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:*** «Epic 3. Цикли. Вкладені Цикли. Функції. Перевантаження функцій. Рекурсія. Змінні»

***Виконав:***

студент групи ШІ-12

Воронін Олександр Олександровчив

**Тема роботи:**

Робота з фунціями , створення циклів, перезавантаження циклу

**Мета роботи:**

Навчитись використовувати цикли, застосовувати перевантаження функції і вміти створювати функції

**Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Theory Education Activities
* Тема №2: Work With Team
* Завдання №3: Lab# programming: VNS Lab 2
* Завдання №4: Lab# programming: VNS Lab 3
* Завдання №5  Lab# programming: VNS Lab 7 task 1
* Завдання №6  Lab# programming: VNS Lab 7 task 2
* Завдання №7 Practice# programming: Class Practice Task
* Завдання №8 Practice# programming: Self Practice Task

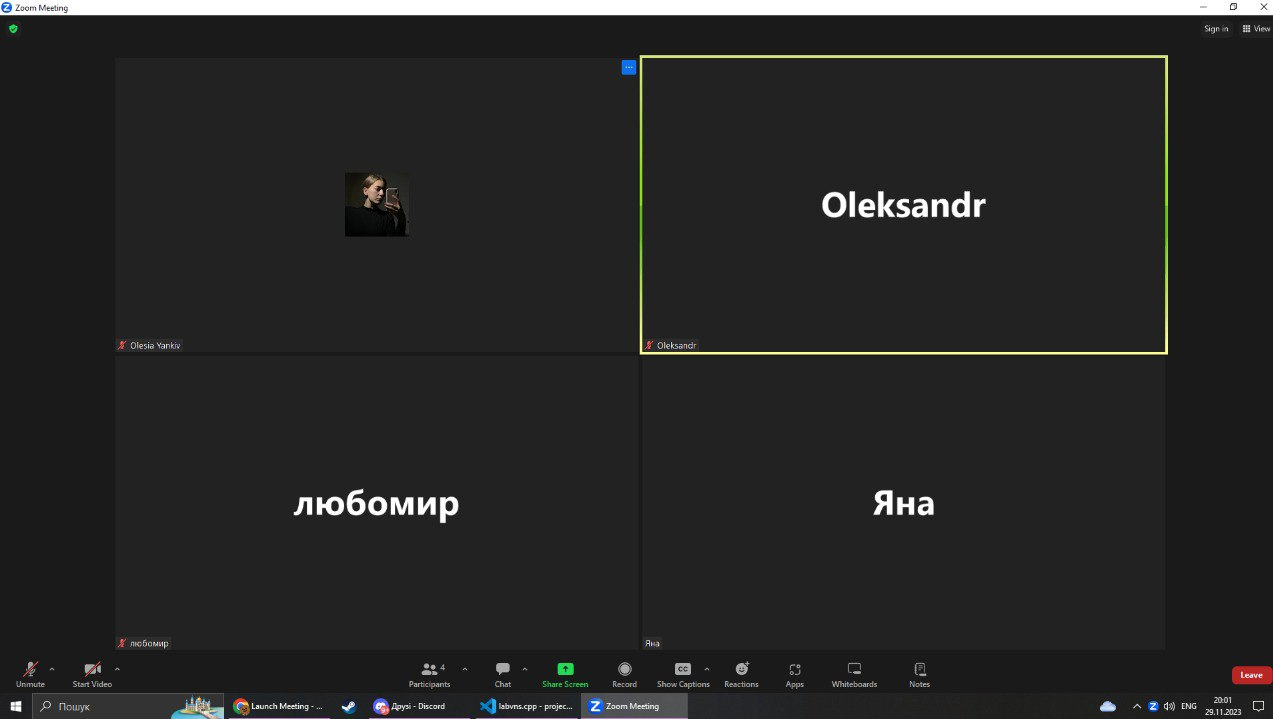
1. Індивідуальний план опрацювання теорії:

