <iostream>
2  #include <conio.h>
3  #include <stdarg.h>
4  #include <cmath>
5
6  using namespace std;
7
8  // алгоритм ератосфена для вирахування простих чисел
9  bool isPrime(int num) {
10     if (num <= 1) return false;
11     if (num <= 3) return true;
12     if (num % 2 == 0 || num % 3 == 0) return false;
13
14     for (int i = 5; i * i <= num; i += 6) {
15         if (num % i == 0 || num % (i + 2) == 0)
16             return false;
17     }
18
19     return true;
20 }
21
22
23 void prost(int a, int b) {
24     for (int i = a; i <= b; ++i) {
25         if (isPrime(i)) {
26             cout << i << " ";
27         }
28     } cout << endl;
29 }
30
31
32 int main(){
33     // пари інтервалів
34     int a, b;
35     int c, d;
36     int e, f;
37
38     cin >> a >> b;
39     cin >> c >> d;

```

```

39     cin >> c >> d;
40     cin >> e >> f;
41
42
43     prost(a, b);
44     prost(c, d);
45     prost(e, f);
46     return 0;
47 }

```

Task 5(2) - Lab# programming: VNS Lab 7 Task 2 Варіант 17

Код програми:

```
1  #include <iostream>
2  #include <string>
3  #include <cctype>
4
5  using namespace std;
6
7
8  double calculate(int array[], int size){
9      int sum;
10     if(size == 0){
11         return 0;
12     }
13
14     for(int i = 0; i < size; i++){
15         sum += array[i];
16     }
17     return sum/size;
18 }
19
20
21 int calculate(const string &symbols) {
22     int counter = 0;
23     for (char ch : symbols) {
24         if (isalpha(ch)) { // Перевіряє, чи символ є літерою
25             ++counter;
26         }
27     }
28     return counter;
29 }
30
31
32 int main(){
33
34     int size;
35     cout << "Enter size of mass: ";
36     cin >> size;
37     int array[size];
38     for(int i = 0; i < size; i++){
39         cout << "Num" << i+1 << ": ";
```

```

40         cin >> array[i];
41     }
42     double average = calculate(array, size);
43     cout << average << endl;
44
45
46     string symbols;
47     cout << "Warning! Only one word, or phrase without space between words" << endl;
48     cout << "Enter symbols: ";
49     cin >> symbols;
50
51     int letters = calculate(symbols);
52     cout << letters << endl;
53
54     return 0;
55 }
56

```

Task 6 - Practice# programming: Class Practice Task

Код програми:

```

1  #include <iostream>
2  #include <vector>
3  #include <string>
4
5  using namespace std;
6
7  vector<string> Books = {"Charly", "Fedko", "Physics", "George", "Cats"};
8  vector<bool> IsAvailable = {true, true, true, true, true};
9
10 // Функція для підрахунку кількості доступних книг
11 int QuantityOfBooks() {
12     int quantity = 0;
13     for (bool isAvailable : IsAvailable) { // Для кожного елемента масиву IsAvailable
14         if (isAvailable) {
15             ++quantity;
16         }
17     }
18     return quantity;
19 }
20
21 int main() {
22     int size = Books.size();
23     int firstanswer;
24     char secondanswer;
25     char finalanswer;
26     int l; // Індекс книги
27     int quantity;
28     quantity = QuantityOfBooks();
29
30     do{
31         main: // Мітка для повернення до початку меню
32         cout << "What would you like to do?\n";
33         cout << "1. Count books\n2. Borrow book\n3. Return book\n4. Quit program\n";
34         cin >> firstanswer;
35
36         switch (firstanswer) {
37             case 1:
38                 cout << quantity << " book(s) available" << endl;
39                 break;

```

```

40
41     case 2:
42         for(int i = 0; i < size; i++){
43             cout << i+1 << ". " << Books[i] << endl;
44         }
45         cout << "Enter the number of the book you want to borrow: ";
46         cin >> l;
47         if (l < 1 || l > size) {
48             cout << "Invalid book number. Try again." << endl;
49             goto main;
50         }
51
52         if (IsAvailable[l - 1] == true) {
53             cout << "Book \"\" << Books[l - 1] << "\" is available. Would you like to take it now (y/n)? ";
54             cin >> secondanswer;
55             if (secondanswer == 'y') {
56                 IsAvailable[l - 1] = false;
57                 cout << "You have borrowed the book \"\" << Books[l - 1] << "\"." << endl;
58             } else if (secondanswer == 'n') {
59                 goto main;
60             } else {
61                 cout << "Invalid input. Try again." << endl;
62                 goto main;
63             }
64         } else {
65             cout << "Book \"\" << Books[l - 1] << "\" is not available." << endl;
66         }
67         break;
68
69     case 3:
70         for(int i = 0; i < size; i++){
71             cout << i+1 << ". " << Books[i] << endl;
72         }
73         cout << "Enter the number of the book you want to return: ";
74         cin >> l;
75         if (l < 1 || l > size) {

```

