Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт № 3**

***з дисципліни:*** «Мови та парадигми програмування»

***з розділу***: «Цикли. Вкладені Цикли. Функції. Перевантаження функцій. Рекурсія»

***Виконав:***

студент групи ШІ-11

Стожук Роман Ігорович

# **Тема роботи:**

# Цикли. Вкладені Цикли. Функції. Перевантаження функцій. Рекурсія

# **Мета роботи:**

Навчитися працювати з циклами, функціями, перевантажувати функції, писати рекурсивні функції

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Цикли
* Тема №2: Перевантажування функцій
* Тема №3: Рекурсивні функції

1. Індивідуальний план опрацювання теорії:

* Тема №1: Цикли
  + Джерела Інформації
    - <https://acode.com.ua/urok-70-tsykl-while/>
    - <https://acode.com.ua/urok-71-tsykl-do-while/>
    - <https://acode.com.ua/urok-72-tsykl-for/>
    - <https://acode.com.ua/urok-73-operatory-break-i-continue/>
  + Що опрацьовано:
    - Роботу з циклами for, while, do while та операторами break і continue
  + Статус: Ознайомлений
  + Початок опрацювання теми: 18.11.2023
  + Звершення опрацювання теми: 18.11.2023
* Тема №2: Перевантажування функцій
  + Джерела Інформації:
    - <https://acode.com.ua/urok-102-parametry-i-argumenty-funktsij/>
    - <https://acode.com.ua/urok-15-funktsiyi-i-operator-return/>
    - <https://acode.com.ua/urok-108-perevantazhennya-funktsij/>
  + Що опрацьовано:
    - Перевантажування функцій
  + Статус: Ознайомлений
  + Початок опрацювання теми: 19.11.2023
  + Звершення опрацювання теми: 19.11.2023
* Тема №3: Рекурсивні функції
  + Джерела Інформації:
    - <https://acode.com.ua/urok-113-rekursiya-i-chysla-fibonachchi/>
  + Що опрацьовано:
    - Написання та роботу з рекурсивними функціями
  + Статус: Ознайомлений
  + Початок опрацювання теми: 20.11.2023
  + Звершення опрацювання теми: 20.11.2023

# **Виконання роботи:**

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

Завдання №3 VNS Lab 2

* Варіант завдання: 6
* Знайти суму ряду з точністю ε=0.0001, загальний член якого
* При визначенні суми членів ряду варто використовувати рекурентну формулу для отримання наступного члена ряду. При складанні програми вважати, що точність досягнута, якщо аn<ε.

Завдання №4 VNS Lab 3

* Варіант завдання: 6
* Для х, що змінюється від a до b з кроком (b-a)/10, обчислити функцію f(x), використовуючи її розклад в степеневий ряд для заданого n та для заданої точності ε=0.0001. Член суми доцільно представити у вигляді двох співмножників, один із яких обчислюється за рекурентним співвідношенням, а інший безпосередньо.
* Алгоритм розв’язання завдання зводиться до трьох циклів, причому два з них вкладені в третій. Внутрішні цикли підсумують доданки при фіксованому параметрі x, один для заданого n, інший для заданої точності ε. Зовнішній цикл організує зміну параметра х.

Завдання №5.1 VNS Lab 7 Task 1

* Варіант завдання: 6
* Написати функцію min зі змінною кількістю параметрів, що знаходить мінімальне із чисел типу int. Написати викликаючу функцію main, що звертається до функції min не менше трьох разів з кількістю параметрів 5, 10, 12.

Завдання №5.2 VNS Lab 7 Task 2

* Варіант завдання: 6
* Написати перевантажені функції для віднімання дійсних чисел і комплексних чисел й основну програму, що їх викликає.

Завдання №6 Class Practice Task

* Перерахувати всі книги.
* 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.
* 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.

Завдання №7 Self Practice Task

* <https://algotester.com/uk/ArchiveProblem/DisplayWithEditor/20020>
* За заданим повідомленням s, яке складається з великих латинських символів, виведіть YES, якщо Тойлет-мен погодиться допомогти людині, яка написала це повідомлення. У протилежному разі виведіть NO.
* Обмеження: 1≤|s|,k≤