* **Тема №1: Theory Education Activities**

1. Джерела Інформації

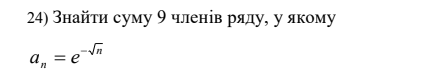
* <https://www.youtube.com/watch?v=oDC208zvsdg&t=405s&ab_channel=CodeVault>
* <https://www.w3schools.com/cpp/cpp_functions.asp>
* <https://www.w3schools.com/cpp/cpp_function_param.asp>
* <https://www.w3schools.com/cpp/cpp_for_loop.asp>
* <https://www.youtube.com/watch?v=3KJfisev6SI&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%A3%D1%80%D0%BE%D0%BA%D0%B8%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F>
* <https://stackoverflow.com/questions/6892145/c-unlimited-parameters-to-array>

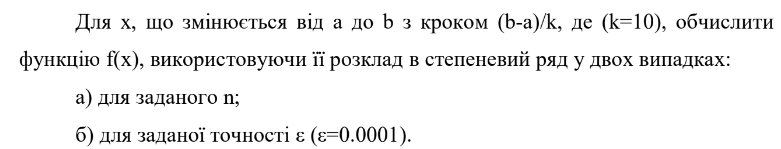
1. Що опрацьовано: цикли, функції, перевантаження функції
2. Статус: Ознайомлений

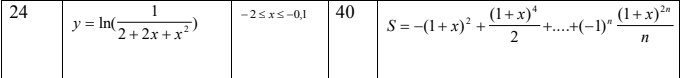
* **Тема №2: Work With Team**
  + - **Провели онлайн зустріч**
    - 

**Виконання роботи:**

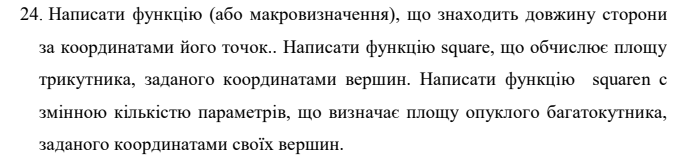
* 1. **Опрацювання завдання та вимог до програм та середовища:**
* Task 3: Lab# programming: VNS Lab 2



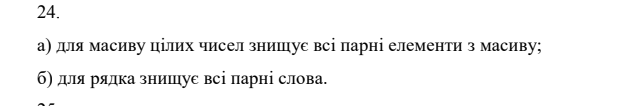
* Task 4: Lab# programming: VNS Lab 3



* Task 5: Lab# programming: VNS Lab 7 task 1



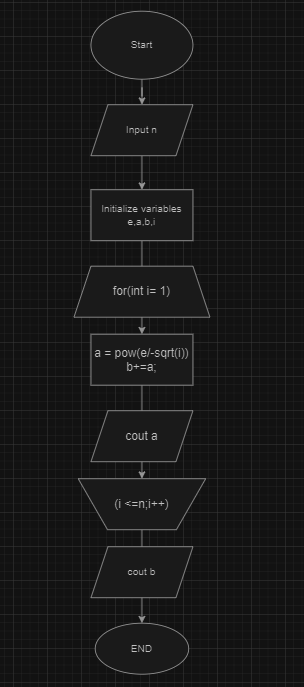
* Task 6: Lab# programming: VNS Lab 7 task 2



* Task 7: Practice# programming: Class Practice Task
  + Створити програму “Бібліотека”, в якій можна позичати та поветати книжки
* Task 8: Practice# programming: Self Practice Task
  + Марічка і печиво
    - <https://algotester.com/uk/ArchiveProblem/DisplayWithEditor/2#mySolutions>

