Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту

A blue and white logo

Description automatically generated

**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

Станько Олег Ігорович

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:**

Ознайомитися з основами циклів, дослідити концепцію функцій у C++, навчитися зображати функції в блок схемах, розібратися зі способами завершення виконання циклів.

**Теоретичні відомості:**

Тема №1: Введення в Цикли та їх Види в С++:

Тема №2: Управління Виконанням Циклів:

Тема №3: Вкладені Цикли:

Тема №4: Основи Функцій у С++:

Тема №5: Перевантаження Функцій та Простір Імен:

Тема №6: Розширені Можливості Функцій:

Тема №7: Вбудовані Функції в С++:

**Індивідуальний план опрацювання теорії:**

Тема №1: Введення в Цикли та їх Види в С++:

Вивчив такі види циклів: for, while, do-while.

[C++. Цикли. Оператори циклу for, while, do…while | BestProg](https://www.bestprog.net/uk/2017/09/04/cycles-operators-of-the-cycle-for-while-do-while_ua/#:~:text=%D0%A3%20%D0%BC%D0%BE%D0%B2%D1%96%20C%2B%2B%20%D1%96%D1%81%D0%BD%D1%83%D1%94%203%20%D0%B2%D0%B8%D0%B4%D0%B8%20%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80%D1%96%D0%B2%20%D1%86%D0%B8%D0%BA%D0%BB%D1%83,%D1%86%D0%B8%D0%BA%D0%BB%20while%20%D0%B7%20%D0%BF%D0%B5%D1%80%D0%B5%D0%B4%D1%83%D0%BC%D0%BE%D0%B2%D0%BE%D1%8E%3B%20%D1%86%D0%B8%D0%BA%D0%BB%20do%E2%80%A6while%20%D0%B7%20%D0%BF%D0%BE%D1%81%D1%82%D1%83%D0%BC%D0%BE%D0%B2%D0%BE%D1%8E.)

Витратив 10 хвилин

Тема №2: Управління Виконанням Циклів:

Ознайомився з exit(),goto,break.

[**Оператори управління потоком виконання програм в С++ / aCode**](https://acode.com.ua/urok-66-operatory-upravlinnya-potokom-vykonannya-program/)

Витратив 5 хвилин

Тема №3: Вкладені Цикли:

Ознайомився з вкладеними циклами.

[C++ ⦁ Теорія ⦁ Урок 38 ⦁ Вкладені конструкції](https://www.youtube.com/watch?v=mBPHKQx21eE&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)

Витратив 7 хвилин

Тема №4: Основи Функцій у С++:

Ознайомився з функціями, зрозумів як їх зображати в блок схемах

[Функції в С++ / aCode](https://acode.com.ua/urok-15-funktsiyi-i-operator-return/)

Витратив 20 хвилин

Тема №5: Перевантаження Функцій та Простір Імен:

Ознайомився з перевантаженням функцій.

[Перевантаження функцій в С++ / Уроки по С++ / aCode](https://acode.com.ua/urok-108-perevantazhennya-funktsij/)

Витратив 7 хвилин

Тема №6: Розширені Можливості Функцій:

Ознайомився з функцією зі змінною кількістю параметрів, областю видимості функції – static, extern, рекурсією, передачею масивів та об'єктів як параметрів, поверненням масивів та об'єктів з функцій.

[**Еліпсис в С++ / Уроки по С++ / aCode**](https://acode.com.ua/urok-117-elipsys/)

Витратив 2 години

Тема №7: Вбудовані Функції в С++:

Ознайомився з вбудованими функціями

[Вбудовані функції в С++ / Уроки по С++ / aCode](https://acode.com.ua/urok-107-vbudovani-funktsiyi/#:~:text=%D0%9F%D1%80%D0%B8%20%D0%BA%D0%BE%D0%BC%D0%BF%D1%96%D0%BB%D1%8F%D1%86%D1%96%D1%97%20%D0%B2%D0%B0%D1%88%D0%BE%D0%B3%D0%BE%20%D0%BA%D0%BE%D0%B4%D1%83%2C%20%D0%B2%D1%81%D1%96%20%D0%B2%D0%B1%D1%83%D0%B4%D0%BE%D0%B2%D0%B0%D0%BD%D1%96%20%D1%84%D1%83%D0%BD%D0%BA%D1%86%D1%96%D1%97%20%28%D0%B0%D0%BD%D0%B3%D0%BB.,%D0%B1%20%D0%B1%D1%83%D1%82%D0%B8%20%D0%B2%D0%B8%D1%82%D1%80%D0%B0%D1%87%D0%B5%D0%BD%D1%96%20%D0%BD%D0%B0%20%D0%B2%D0%B8%D0%BA%D0%BB%D0%B8%D0%BA%20%D1%86%D1%96%D1%94%D1%97%20%D1%84%D1%83%D0%BD%D0%BA%D1%86%D1%96%D1%97%2C%20%D0%B7%D0%B1%D0%B5%D1%80%D1%96%D0%B3%D0%B0%D1%8E%D1%82%D1%8C%D1%81%D1%8F%21)

Витратив 1 годину

**Виконання роботи:**

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

**VNS LAB 2**

Використовуючи оператор циклу, знайти суму елементів, зазначених у конкретному варіанті. Результат надрукувати, надавши відповідний заголовком.

Знайти суму ряду з точністю ε=0.0001, загальний член якого

**VNS LAB 3**

Для х, 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 (ε=0.0001). Для порівняння знайти точне значення функції

|  |  |  |  |
| --- | --- | --- | --- |
| функція | Діапазон зміни аргументу | n | Сума |
|  |  | 50 |  |

**VNS LAB 7 TASK 1**

Написати функцію nok зі змінною кількістю параметрів, що знаходить найменше спільне кратне для декількох чисел. (НСД- найбільший спільний дільник) Написати викликаючу функцію main, що звертається до функції nok не менше трьох разів з кількістю параметрів 3, 5, 6.

**VNS LAB 7 TASK 2**

Написати перевантажені функції й основну програму, що їх викликає.

а) для масиву цілих чисел знаходить максимальний елемент;

б) для рядка знаходить довжину найдовшого слова .

**practice work**

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.

Мета Задачі

Навчитися користуватися операторами циклів та функцією переходу на мітку:

1. for( ) { … }
2. for each
3. while( ) { … }
4. do { … } while( )
5. go to

Вимоги:

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

**Self-practice task**

Останнім часом Зеник полюбив викладати портрети визначних історичних постатей України, а також крилаті фрази із ... незвичних матеріалів. Портрет можна уявити як прямокутну сітку розміру n на m, де клітинки позначені «.», якщо вони повинні бути порожніми і «\*», якщо у них повинен лежати ... незвичний матеріал. Оскільки один шматок незвичного матеріалу досить важкий (в середньому кілограмів 74), Зеник підвозить матеріали вантажівкою.

За один заїзд вантажівки Зеник може підвезти від 1 до k шматків матеріалу. Вантажівка досить неповоротка, тому за один заїзд здатна підвозити матеріали лише до одного рядка. Тобто перед тим, як вивантажити чергові матеріали, Зеник вибирає один рядок сітки і вивантажує всі матеріали саме в тому рядку. Тоді Зеник знову їде за матеріалами, знову вибирає рядок і знову вивантажує всі матеріали у вибраному рядку і так доти не викладе, те що він початково запланував.

Скільки поїздок вантажівкою необхідно Зенику, щоб викласти портрет?

**Вхідні дані**

У першому рядку задано три числа n, m і k — розміри сітки і кількість незвичних матеріалів, які Зеник здатен перевести за один раз у один рядок.

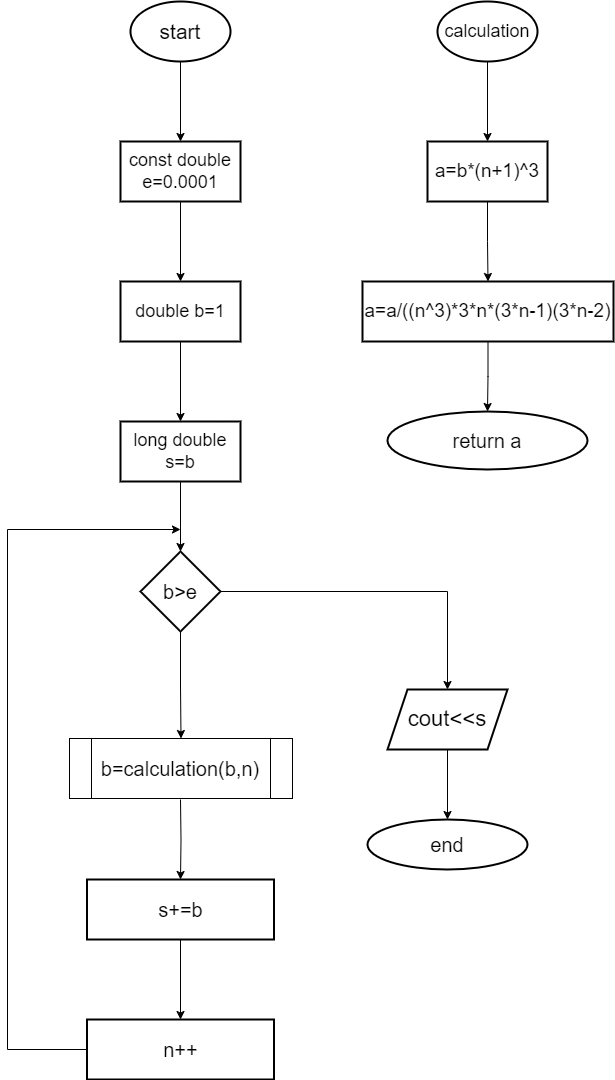
У кожному з наступних n рядків задано сітку, яка складається із символів «.» i «\*».

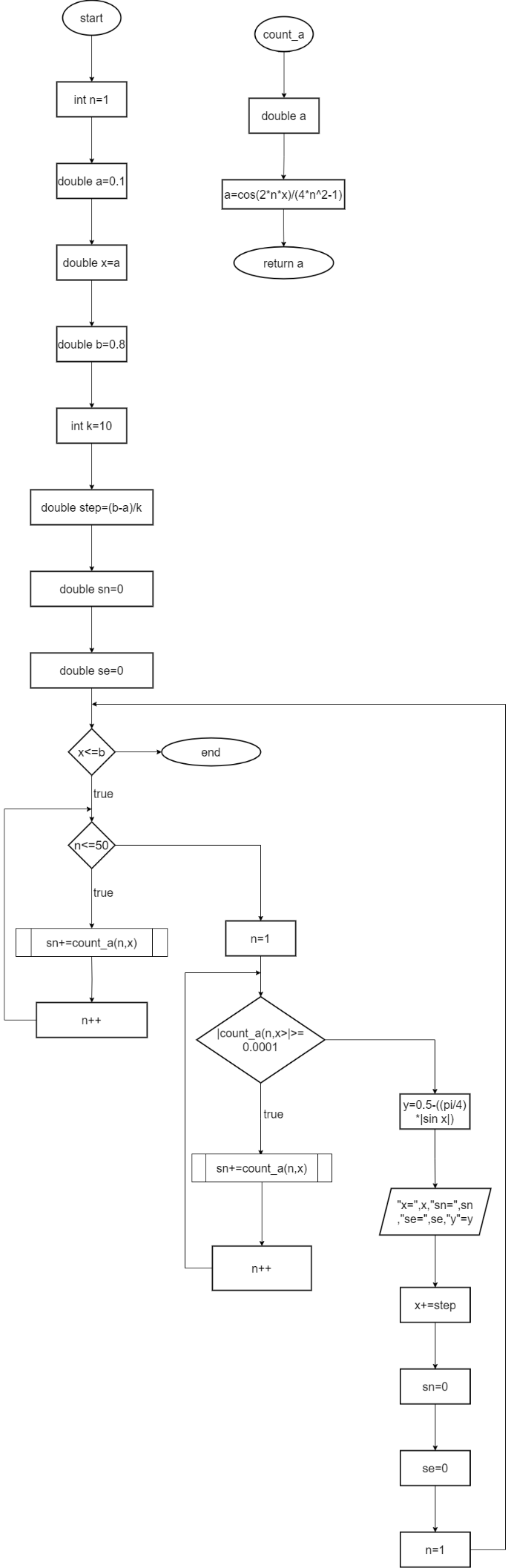
**Вихідні дані**

Виведіть одне число — мінімальну кількість поїздок вантажівки, яку повинен здійснити Зеник, для того, щоб викласти портрет.

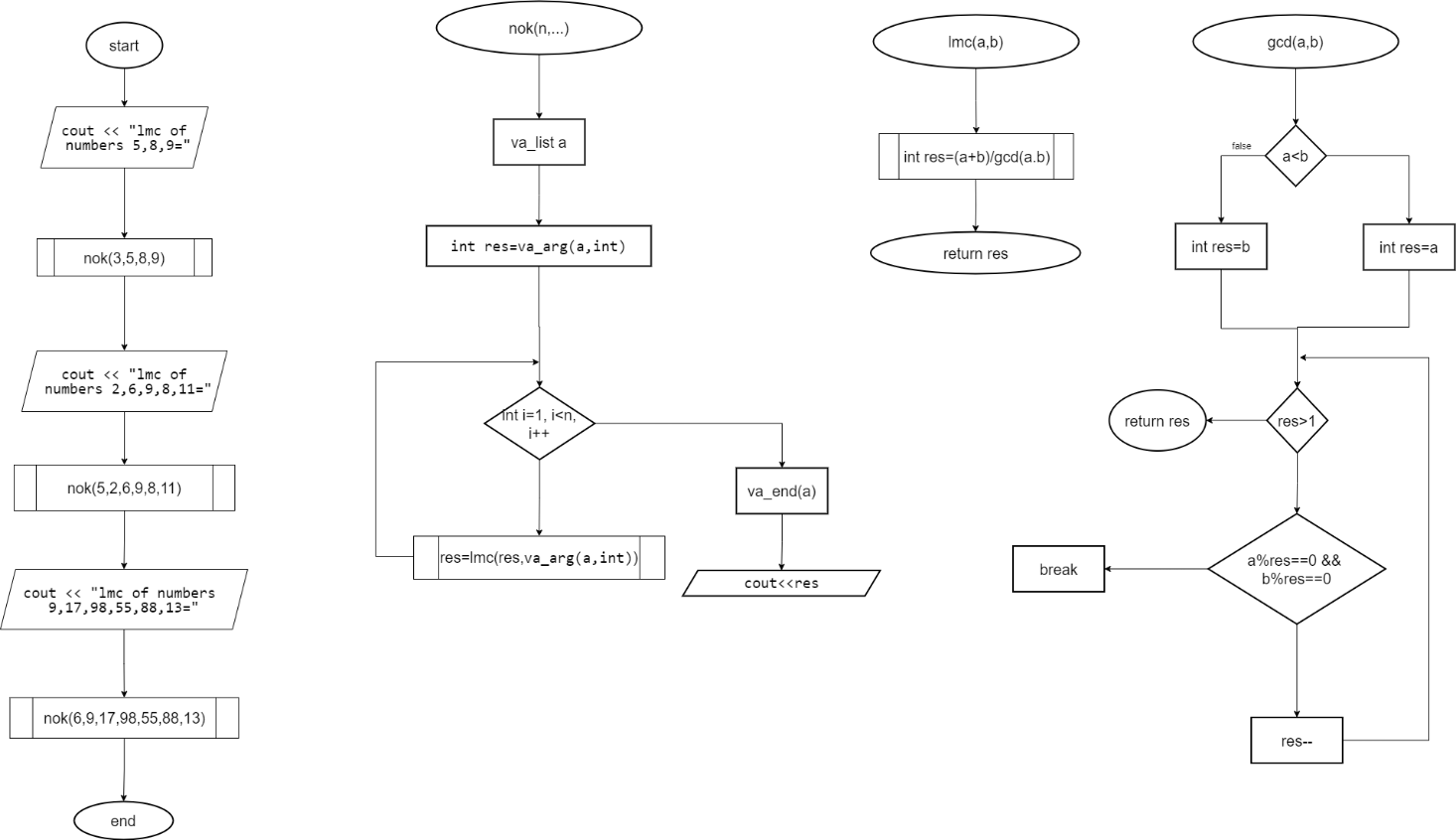
2. Блок-схеми

**VNS LAB 2**

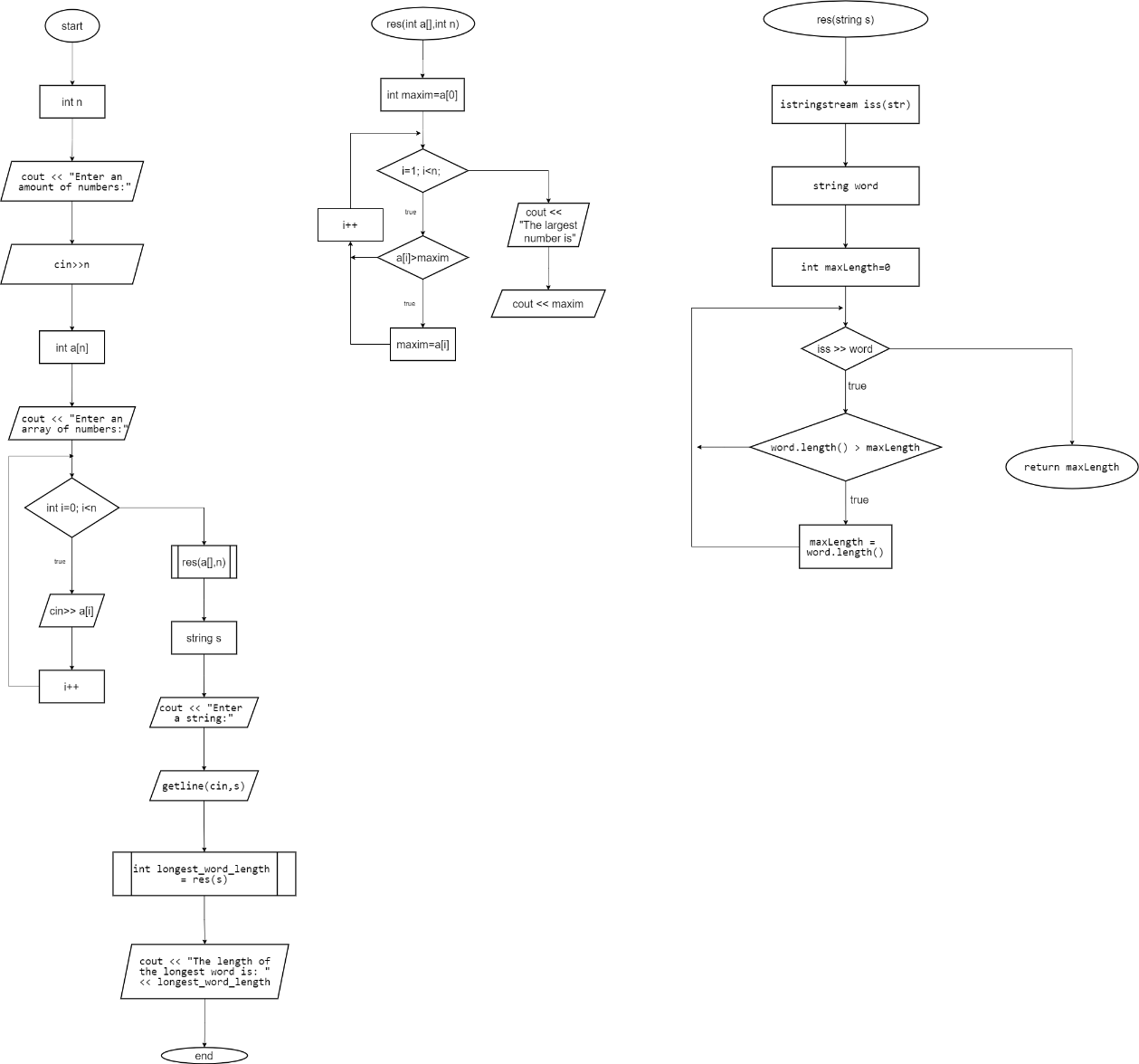


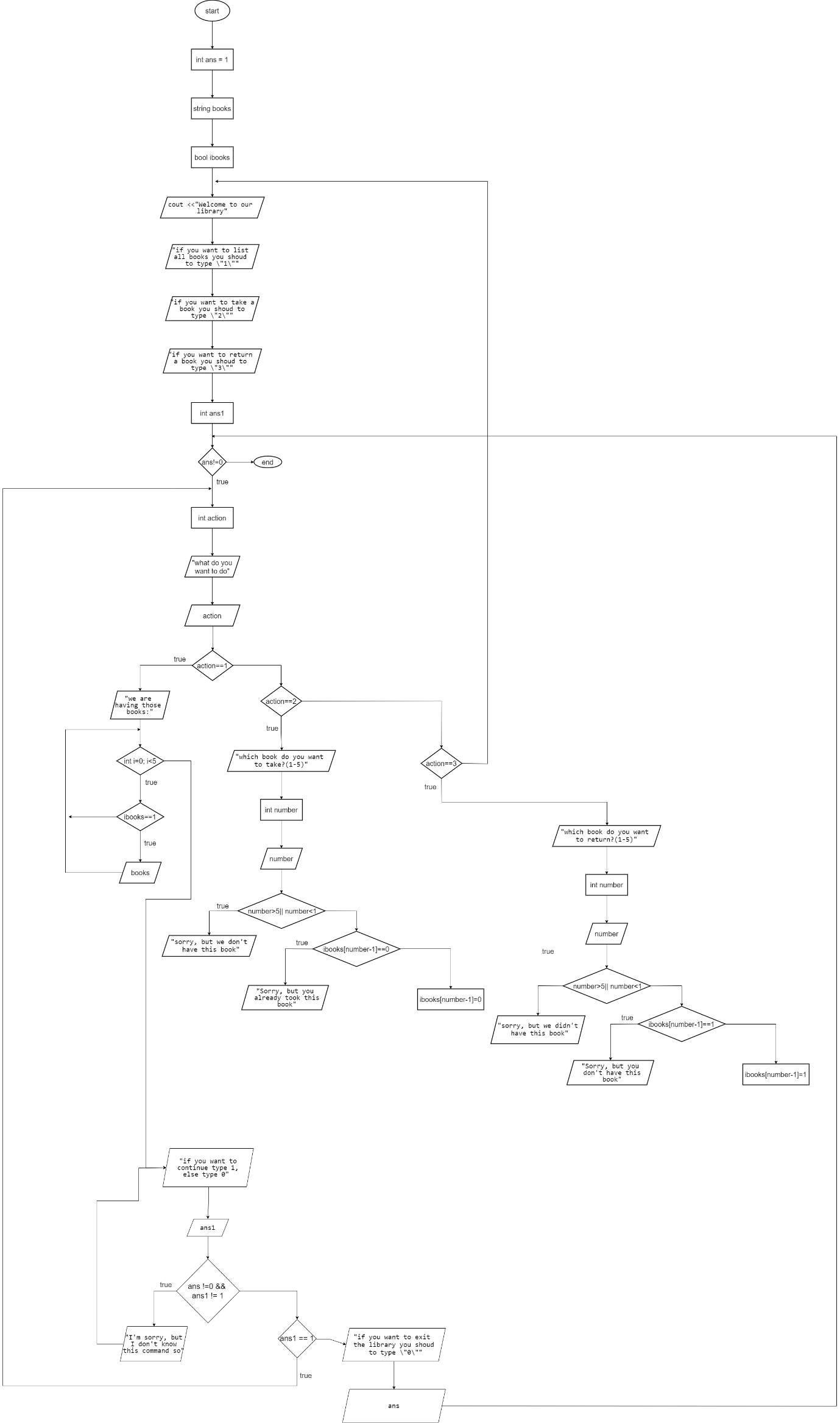
**VNS LAB 3**

**VNS LAB 7 TASK 1**

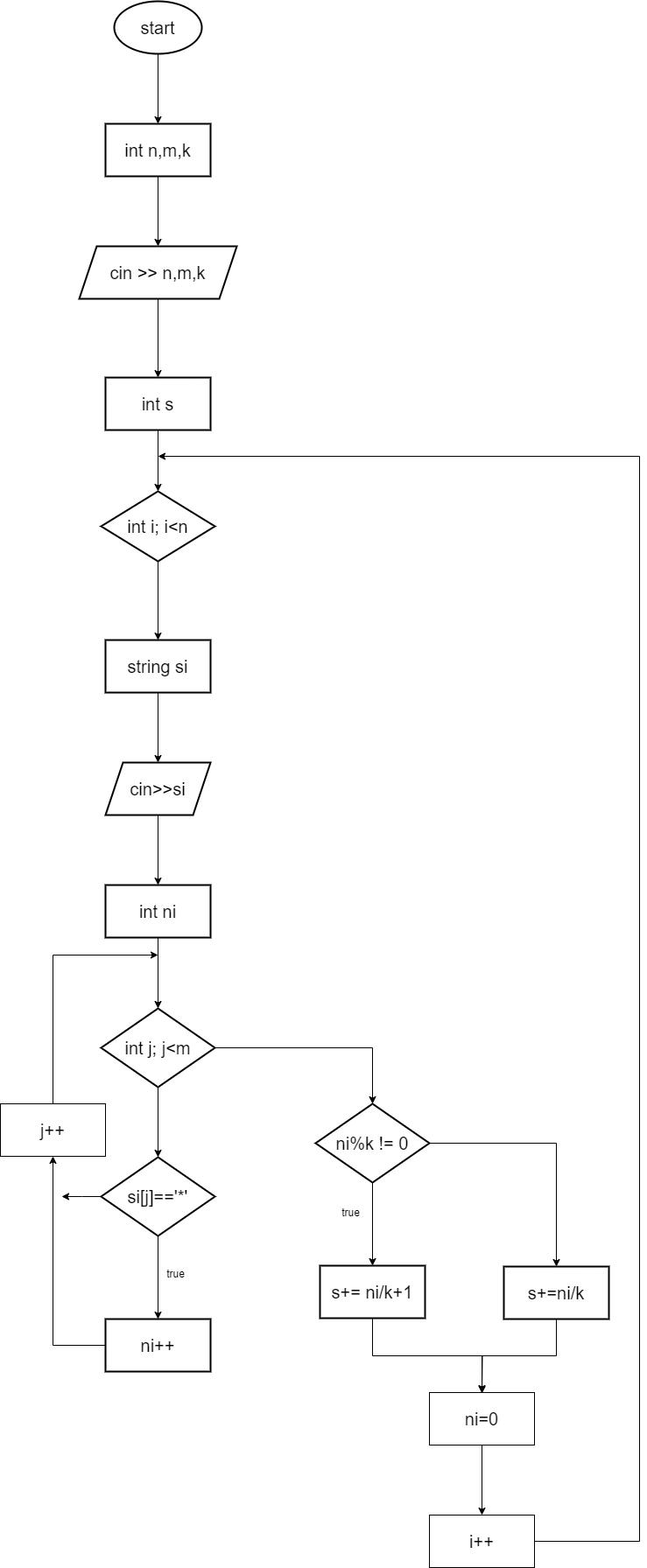
****

**VNS LAB 7 TASK 2**



 **practice work**

**Self-practice task**

****

Коди:

**VNS LAB 2**

#include <iostream>

#include <math.h>

using namespace std;

double calculation(double b, int n)

{

    double a;

    a=b\*pow(n+1,3)/(pow(n,3)\*3\*n\*(3\*n-1)\*(3\*n-2));

    return a;

}

int main()

{

    const double e=0.0001;

    double b=1;

    int n=1;

    long double s=b;

    while (b>e)

    {

    b=calculation(b,n);

    s+=b;

    n++;

    }

    cout<<s<<"\n";

    return 0;

}

**VNS LAB 3**

#include <stdio.h>

#include <iostream>

#include <cmath>

using namespace std;

double count\_a(int n, double x)

{

    double a =cos(2\*n\*x)/(4\*n\*n-1);

    return a;

}

int main()

{

    int n=1;

    double a=0.1;

    double x=a;

    double b=0.8;

    int k=10;

    double step=(b-a)/k;

    double sn=0;

    double se=0;

    double y;

    while (x<=b+0.00000001)

    {

        while (n<=50)

        {

            sn+=count\_a(n,x);

            n++;

        }

        n=1;

        while (fabs(count\_a(n,x))>=0.0001)

        {

            se+=count\_a(n,x);

            n++;

        }

        y=0.5-M\_PI/4\*fabs(sin(x));

        printf("x= %lf \t sn= %lf \t se= %lf \t y= %lf \n", x, sn, se, y);

        x+=step;

        sn=0;

        se=0;

        n=1;

    }

    return 0;

}

**VNS LAB 7 TASK 1**

#include <iostream>

#include <cstdarg>

#include <math.h>

using namespace std;

int gcd(int a,int b)

{

    int res;

    if (a<b)

    {

        res=a;

    }

    else

    {

        res=b;

    }

    while (res > 1) {

        if (a % res == 0 && b % res == 0)

            break;

        res--;

    }

    return res;

}

int lmc(int a, int b)

{

    return((a\*b)/gcd(a,b));

}

int nok(int n,...)

{

    va\_list a;

    va\_start(a,n);

    int res= va\_arg(a, int);

    for (int i=1; i<n;i++)

    {

        res=lmc(res, va\_arg(a, int));

    }

    va\_end(a);

    return res;

}

int main()

{

    cout << "lmc of numbers 5,8,9="<<nok(3,5,8,9)<<"\n";

    cout << "lmc of numbers 2,6,9,8,11="<<nok(5,2,6,9,8,11)<<"\n";

    cout << "lmc of numbers 9,17,98,55,88,13="<<nok(6,9,17,98,55,88,13)<<"\n";

    return 0;

}

**VNS LAB 7 TASK 2**

#include <iostream>

#include <sstream>

using namespace std;

int res(string& str)

{

    istringstream iss(str);

    string word;

    int maxLength = 0;

    while (iss >> word)

    {

        if (word.length() > maxLength)

        {

            maxLength = word.length();

        }

    }

    return maxLength;

}

void res(int a[],int n)

{

    int maxim = a[0];

    for (int i = 1; i < n; i++)

    {

        if (a[i] > maxim)

        {

            maxim = a[i];

        }

    }

    cout << "The largest number is " << maxim << endl;

}

int main()

{

    int n;

    cout << "Enter an amount of numbers: \n";

    cin >> n;

    int a[n];

    cout << "Enter an array of numbers: \n";

    for (int i = 0; i < n; i++)

    {

        cin >> a[i];

    }

    res(a,n);

    string s;

    cout << "Enter a string: \n";

    cin.ignore();

    getline(cin, s);

    int longest\_word\_length = res(s);

    cout << "The length of the longest word is: " << longest\_word\_length << endl;

    return 0;

}

**practice work**

#include <iostream>

using namespace std;

int main()

{

    int ans=1;

    string books[5]={"Poliana","The C++ Programming Language","Measure, Integration & Real Analysis","Journey into Discrete Mathematics","English grammar"};

    bool ibooks[5]={1,1,1,1,1};

    menu:

    cout <<"Welcome to our library\n";

    cout <<"if you want to list all books you shoud to type \"1\" \n";

    cout <<"if you want to take a book you shoud to type \"2\" \n";

    cout <<"if you want to return the book you shoud to type \"3\" \n";

    int ans1;

    while (ans != 0)

    {

        do

        {

            int action;

            cout <<"what do you want to do\n";

            cin >> action;

            if (action==1)

            {

                cout << "we are having those books:\n";

                for(int i=0;i<5;i++)

                {

                    if (ibooks[i]==1)

                    {

                        cout << books[i] << "\n";

                    }

                }

            }

            else if (action==2)

            {

                cout << "which book do you want to take?(1-5)\n";

                int number;

                cin >> number;

                if (number>5 || number<1)

                {

                    cout << " sorry, but we don't have this book\n";

                }

                else

                {

                    if (ibooks[number-1]==0)

                    {

                        cout << "Sorry, but you already took this book\n";

                    }

                    else

                    {

                        ibooks[number-1]=0;

                    }

                }

            }

            else if (action==3)

            {

                cout << "which book do you want to return the book?(1-5)\n";

                int number;

                cin >> number;

                if (number>5 || number<1)

                {

                    cout << " sorry, but we didn't have this book\n";

                }

                else

                {

                    if (ibooks[number-1]==1)

                    {

                        cout << "Sorry, but you don't have this book\n";

                    }

                    else

                    {

                        ibooks[number-1]=1;

                    }

                }

            }

            else

            {

                goto menu;

            }

            question:

            cout<<"if you want to continue type 1, else type 0 \n";

            cin>>ans1;

            if (ans1 != 1 && ans1 != 0)

            {

                cout << "I'm sorry, but I don't know this command\n so ";

                goto question;

            }

        }while (ans1==1);

    cout <<"if you want to exit the library you shoud to type \"0\" \n";

    cin >> ans;

    };

    return 0;

}

**Self-practice task**

#include <iostream>

using namespace std;

int main()

{

    int n, m, k;

    cin >> n >> m >> k;

    int s = 0;

    for (int i = 0; i < n; i++)

    {

        string si;

        cin >> si;

        int ni=0;

        for (int j=0;j<m;j++)

        {

            if (si[j]=='\*')

            {

                ni++;

            }

        }

        if (ni % k != 0)

        {

            s += ni / k + 1;

        }

        else

        {

            s += ni / k;

        }

        ni=0;

    }

    cout << s << endl;

    return 0;

}

**Тестування**

**VNS LAB 2**

**A black screen with white text

Description automatically generated**

**VNS LAB 3**

**A black background with white numbers and symbols

Description automatically generated**

**VNS LAB 7 TASK 1**

**A black background with white numbers and a black background

Description automatically generated**

**VNS LAB 7 TASK 2**

**A screenshot of a computer program

Description automatically generated**

**Робота в команді:**

Ми з командою збиралися разом 1 раз для обговорення 3 епіка.



**Посилання на файли у пулл реквесті**

**VNS LAB 2**

[epic 3 - Oleh Stanko by Stankoleh · Pull Request #182 · artificial-intelligence-department/ai\_programming\_playground\_2024](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/182/files#diff-496186e419db4e3a54f185412e5b3ca4e50f972d9a87575d0272b6607bb50a32)

**VNS LAB 3**

[epic 3 - Oleh Stanko by Stankoleh · Pull Request #182 · artificial-intelligence-department/ai\_programming\_playground\_2024](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/182/files#diff-7af9c9fa5fb1b946fc8d7b857e6b03873ee3da560150030ce44d5bfae6cd9b6e)

**VNS LAB 7 TASK 1**

[epic 3 - Oleh Stanko by Stankoleh · Pull Request #182 · artificial-intelligence-department/ai\_programming\_playground\_2024](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/182/files#diff-847999784f2e27a1d0d887aa06562c41356c1be052888be71a48d8c87d10557e)

**VNS LAB 7 TASK 2**

[epic 3 - Oleh Stanko by Stankoleh · Pull Request #182 · artificial-intelligence-department/ai\_programming\_playground\_2024](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/182/files#diff-775f44d012a6762ef44e3d84af15827f9297e870f2a307902f92df44b06f4a63)

Висновок: я навчився використовувати перезавантажені функції та покращив роботу з циклами й їх застосуваннями. Також я ознайомився з базовими математичними функціями