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

Програма №3 VNS Lab 2

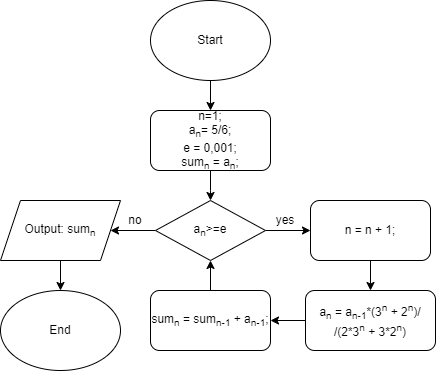


Рисунок 1Блок схема до програми №3

* Планований час на реалізацію: 20 хв

Програма №4 VNS Lab 3

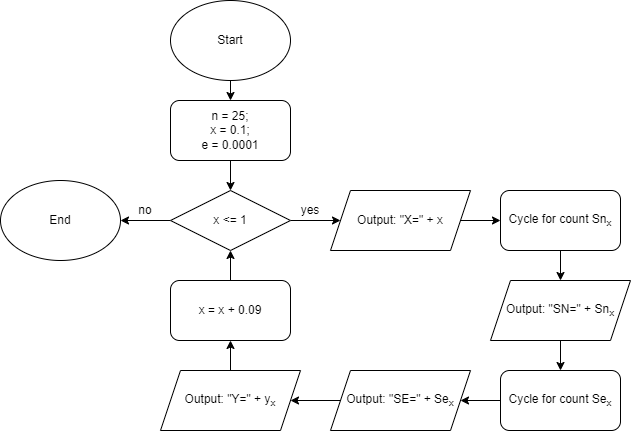


Рисунок 2Блок схема до програми №4

* Планований час на реалізацію: 60 хв

Програма №5.1 VNS Lab 7 task 1

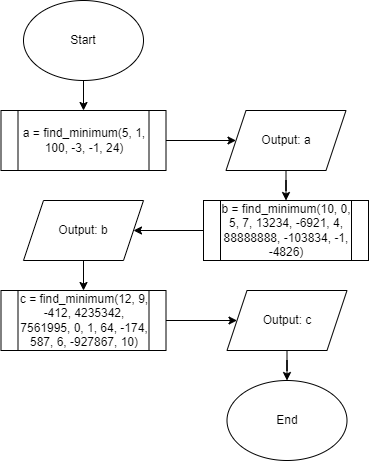


Рисунок 3Блок схема до програми №5.1

* Планований час на реалізацію: 25хв

Програма №5.2 VNS Lab 7 task 2

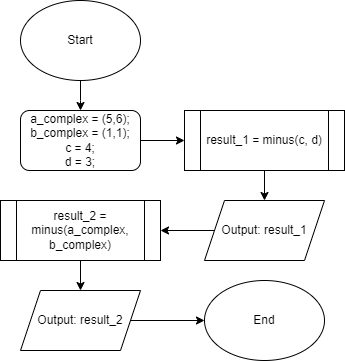


Рисунок 4Блок схема до програми №5.2

* Планований час на реалізацію: 20хв

Програма №6 Class Practice Task

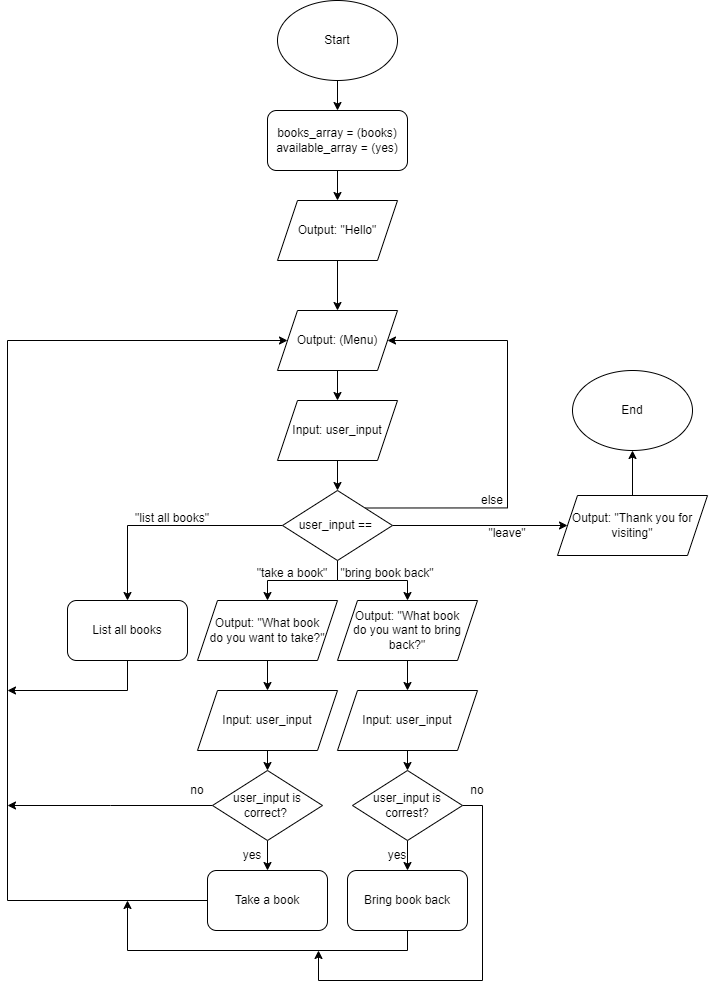


Рисунок 5Блок схема до програми №6

* Планований час на реалізацію: 1 пара

Програма №7 Self Practice Task

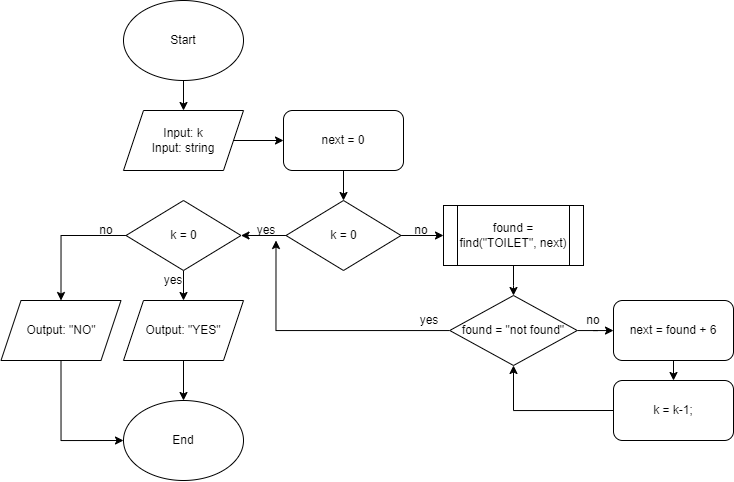


Рисунок 6Блок схема до програми №7

* Планований час на реалізацію: 20хв

## **3. Конфігурація середовища до виконання завдань:**

## Змін у конфігурацію середовища не вносив.

## **4. Код програм з посиланням на зовнішні ресурси:**

Завдання №3 VNS Lab 2

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/447/files#diff-6efe925040e9e4397893712ad4d7ebef20fcec5717970417b62890c866173a27>

#include <iostream>

#include <cmath>

int main() {

    double a = (5.0/6), sum = a, e = 0.0001, n = 1;

    while (a>=e) {

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

        sum += a;

        ++n;

    }

    printf("%.4f", sum);

    return 0;

}

Завдання №4 VNS Lab 3

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/447/files#diff-e1aefd7f5c511fd8c488e6313a029b3a1ab8a1d83441c572445159c65bb70c91>

#include <iostream>

#include <cmath>

#include <iomanip>

const long double PI\_4 = 0.785398163397448309616;

const long double E = 2.71828182845904523536;

int main() {

    short n = 25;

    double x = 0.1, e = 0.0001;

    while(x <= 1) {

        std::cout << "X=" << std::setw(10) << std::left << x;

        double sn = 1 + cos(PI\_4) \* x, pan = x;

        //an = pan \* cos\*;

        for(short i = 2; i <= n; i++) {

            pan \*= x/n;

            sn += cos(n\*PI\_4) \* pan;

        }

        std::cout << "SN=" << std::setprecision(9) << std::setw(16) << std::left << sn;

        double se = 1 + cos(PI\_4) \* x, pae = x, i = 2, ae;

        do {

            pae \*= x/n;

            ae = pae \* cos(n\*PI\_4);

            se += ae;

        } while (ae > e);

        std::cout << "SE=" << std::setw(16) << std::left << se;

        std::cout << "Y=" << (pow(E, x\*cos(PI\_4) ) \* cos(x\*sin(PI\_4))) << std::endl;

        x += (1-0.1)/10;

    }

    return 0;

}

Завдання №5.1 VNS Lab 7 Task 1

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/447/files#diff-c446a091aecae054ea779b838e0e0138d3ea5f7819adea6ab9596d67e8dad37f>

#include <iostream>

#include <cstdarg>

int min(int n, ...) {

    va\_list VL;

    va\_start(VL, n);

    int min = INT\_MAX;

    for (int i = 0; i < n; i++) {

        int cur = va\_arg(VL, int);

        if (cur < min) min = cur;

    }

    va\_end(VL);

    return min;

}

int main( ) {

    std::cout << min(5, 1, 100, -3, -1, 24) << '\n';

    std::cout << min(10, 0, 5, 7, 13234, -6921, 4, 88888888, -103834, -1, -4826) << '\n';

    std::cout << min(12, 9, -412, 4235342, 7561995, 0, 1, 64, -174, 587, 6, -927867, 10) << '\n';

    return 0;

}

Завдання №5.2 VNS Lab 7 Task 2

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/447/files#diff-21cadf211b293c3d3f71e64adac39751801ab972773a5e4d1bc3013d4229b88e>

#include <iostream>

class Complex {

private:

double m\_real;

double m\_imagine;

public:

Complex(double r=0, double im=0): m\_real(r), m\_imagine(im) {}

double getReal() {return m\_real;}

double getImagine() {return m\_imagine;}

};

Complex minus(Complex first, Complex second) {

    return Complex(first.getReal() - second.getReal(), first.getImagine() - second.getImagine());

}

std::ostream& operator<<(std::ostream &out, Complex number) {

    out << "(" << number.getReal() << "," << number.getImagine() << ")";

    return out;

}

double minus(double first, double second) {return first - second;}

int main() {

    Complex a(5,6), b(1,1);

    std::cout << minus(4, 3) << '\n' << minus(a,b);

    return 0;

}

Завдання №6 Class Practice Task

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/447/files#diff-01616db55fe758da965f6146f1d90e0748cc9929cb13dccc4872ce19df7d936a>

#include <iostream>

#include <vector>

#include <string>

using namespace std;

int main() {

    const short quantity = 4;

    std::vector<std::string> books = {"Harry Potter", "The Hobbit", "Pride and Prejudice", "1984"};

    std::vector<bool> available = {1, 1, 1, 1};

    cout << "Hello!" <<endl;

    string input;

    do {

        Again:

        cout << "How can I help You?\n(list all books/take a book/bring book back/leave)\n-";

        getline(cin, input);

        if (input == "list all books") {

            for (int i = 0; i < quantity; i++) {

            cout << books[i] << " is " << (available[i] ? "available." : "not available.") << '\n';

            }

            cout << endl;

        }

        else if (input == "take a book") {

            cout << "What book do you want to take?\n-";

            getline(cin, input);

            bool correctInput = false;

            for (const auto &book : books) {

                if (input == book) {

                    correctInput = true;

                    if (available[&book - &books[0]]) {

                        cout << "Here you are!" << endl;

                        available[&book - &books[0]] = 0;

                    }

                    else { cout << "This book is not available! Try again later" << endl; }

                    break;

                }

            }

            cout << endl;

            if (!correctInput) goto Again;

        }

        else if (input == "bring book back") {

            cout << "What book do you want to bring back?\n-";

            getline(cin, input);

            bool correctInput = false;

            for (const auto &book: books) {

                if (input == book) {

                    correctInput = true;

                    if (!(available[&book - &books[0]]) ) {

                        cout << "Thank You!" << endl;

                        available[&book - &books[0]] = 1;

                    }

                    else { cout << "You did not take this book from our library!" << endl; }

                    break;

                }

            }

            cout << endl;

            if (!correctInput) goto Again;

        }

        else if (input != "leave") {

            cout << endl;

            goto Again;

        }

    } while (input != "leave");

    cout << "Thank you for visiting!";

    return 0;

}

Програма №7 Self Practice Task

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/447/files#diff-03921e801d280c41a242e48670eb411c74f8b9eb4374d7027084596f0762abf5>

#include <iostream>

int main() {

    int k;

    std::string s, need = "TOILET";

    std::cin >> k >> s;

    size\_t next = 0;

    while (k != 0) {

        auto found = s.find(need, next);

        if (found != std::string::npos) {

            next = found + 6;

            k--;

        }

        else {break;}

    }

    std::cout << (k == 0 ? "YES" : "NO");

    return 0;