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

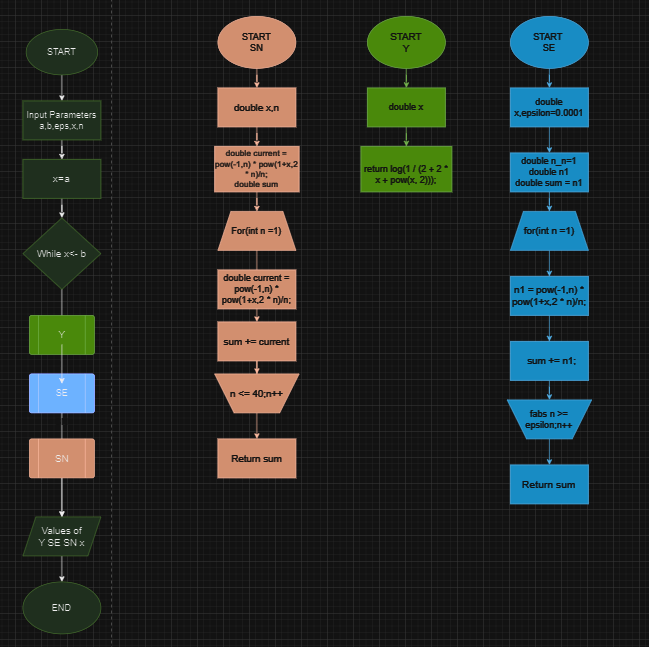
* Task 3  Lab# programming: VNS Lab 2



*Рисунок 1: Блок схема до програми №3*

Запланований час: 20 хвилин

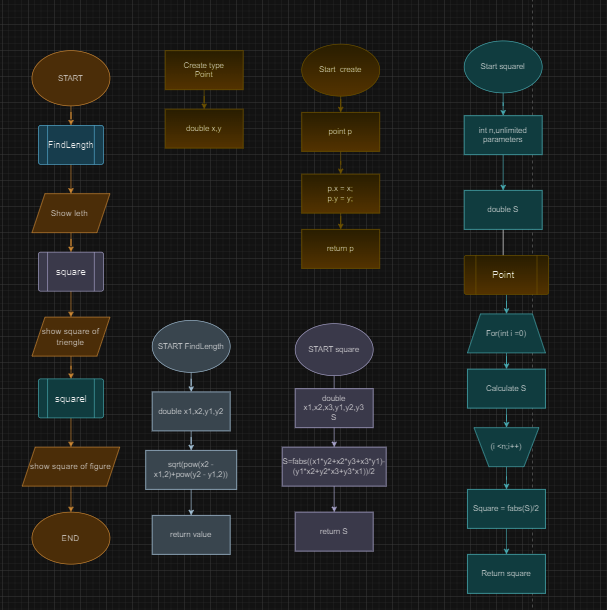
* Task 4: Lab# programming: VNS Lab 3



*Рисунок 3: Блок схема до програми №4*

Запланований час: 2 години

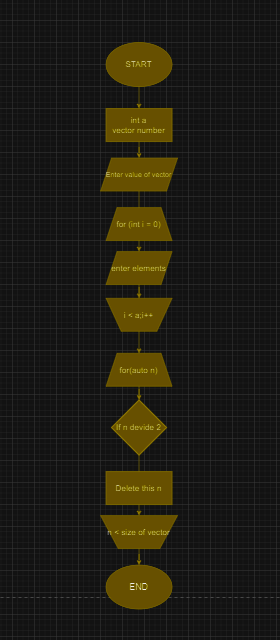
* Task 5: Lab# programming: VNS Lab 7 task 1



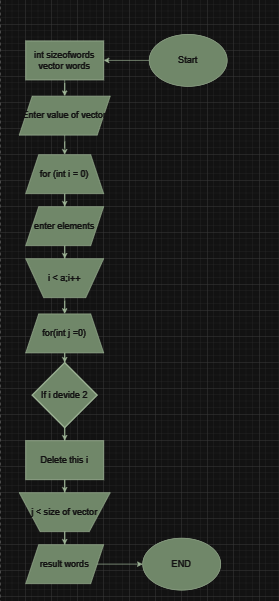
*Рисунок 2: Блок схема до програми №5*

Запланований час: 2 години

* Task 6: Lab# programming: VNS Lab 7 task 2



*Рисунок 4: Блок схема до програми №5*



*Рисунок 4: Блок схема до програми №5*

Запланований час: година

* Task 7: Practice# programming: Class Practice Task

Запланований час: 2 години

* Task 8: Practice# programming: Self Practice Task

Запланований час: пів години

* 1. **Код програм з посиланням на зовнішні ресурси:**
* Task 3  Lab# programming: VNS Lab 2
* #include <iostream>
* #include <cmath>
* using namespace std;
* int main(){
* const double e = 2.7;
* double a,b;
* int n;
* cout << "Enter your n: " << endl;
* cin >> n;
* for (int i = 1; i <= n; i++)
* // цикл від 1 до 9
* {
* a= pow(e,-sqrt(i));
* b +=  a;
* cout << a <<endl;
* }
* cout << b << " result" << endl;
* }

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/593/commits/018027b3861dd61b74970e1080b7fd6943c73e1f#diff-099966fea410d59725b6555356e9116e89d5ca98c7b79071c014f63df946acf9>

* Task 4: Lab# programming: VNS Lab 3

#include <iostream>

#include <cmath>

using namespace std;

double Y(double x)

{

    return log(1 / (2 + 2 \* x + pow(x, 2)));

}

double SN(double x, double n) {

    double current = pow(-1,n) \* pow(1+x,2 \* n)/n;

    double sum =0;

    for (int n = 1; n <= 40; n++)

    {

        current = pow(-1,n) \* pow(1+x,2 \* n)/n;

        sum += current;

    }

    return sum;

}

double SE(double x, double epsilon = 0.0001)

    {

    double n\_n=1;

    double n1 = pow(-1,n\_n) \* pow(1+x,2 \* n\_n)/n\_n;

    double sum = n1;

    for (int n = 1; fabs(n1) >= epsilon; ++n) {

        n1 = pow(-1,n) \* pow(1+x,2 \* n)/n;

        sum += n1;

    }

    return sum;

    }

int main() {

    int n;

    double a = -2.0;

    double b = -0.1;

    double eps = 0.0001;

    double x = a;

    while (x <= b) {

        cout << "  X = " << x << "   SN = " << SN(x,n) << "  SE = " << SE(x,eps) << "  Y = " << Y(x) << endl;

        x += (b - a) / 10;

    }

}

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/593/commits/018027b3861dd61b74970e1080b7fd6943c73e1f#diff-c2c55656aaf6909f7a89e5c14825ded3fbfb0900e0a7caae6299a18899514b4f>

* 5: Lab# programming: VNS Lab 7 task 1
* #include <iostream>
* #include <cmath>
* #include <cstdarg>
* using namespace std;
* struct Point
* {
* double x, y;
* };
* double FindLength(double x1,double x2,double y1, double y2)
* {
* return sqrt(pow(x2 - x1,2)+pow(y2 - y1,2));
* }
* double square(double x1, double x2,double x3,double y1,double y2,double y3)
* {
* //Shoelace formula
* double S = fabs((x1\*y2+x2\*y3+x3\*y1)-(y1\*x2+y2\*x3+y3\*x1))/2;
* return S;
* }
* //Створення змінної типу Point з кординатами x y
* Point create(double x,double y){
* Point p;
* p.x = x;
* p.y = y;
* return p;
* }
* double squarel(int n, ... )
* {
* va\_list args;
* va\_start(args, n);
* Point  start = va\_arg(args, Point),
* current = start,
* next;
* double S = 0.0;
* for (int i = 0; i < n - 1; ++i) {
* next = va\_arg(args, Point);
* S += current.x \* next.y - next.x \* current.y;
* current = next;
* }
* S += current.x \* start.y - start.x \* current.y;
* va\_end(args);
* double Square = fabs(S) / 2;
* return Square;
* }
* int main()
* {
* cout << "the length of the points " << FindLength(5,4,7,3)<<endl;;
* cout << "Square of triangle " <<square(3,2,3,1,3,3)<<endl;
* cout << "Sixangles "<< squarel(6,
* create(3, 76),
* create(63, 9),
* create(23, -67),
* create(1, 0),
* create(56, 2),
* create(49 ,33)
* ) <<endl;
* cout << "10 angles "<< squarel(10,
* create(3, 76),
* create(63, 9),
* create(23, -67),
* create(1, 0),
* create(56, 2),
* create(49 ,33),
* create(49 ,5),
* create(5 ,-2),
* create(557 ,8),
* create(3 ,3)
* ) <<endl;
* }

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/593/commits/018027b3861dd61b74970e1080b7fd6943c73e1f#diff-23de51a15bd3f5f96d20fcaf3ae93fd7c8e28865b5aa606c002a18a873c5b1ef>

* Task 6: Lab# programming: VNS Lab 7 task 2
* #include <iostream>
* #include <vector>
* #include <string>
* using namespace std;
* int main(){
* int a;
* vector<int> number;
* cout<<"The size of vector"<<endl;
* cin >>a;
* for(int i = 0;i<a ;i++){
* int elements;
* cout <<"Type numbers"<<endl;
* cin >> elements;
* number.push\_back(elements);
* }
* for(auto n = number.begin();n != number.end();){
* if(\*n % 2 !=0){
* n = number.erase(n);
* }
* else{n++;}
* }
* cout << "Elements of vector: ";
* for (int num : number) {
* cout << num << " ";
* }
* cout <<endl;
* vector<string> words;
* int sizeOfrWords;
* cout <<"Size of sentence "<<endl;
* cin >> sizeOfrWords;
* for(int j=0;j<sizeOfrWords;j++){
* string w;
* cout <<"Enter words "<<endl;
* cin >> w;
* words.push\_back(w);
* }
* string resultString;
* for (int i = 0; i < words.size(); ++i)
* {
* if (i % 2 == 0) {
* resultString += words[i] + " ";
* }
* }
* cout << resultString <<endl;
* }

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/593/commits/018027b3861dd61b74970e1080b7fd6943c73e1f#diff-5a0365f57071d7cef17402d9eaba4d952d2b6625ec523a7a4b827960cf3fb081>

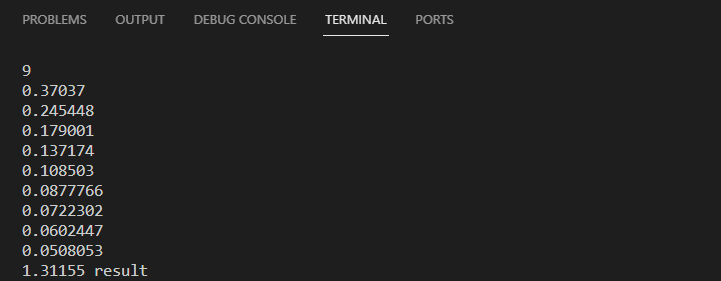
* Task 7: Practice# programming: Class Practice Task
* #include <iostream>
* #include <vector>
* #include <string>
* #include <cstdio>
* using namespace std;
* int main(){
* vector<string>books ={"Atlant", "1984", "Pepa Pig", "Game of thrones"};
* vector<bool>available ={true, true, true, true};
* int choice;
* string bookName;
* char continueChoice;
* bool exit =false;
* while(!exit){
* menu:
* cout << "\_\_\_\_LIBRARY\_\_\_\_ \n";
* cout << "1. List all books\n";
* cout << "2. Borrow a book\n";
* cout << "3. Return a book\n";
* cout << "4. Exit\n";
* cin >> choice;
* switch (choice)
* {
* case 1:
* for(int i=0;i<books.size();i++)
* {
* cout << i + 1  << ". " << books[i] << (available[i] ? " (Available)" : " (Borrowed)") << "\n";
* }
* break;
* case 2:
* cout << "Which book you want to take \n";
* cin.ignore();
* getline(cin, bookName);
* for(const auto &book : books)
* {
* if(book==bookName)
* {
* if(available[&book - &books[0]])
* {
* available[&book - &books[0]]= false;
* cout << "Book is borrowed: " << bookName << endl;
* }
* else
* {
* cout << "Sorry,this book is already borrowed \n";
* }
* break;
* }
* }
* break;
* case 3:
* cout << "Which book you want to return \n";
* cin.ignore();
* getline(cin, bookName);
* for(const auto &book : books)
* {
* if(book==bookName)
* {
* if(!(available[&book - &books[0]]))
* {
* available[&book - &books[0]]= true;
* cout << "Book is returned " << bookName << endl;
* break;
* }
* else
* {
* cout << "Book wasn't borrowed.\n";
* break;
* }
* }
* }
* break;
* case 4:
* cout << "Bye" <<endl;
* return 0;
* break;
* default:
* cout << "Uncorrect choice" << endl;
* goto menu;
* }
* do{
* cout << "Do you want to perform another operation? (Y/N): ";
* cin >> continueChoice;
* }while (continueChoice != 'Y' && continueChoice != 'N' && continueChoice != 'y' && continueChoice != 'n');
* if (continueChoice == 'Y' || continueChoice == 'y') {
* goto menu;
* } else {
* cout << "Goodbye!";
* exit = true;
* }
* }
* return 0;
* }

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/593/commits/018027b3861dd61b74970e1080b7fd6943c73e1f#diff-ce25db7b2fac1c8ab4fb260318b3cf70f6d64722b8989242b3d419ac9e53e4da>

* Task 8: Practice# programming: Self Practice Task
* #include <iostream>
* using namespace std;
* int main(){
* long long int n;
* cin >> n;
* long long int array[n];
* long long int result=0;
* for(int i=0;i < n;i++){
* cin >> array[i];
* array[i] -= 1;
* result += array[i];
* }
* cout <<result;
* return 0;
* }

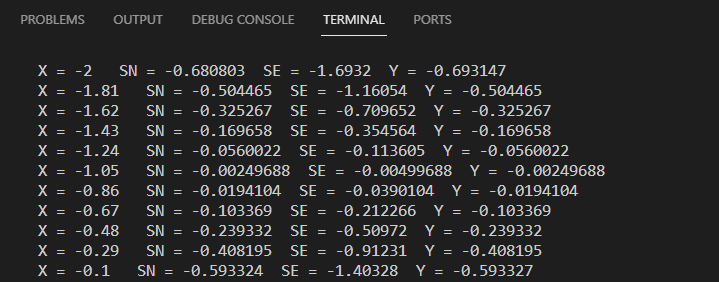
<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/593/commits/018027b3861dd61b74970e1080b7fd6943c73e1f#diff-e264fe67e9b8d8d5a43e37555c21275a98692919d34b3d6cdff99af4e1cd4147>

* 1. **Результати виконання завдань, тестування та фактично затрачений час:**
* Task 3  Lab# programming: VNS Lab 2



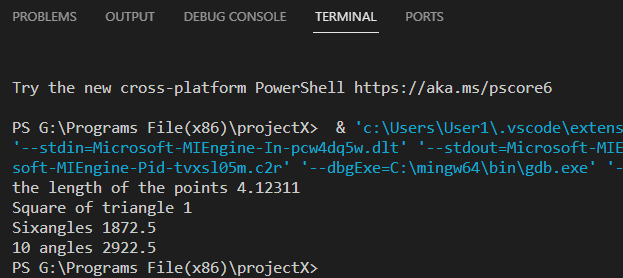
Затрачено пів години

* Task 4: Lab# programming: VNS Lab 3



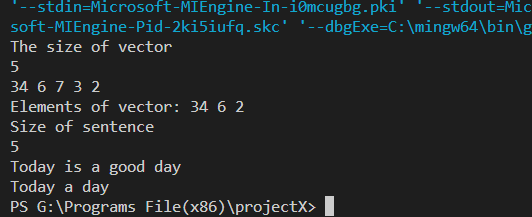
Затрачено 5 годин

* Task 5: Lab# programming: VNS Lab 7 task 1

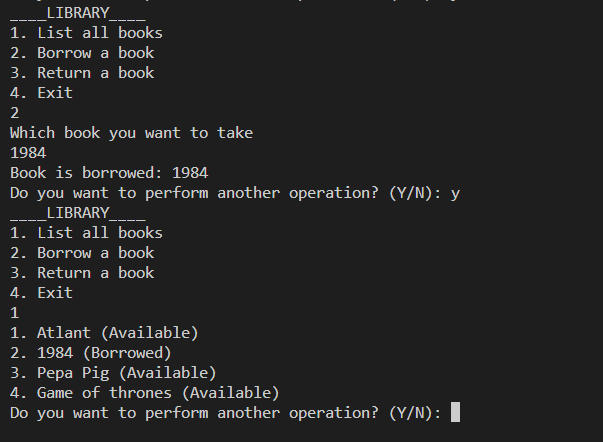


Затрачено 3 години

* Task 6: Lab# programming: VNS Lab 7 task 2

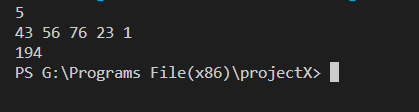


Затречений час 1 година

* Task 7: Practice# programming: Class Practice Task
* 

Затрачено півтори години

* Task 8: Practice# programming: Self Practice Task



Затрачено 20 хв

**Висновки:** Навчився використовувати функції зі змінною кількістю параметрів, перевантаження функції, глибше ознайомився з циклами.