```

75         if (l < 1 || l > size) {
76             cout << "Invalid book number. Try again." << endl;
77             goto main;
78         }
79
80         if (IsAvailable[l - 1] == false) {
81             IsAvailable[l - 1] = true;
82             cout << "Thank you for returning the book \"\" << Books[l - 1] << "\"." << endl;
83         } else {
84             cout << "This book is already here, maybe you mean another book?" << endl;
85             goto main;
86         }
87         break;
88
89     case 4:
90         cout << "Are you sure you want to quit? (y/n)\n";
91         cin >> finalanswer;
92         if(finalanswer == 'n'){
93             goto main;
94         } else return 0;
95
96     default:
97         cout << "Invalid option. Try again." << endl;
98         goto main;
99 }
100 cout << "Would you like to continue (y/n)?" << endl;
101 cin >> finalanswer;
102 if (finalanswer == 'y'){
103     goto main;
104 }
105 } while(finalanswer != 'n');
106 return 0;
107 }

```

Task 7 - Practice# programming: Self Practice Task

Код програми:

```
1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4
5  using namespace std;
6
7
8  int main(){
9      long int n, k;
10     cin >> n >> k;
11
12     vector<long int> a(n);
13
14
15     int b = 0;
16     int f = 0;
17
18     for(int i = 0; i < n; i++){
19         cin >> a[i];
20         if(a[i] >= k){
21             ++b;
22             f = max(f, b);
23         } else{
24             b = 0;
25         }
26     }
27     cout << f << endl;
28     return 0;
29 }
```

5) Результати виконання завдань та фактично затрачений час

Task 3 - Lab# programming: VNS Lab 2 Варіант 17

```
4  
0.1126  
PS C:\Users\Yura\Desktop\c++>
```

```
5  
0.11284  
PS C:\Users\Yura\Desktop\c++>
```

```
6  
0.11296  
PS C:\Users\Yura\Desktop\c++>
```

```
7  
0.11296
```

```
100  
0.11296
```

Як ми бачимо, після $n = 6$, числа стають настільки малими, що є меншими за ϵ і програма їх не враховує, навіть при великих значеннях n .

Фактично затрачений час: 28 хв

Task 4 - Lab# programming: VNS Lab 3 Варіант 17

```
1.43309  
1.43309  
1.43309  
PS C:\Users\Yura\Desktop\c++>
```

3 значення функції ідентичні при $x = 0.9$

```
1.08107  
1.08107  
1.08107
```

При $x = 0.4$

```
1.02007
1.02007
1.02007
PS C:\Users\Yura\Desktop\c++>
```

Та при $x = 0.2$

Це означає, що програма працює справно.

Фактичний час затрачений на виконання: 50 хв

Task 5(1) - Lab# programming: VNS Lab 7 Task 1 Варіант 17

```
4 7
10 15
18 24
5 7
11 13
19 23
```

Перші три рядки ввід інтервалів, 3 наступні рядки знайдені на цих інтервалах прості числа.

Фактичний час затрачений на виконання: 1 год

Task 5(2) - Lab# programming: VNS Lab 7 Task 2 Варіант 17

```
Enter size of mass: 5
Num1: 4
Num2: 2
Num3: 3
Num4: 6
Num5: 10
5
Warning! Only one word, or phrase without space between words
Enter symbols: 3epictest
8
PS C:\Users\Yura\Desktop\c++>
```

Перевантежнням функції, і викликом її, ми знайшли середня арифметичне з чисел масиву, а також знайшли кількість в букв в рядку типу string, цифру 3 не було врахована, бо це не буква.

Фактичний час затрачений на виконання: 1 год

Task 6 - Practice# programming: Class Practice Task