}

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

Завдання №3 VNS Lab 2

Вихідні дані: 1.4999

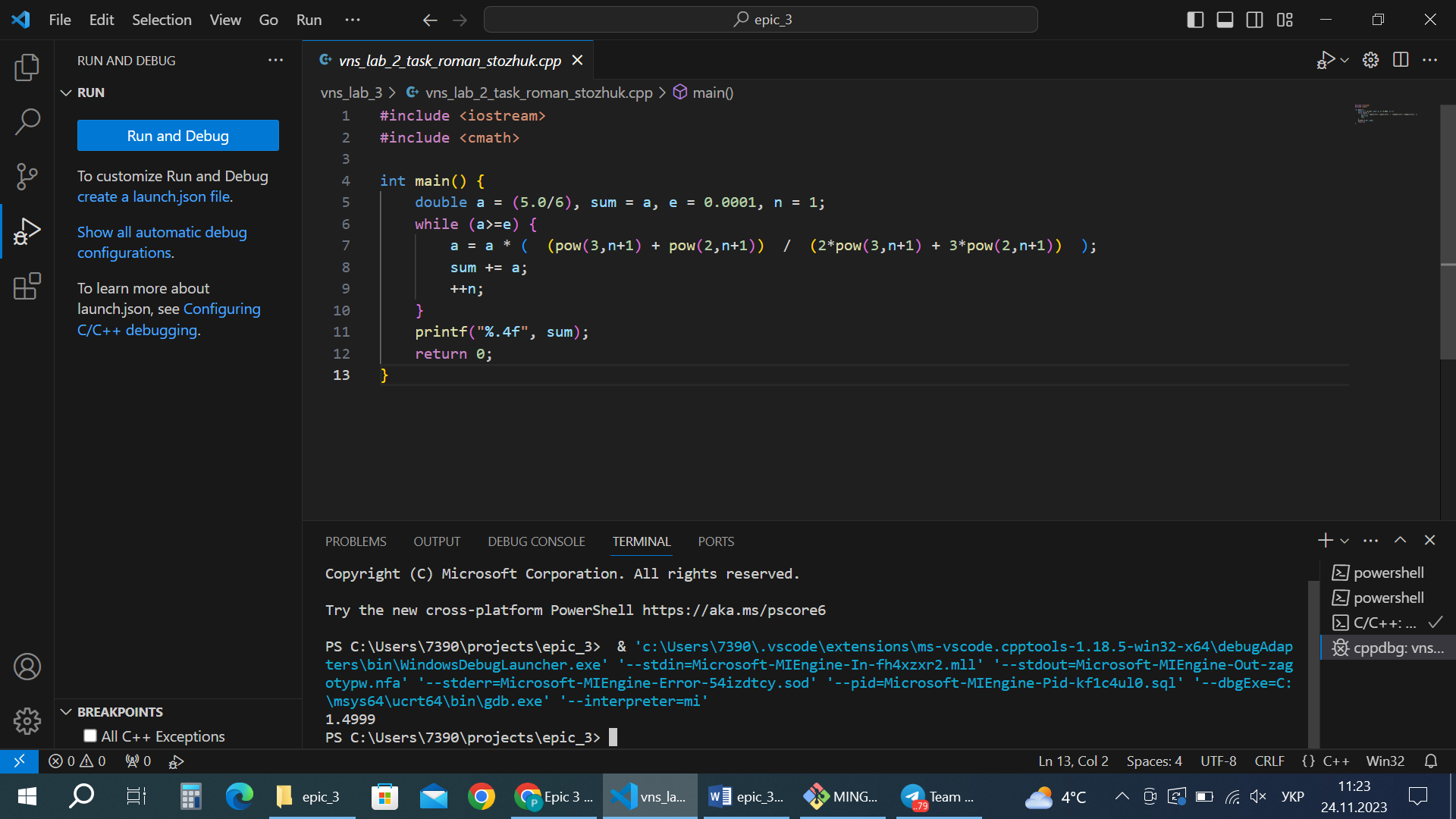


Рисунок 7Результат виконання програми №3

Час затрачений на виконання завдання: 30хв

Завдання №4 VNS Lab 3

Вихідні дані:

X=0.1 SN=1.07099466 SE=1.07099465 Y=1.0705886

X=0.19 SN=1.13537917 SE=1.13537911 Y=1.13348619

X=0.28 SN=1.2002325 SE=1.20023222 Y=1.1951366

X=0.37 SN=1.26555979 SE=1.26555893 Y=1.25483838

X=0.46 SN=1.33136626 SE=1.33136622 Y=1.31181165

X=0.55 SN=1.39765719 SE=1.39765709 Y=1.36519428

X=0.64 SN=1.46443795 SE=1.46443775 Y=1.41403806

X=0.73 SN=1.531714 SE=1.53171361 Y=1.4573053

X=0.82 SN=1.59949086 SE=1.59949017 Y=1.49386558

X=0.91 SN=1.66777415 SE=1.66777298 Y=1.52249292

X=1 SN=1.73656956 SE=1.73656768 Y=1.54186346

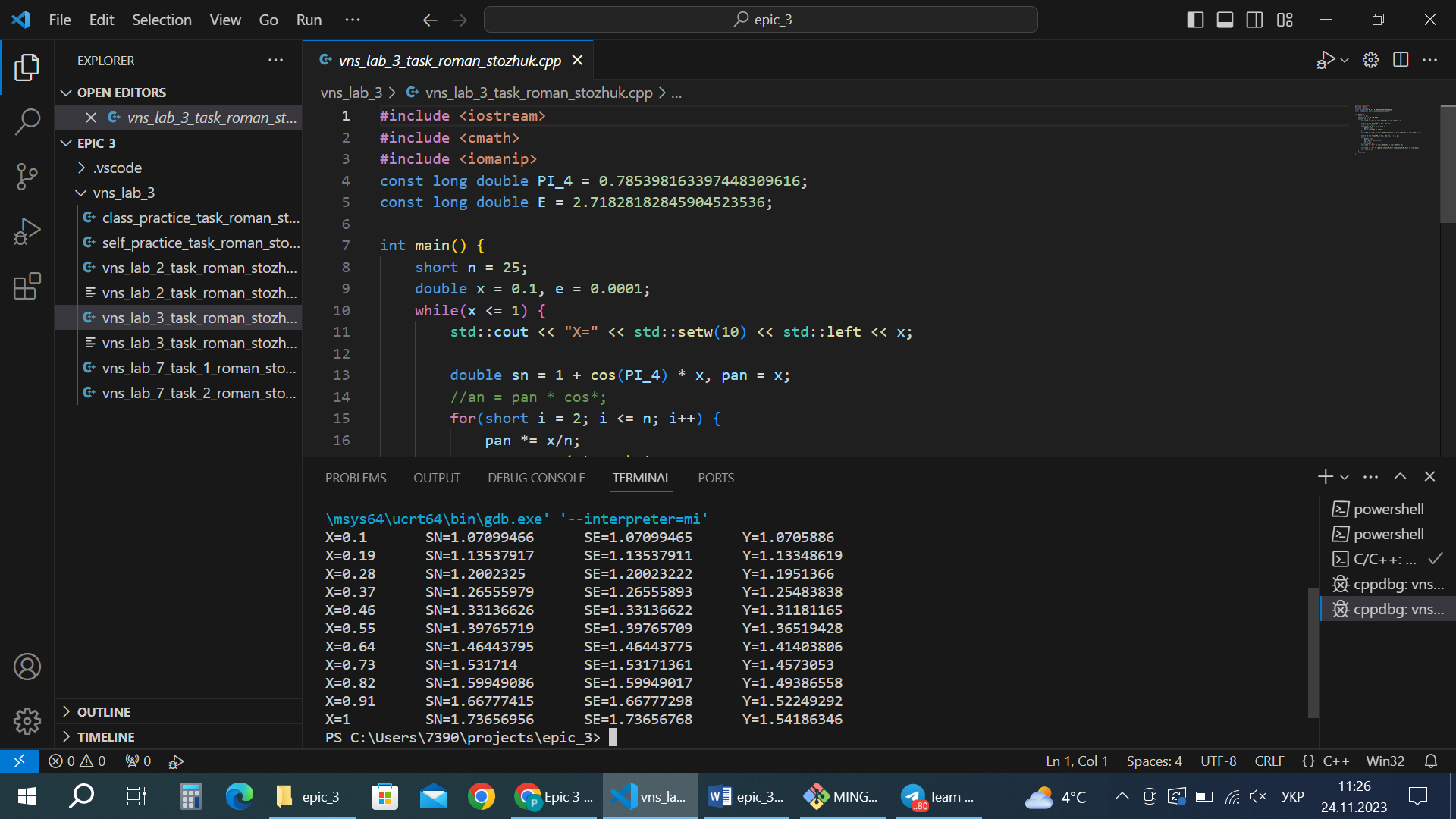


Рисунок 8Результат виконання програми №4

Час затрачений на виконання завдання: 40хв

Завдання №5.1 VNS Lab 7 Task 1

Вихідні дані: -3

-103834

-927867

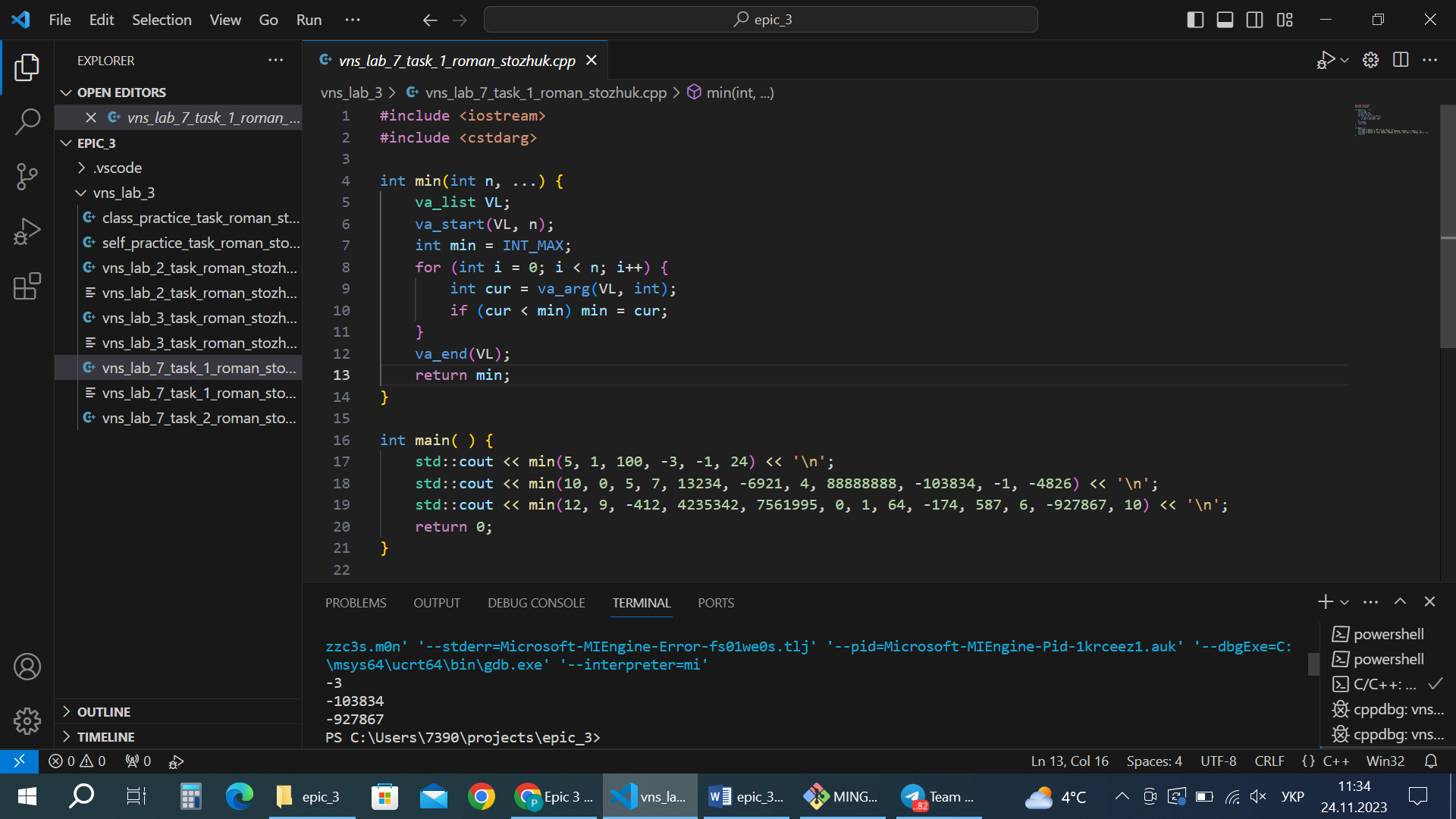


Рисунок 9Результат виконання програми №5.1

Час затрачений на виконання завдання: 30хв

Завдання №5.2 VNS Lab 7 Task 2

Вихідні дані: 1

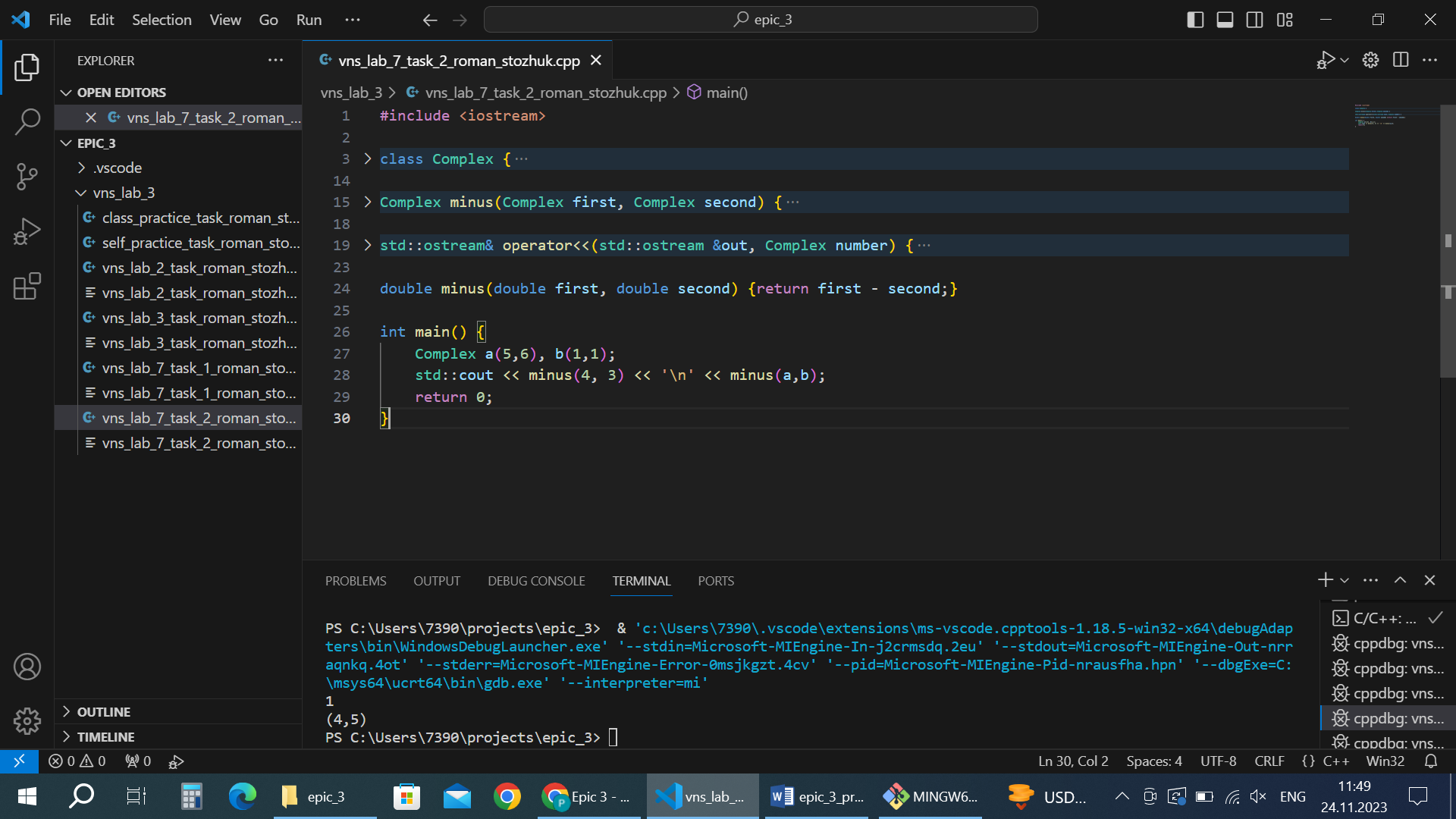
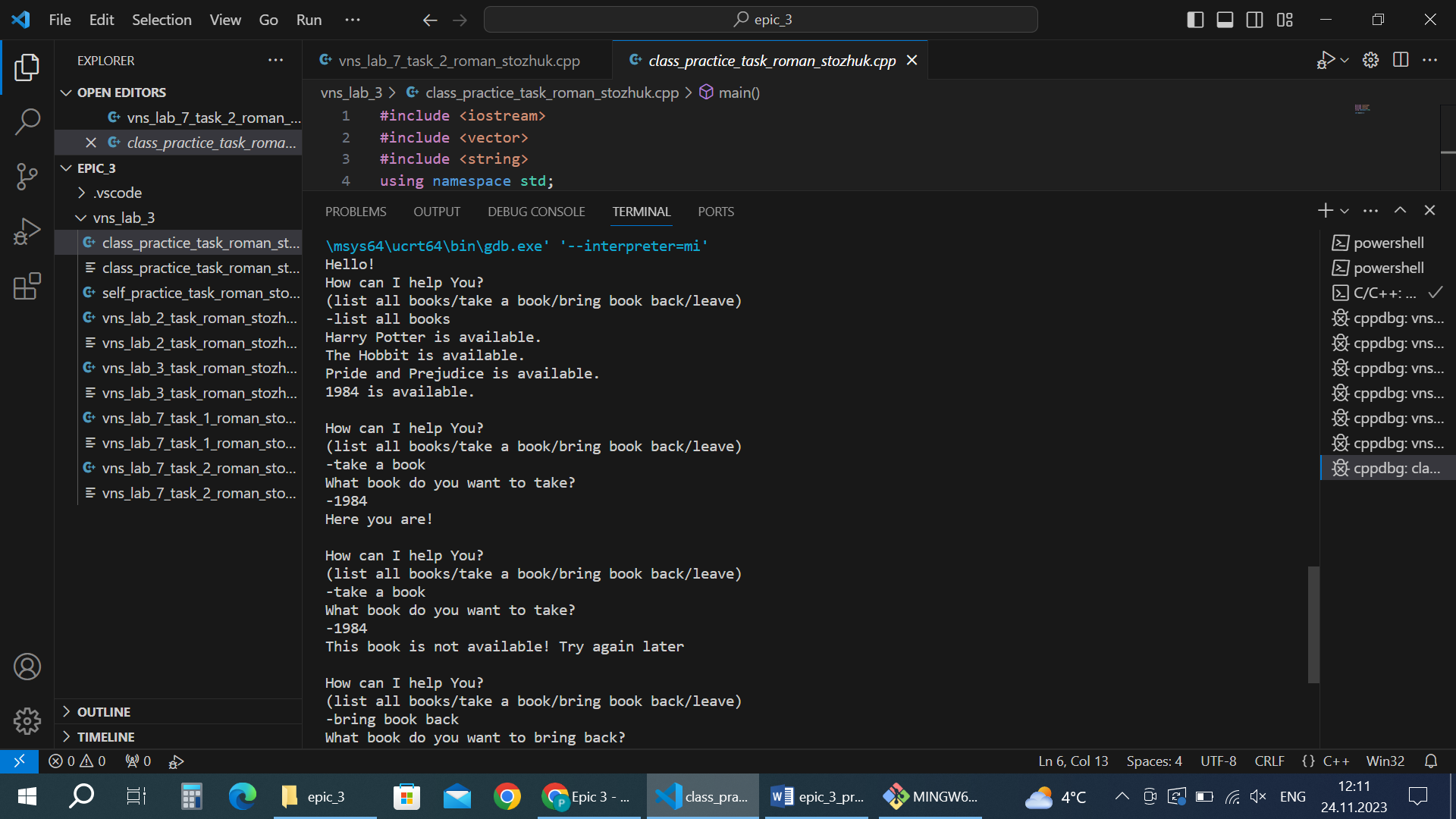
(4,5)

Рисунок 10Результат виконання програми №5.1

Час затрачений на виконання завдання: 3 години

Завдання №6 Class Practice Task



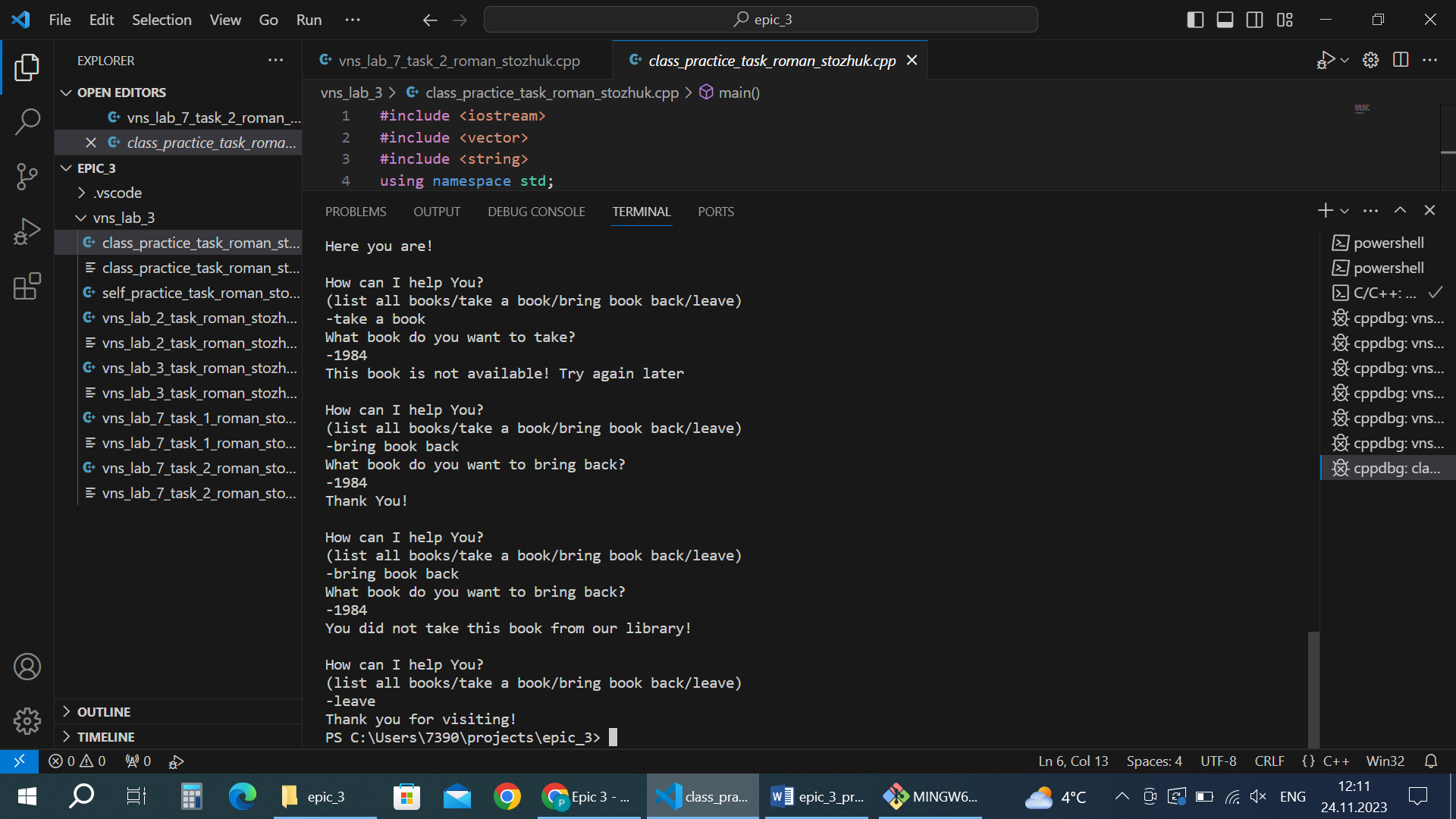


Рисунок 11Результат виконання програми №6

Час затрачений на виконання завдання: 1 пара

Завдання №7 Self Practice Task

Вхідні дані: 2

HELPTOILETMENPLEASETOILET

Вихідні дані: YES

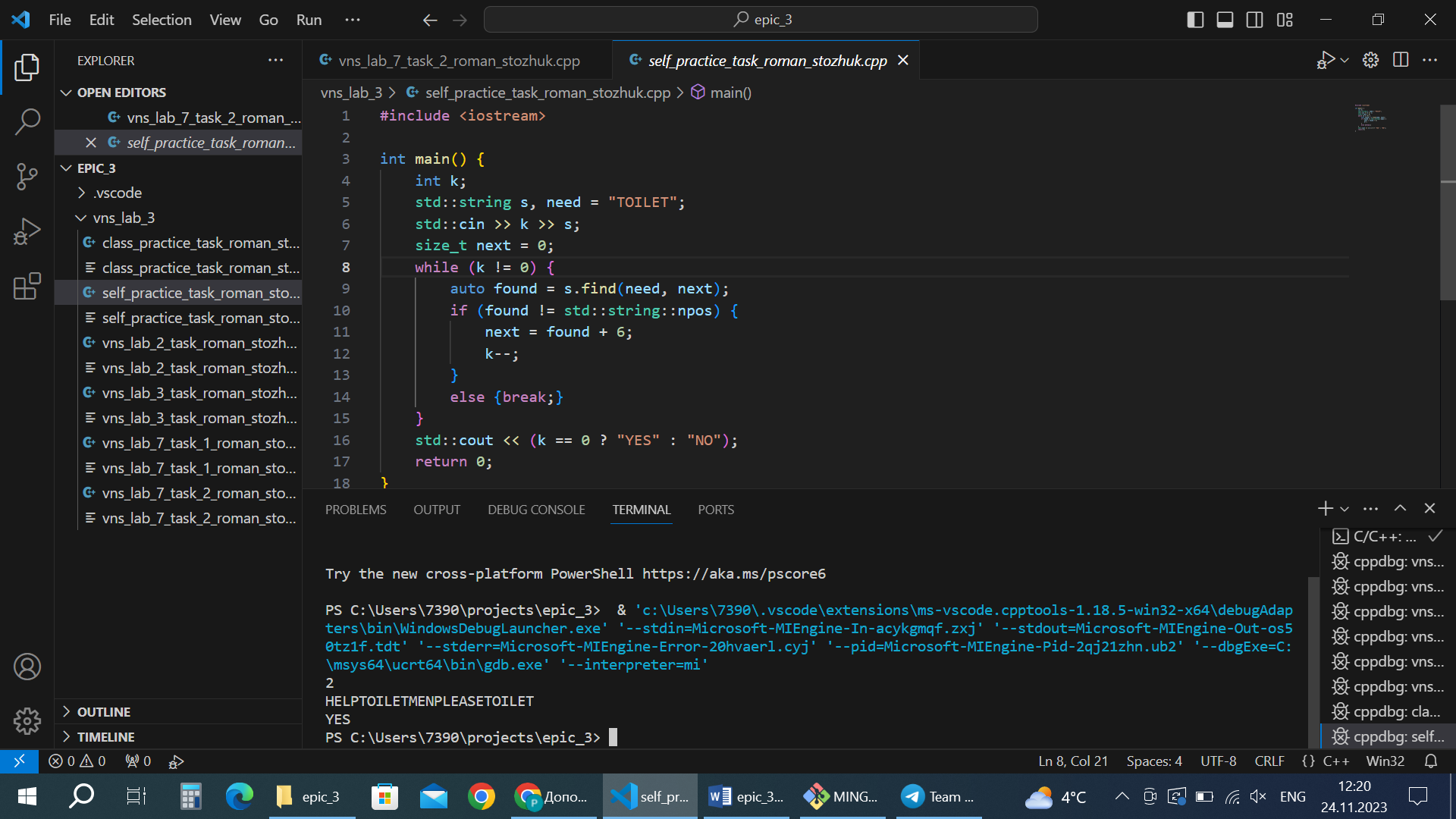


Рисунок 12Результат виконання програми №7

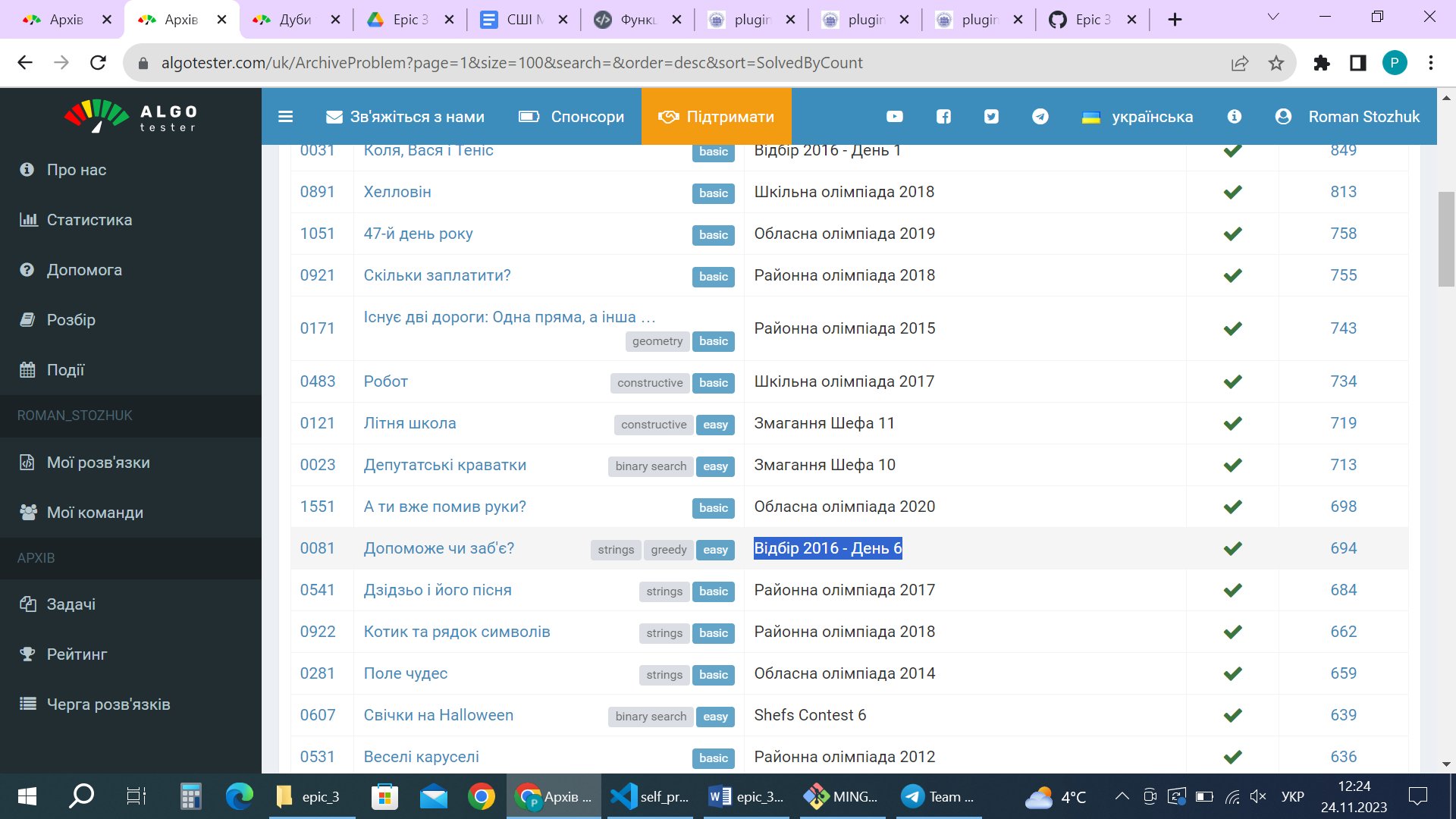


Рисунок 13Підтвердження коректної роботи програми №7

Час затрачений на виконання завдання: 90 хв

# **Висновки:**

Протягом третього епіку я навчився працювати з циклами, писати рекурсивні функції та перевантажувати функції