```
What would you like to do?
1. Count books
2. Borrow book
3. Return book
4. Quit program
1
5 book(s) available
Would you like to continue (y/n)?
y
What would you like to do?
1. Count books
2. Borrow book
3. Return book
4. Quit program
2
1. Charly
2. Fedko
3. Physics
4. George
5. Cats
Enter the number of the book you want to borrow: 4
Book "George" is available. Would you like to take it now (y/n)? y
You have borrowed the book "George".
Would you like to continue (y/n)?
y
What would you like to do?
1. Count books
2. Borrow book
3. Return book
4. Quit program
2
1. Charly
2. Fedko
3. Physics
4. George
5. Cats
Enter the number of the book you want to borrow: 2
Book "Fedko" is available. Would you like to take it now (y/n)? n
```



```
What would you like to do?
1. Count books
2. Borrow book
3. Return book
4. Quit program
2
1. Charly
2. Fedko
3. Physics
4. George
5. Cats
Enter the number of the book you want to borrow: 4
Book "George" is not available.
Would you like to continue (y/n)?
y
What would you like to do?
1. Count books
2. Borrow book
3. Return book
4. Quit program
3
1. Charly
2. Fedko
3. Physics
4. George
5. Cats
Enter the number of the book you want to return: 4
Thank you for returning the book "George".
Would you like to continue (y/n)?
y
What would you like to do?
1. Count books
2. Borrow book
3. Return book
4. Quit program
4
Are you sure you want to quit? (y/n)
y
```

Фактичний час виконання: 2 год

Task 7 - Practice# programming: Self Practice Task

```
hener.exe -s stdin=Microsoft-MIEngine-1
r-kehkgdwn.dw2' '--pid=Microsoft-MIEngine
6
2
1
3
2
4
5
1
4
PS C:\Users\Yura\Desktop\c++>
```

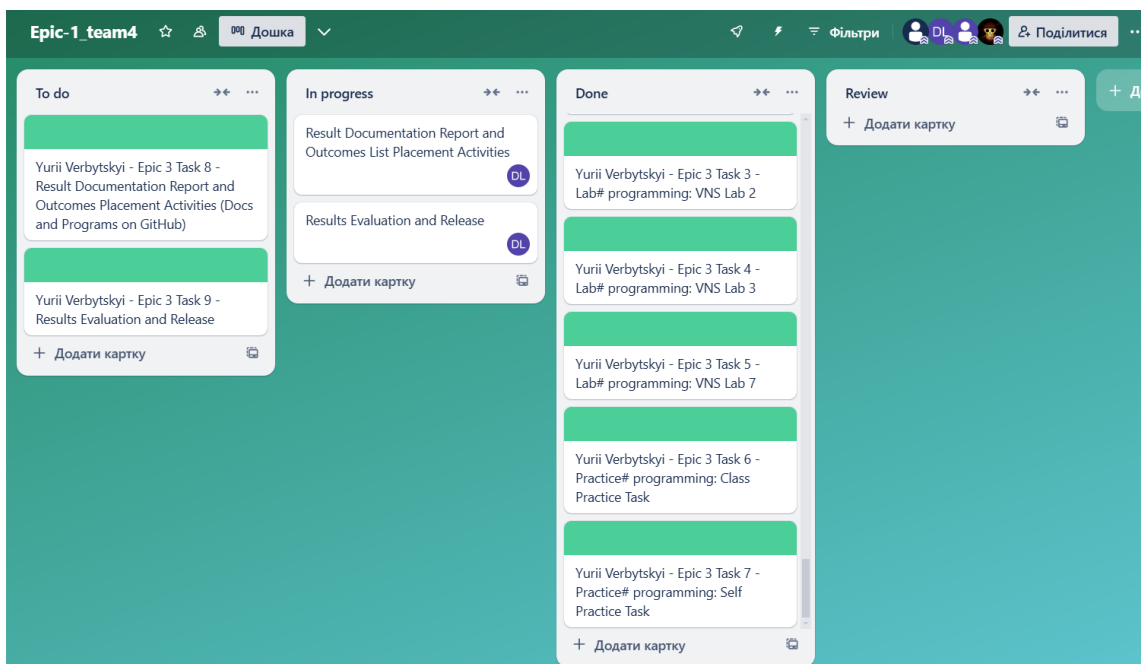
Алготестер:

Створено	Задача	Компілятор	Результат	Час (сек.)	Пам'ять (МіБ)	№
8 годин тому	0182 - Зуби	C++ 23	Зараховано	0.043	2.055	1859089

Фактичний час виконання: 45 хв

б)Робота з комадою

Trello:



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 (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.

Посилання на пул реквест: