Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт № 3(Algolab 2. Algolab 3.VNS Lab 4.VNS Lab 5. Class Practice Work. Self Practice Work)**

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

***з розділу***: «Епік 4: « Прості структури даних. Одновимірні масиви. Двовимірні Масиви. Алгоритми обробки. »

***Виконав:***

***Бачик Анатолій Андрійович*** групи ШІ-13

### Тема роботи:

Прості структури даних. Одновимірні масиви. Двовимірні Масиви. Алгоритми обробки

### Мета роботи:

Ознайомлення з простими структурами даних,одновимірними масивами,двовимиріними масивами,алгоритмами обробки

### Теоретичні відомості:

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Прості структури даних
* Тема №2: Одновимірні масиви
* Тема №3 Двовимірні масиви

1. Індивідуальний план опрацювання теорії:

**Тема №1 Прості структури даних**

* + Джерела Інформації
    - https://acode.com.ua/urok-64-struktury/
  + Що опрацьовано:

Матеріал з джерел інформації

* + Статус: Ознайомлений
  + Початок опрацювання теми: 20.11.2023
  + Звершення опрацювання теми: 20.11.2023

**Тема №2 Одновимірні масиви**

* + Джерела Інформації:

https://www.bestprog.net/uk/2017/03/01/%D0%BC%D0%B0%D1%81%D0%B8%D0%B2%D0%B8-%D1%87%D0%B0%D1%81%D1%82%D0%B8%D0%BD%D0%B0-1-%D0%B2%D0%B8%D0%B7%D0%BD%D0%B0%D1%87%D0%B5%D0%BD%D0%BD%D1%8F-%D0%BC%D0%B0%D1%81%D0%B8%D0%B2%D1%83-%D0%BE%D0%B4/

* + Що опрацьовано:

Матеріал з джерел інформації

* + Статус: Ознайомлений
  + Початок опрацювання теми: 20.11.2023
  + Звершення опрацювання теми: 20.11.2023

**Тема №3 Двовимірні масиви**

* + Джерела Інформації

https://acode.com.ua/urok-81-bagatovymirni-masyvy/

Що опрацьовано:

Матеріал з джерел інформації

* + Статус: Ознайомлений
  + Початок опрацювання теми: 20.11.2023
  + Звершення опрацювання теми: 20.11.2023

### Виконання роботи:

**Завдання №1**

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

**Завдання №1**

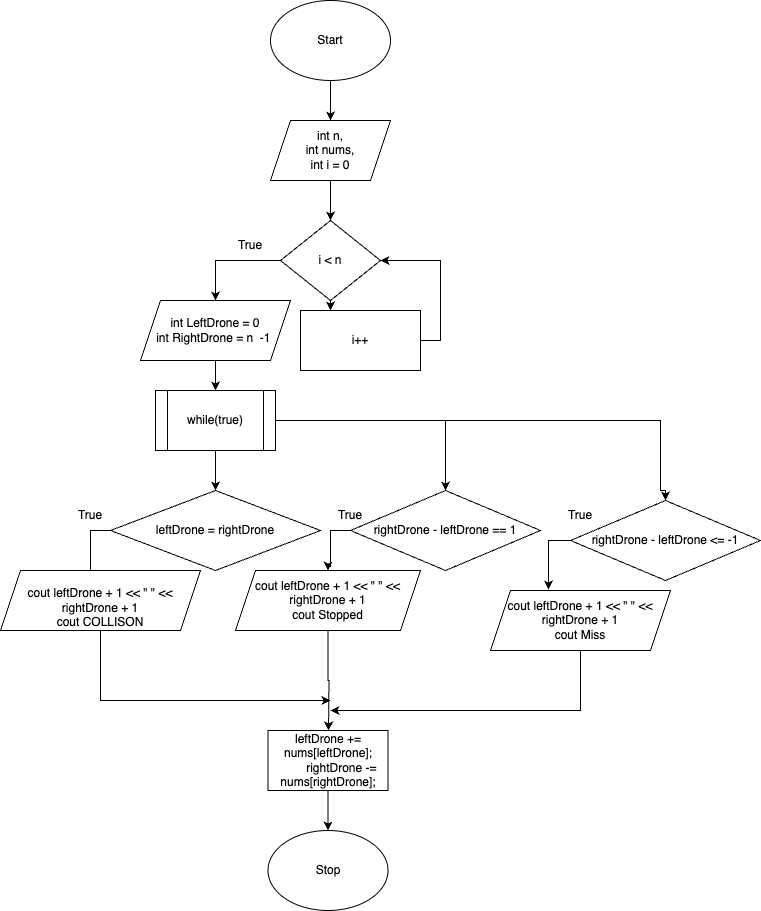
Тут вказуєте

-Варіант 3,Algolab 2

В даній задачі потрібно розглянути рух двох дронів відповідно до вказаних правил. Кожен дрон рухається відповідно до значень, що знаходяться в клітинах, і перевіряються умови взаємодії між ними: Почати рух обох дронів від першої та останньої клітинок відповідно. Перевіряти умови на кожному кроці руху: Якщо дрони опиняються в одній клітинці, вивести "Collision". Якщо лівий дрон опиниться справа від правого, вивести "Miss". Якщо обидва дрони зупиняються один навпроти одного, вивести "Stopped". Отже, програма повинна послідовно обчислювати рух обох дронів та перевіряти умови, вказані в завданні, в залежності від положення та взаємодії дронів.

Запланований час для виконання завдання: 1 день

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

**

1. *Код програми*

#include <iostream>

using namespace std;

int main() {

int n;

cin >> n;

int nums[n];

for (int i = 0; i < n; i++)

cin >> nums[i];

int leftDrone = 0, rightDrone = n - 1;

while(true) {

if (leftDrone == rightDrone)

{

cout << leftDrone + 1 << " " << rightDrone + 1 << endl;

cout << "Collision";

return 0;

}

else if (rightDrone - leftDrone == 1)

{

cout << leftDrone + 1 << " " << rightDrone + 1 << endl;

cout << "Stopped";

return 0;

}

else if (rightDrone - leftDrone <= -1)

{

cout << leftDrone + 1 << " " << rightDrone + 1 << endl;

cout << "Miss";

return 0;

}

leftDrone += nums[leftDrone];

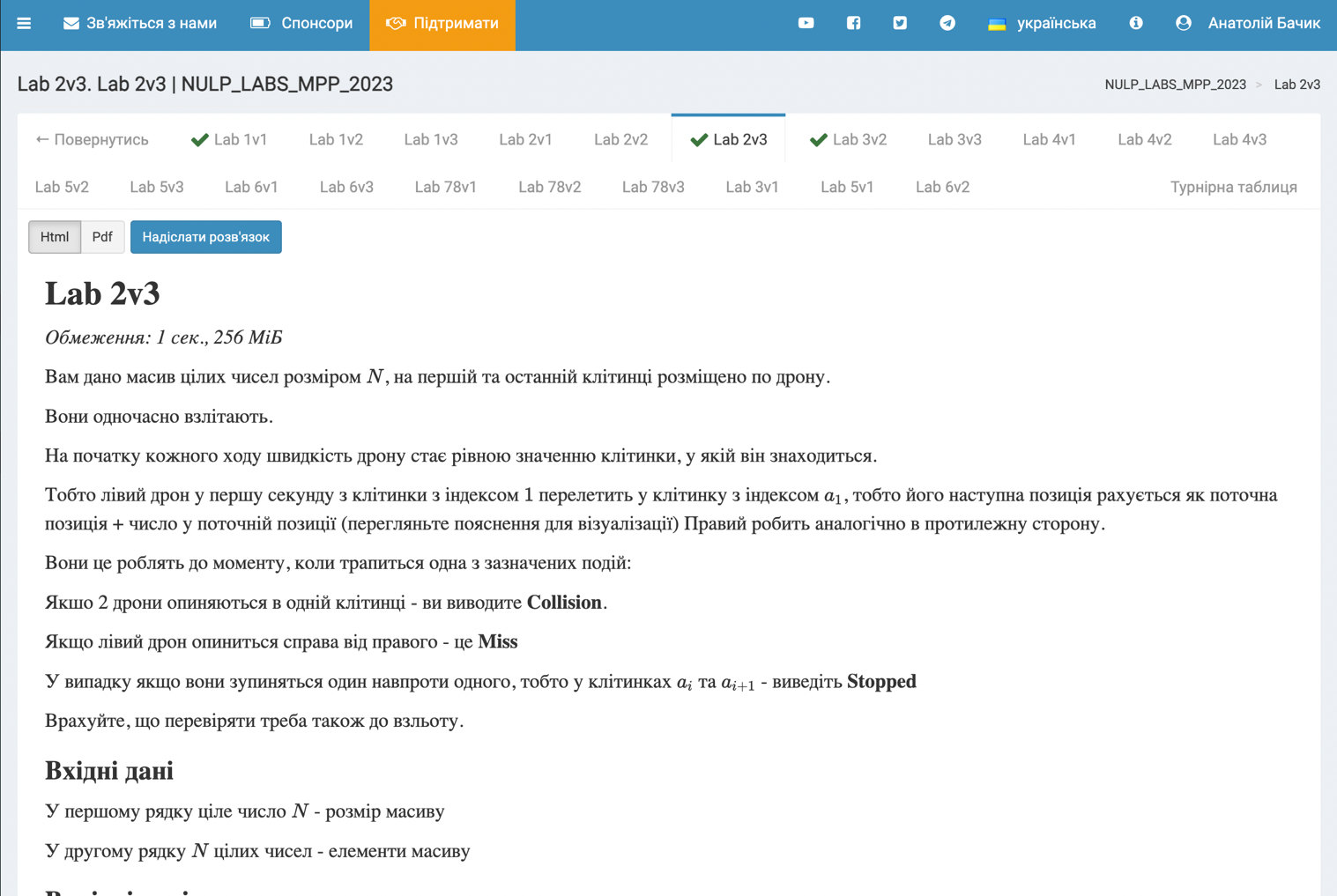
rightDrone -= nums[rightDrone];

}

return 0;

}

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

****



Потрачено часу: 1 день

**Завдання №2**

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

-Варіант 2,Algolab 3

- Обчислити функцію f(x), використовуючи її розклад в степеневий ряд

1. *Код програми*

#include <iostream>

#include <vector>

using namespace std;

int main() {

int N, M;

cin >> N;

vector<int> A(N);

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

cin >> A[i];

}

cin >> M;

vector<int> B(M);

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

cin >> B[i];

}

vector<int> countU(101, 0), countS(101, 0);

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

countU[A[i]]++;

}

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

countS[B[i]]++;

}

int common = 0, unique = 0;

for(int i = 0; i <= 100; i++) {

if(countU[i] > 0 && countS[i] > 0) {

common++;

}

if(countU[i] > 0 || countS[i] > 0) {

unique++;

}

}

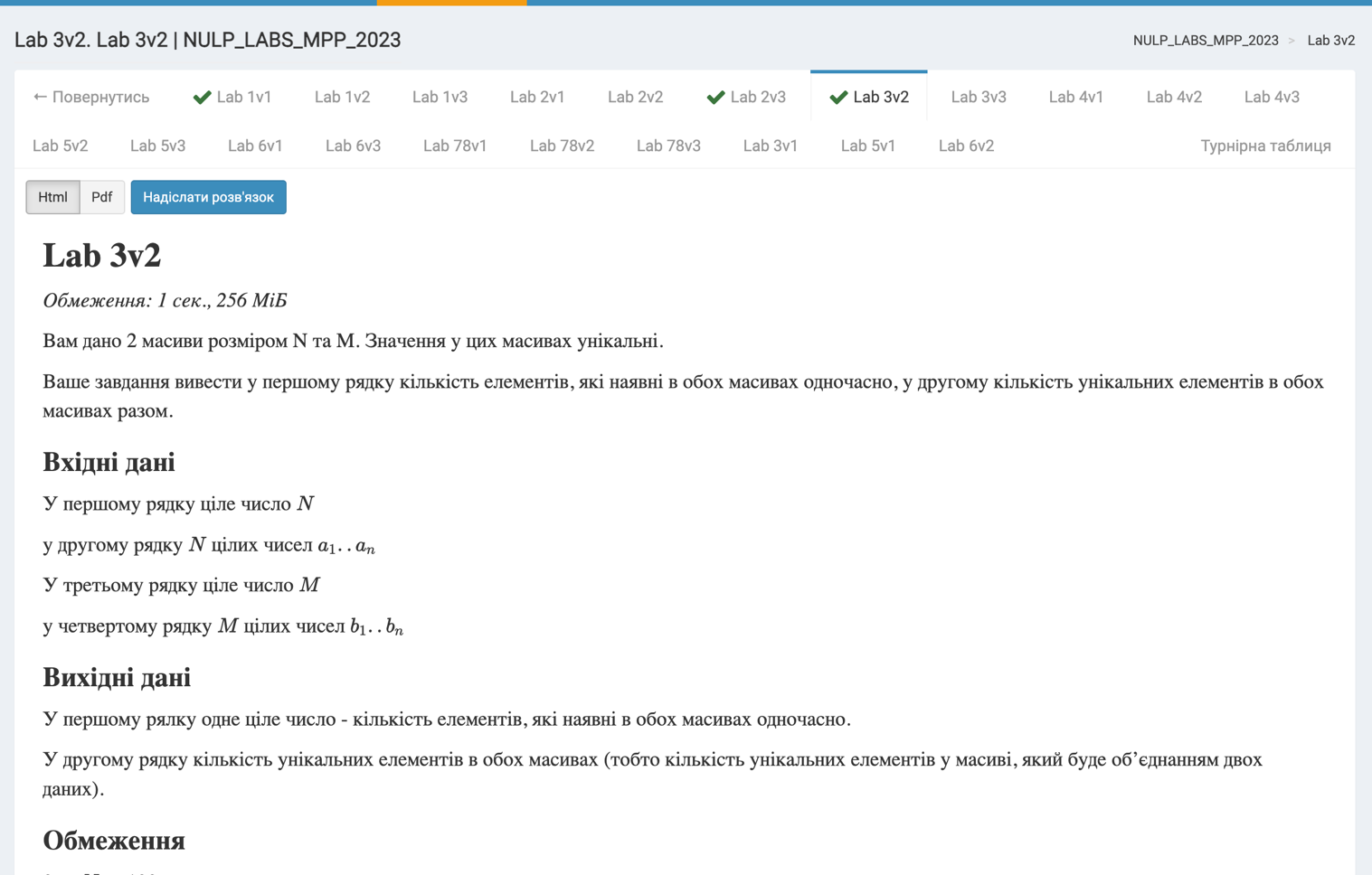
cout << common << endl;

cout << unique << endl;

return 0;

}

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



Потрачено часу: 1 день

**Завдання №3**

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

-Варіант 25,VNS Lab 4 Task 1

-

1) Реалізувати з використанням масиву однонаправлене кільце (перегляд

можливий зправа наліво, від першого елемента можна перейти до

останнього).

2) Роздрукувати отриманий масив, починаючи з К-ого елемента і до К+1.

3) Впорядкувати елементи по спаданню.

4) Знищити з кільця непарні елементи.

5) Роздрукувати отриманий масив, починаючи з К-ого елемента і до К+1.

1. *Код програми*

#include <iostream>

#include <vector>

#include <algorithm>

using namespace std;

void printElements(const vector<int>& ring, int k) { // Функція для виведення елементів кільця, починаючи з K-ого і до K+1

int size = ring.size();

for (int i = k; i != (k + 2) % size; i = (i + 1) % size) {

cout << ring[i] << " ";

}

cout << endl;

}

int main() {

int k;

cout << "Введіть розмір масиву: ";

int size;

cin >> size;

vector<int> ring(size);

cout << "Введіть елементи масиву: ";

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

cin >> ring[i];

}

cout << "Введіть K: ";

cin >> k;

cout << "Елементи кільця, починаючи з K-ого і до K+1: ";

printElements(ring, k);

sort(ring.rbegin(), ring.rend());// впорядкування елементів по спаданню

ring.erase(remove\_if(ring.begin(), ring.end(), [](int x) { return x % 2 != 0; }), ring.end()); // знищення непарних елементів

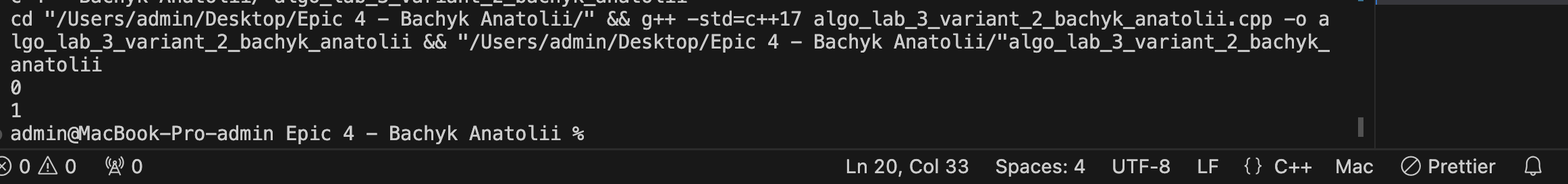
cout << "Елементи кільця після видалення непарних, починаючи з K-ого і до K+1: ";

printElements(ring, k);

return 0;

}

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

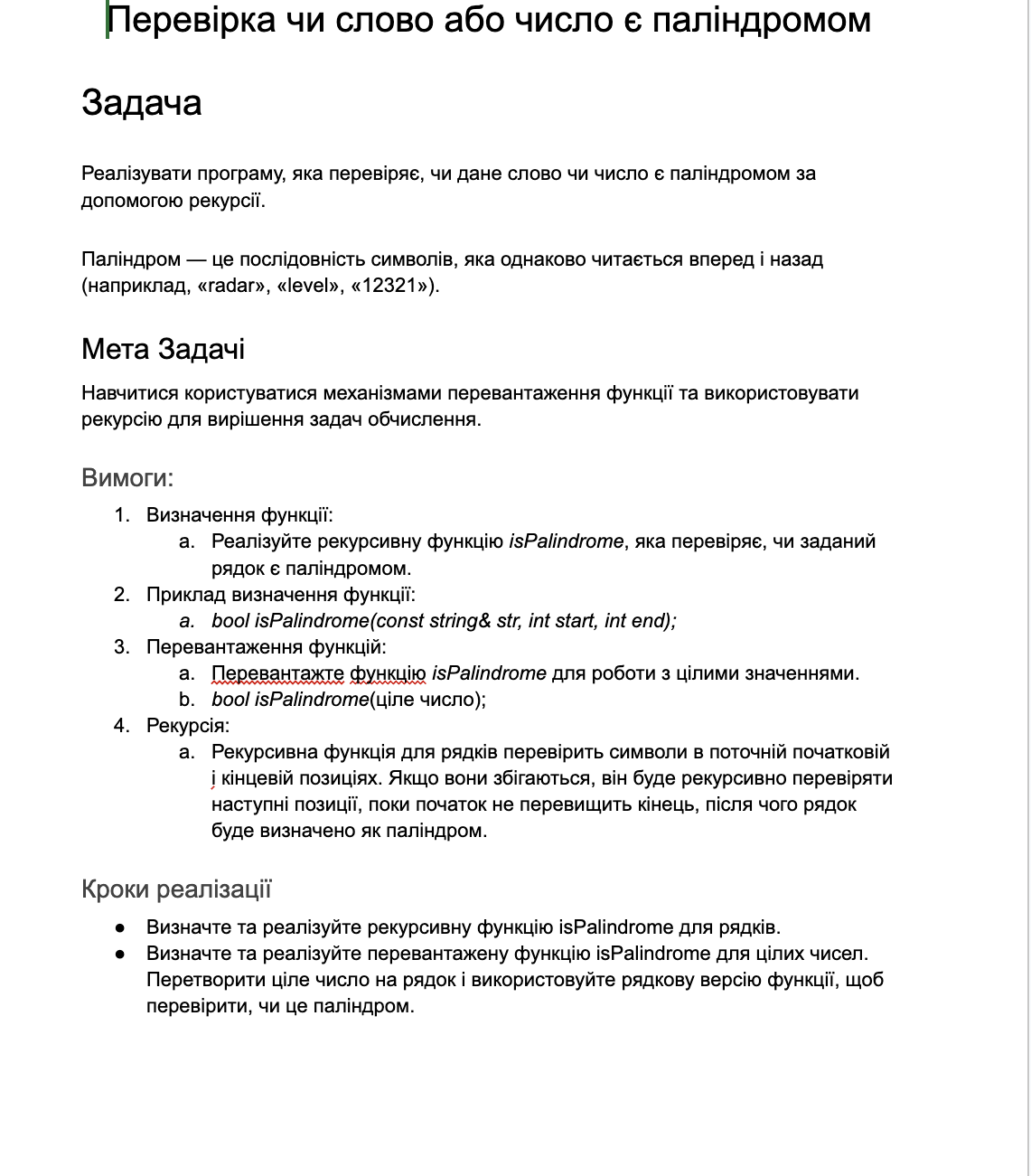


Потрачено часу: 1 день

**Завдання №4**

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

- Class Practice Work

- 

1. *Код програми*

#include <iostream>

#include <string>

using namespace std;

bool isPalindrome(const string& str, int start, int end) {

if (start >= end)

return true;

if (str[start] != str[end])

return false;

return isPalindrome(str, start + 1, end - 1);

}

bool isPalindrome(int num) {

string str = to\_string(num);

return isPalindrome(str, 0, str.length() - 1);

}

int main() {

string str;

cout << "Enter a string: ";

cin >> str;

if (isPalindrome(str, 0, str.length() - 1))

cout << str << " is a palindrome.\n";

else

cout << str << " is not a palindrome.\n";

int num;

cout << "Enter a number: ";

cin >> num;

if (isPalindrome(num))

cout << num << " is a palindrome.\n";

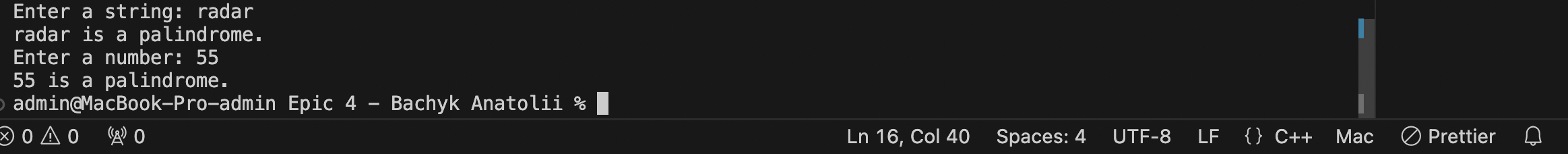
else

cout << num << " is not a palindrome.\n";

return 0;

}

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



Потрачено часу: 1 день

**Завдання №5**

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

-Self Practice Work

- У цьому завданні потрібно знайти мінімальну кількість символів, які можна видалити з рядка першого пінгвіна так, щоб він став лексикографічно меншим за рядок другого пінгвіна. Тобто, необхідно порівняти два рядки так, щоб перший рядок був "меншим" (відповідав умовам задачі) за другий. Необхідно вирахувати мінімальну кількість символів, які потрібно видалити з першого рядка для досягнення цієї мети.

1. *Код програми*

#include <iostream>

#include <string>

#include <algorithm>

using namespace std;

int minDeletions(string s1, string s2, int i, int j) {

// якщо один з рядків порожній, повертаємо довжину іншого рядка

if (i == s1.size()) return s2.size() - j;

if (j == s2.size()) return s1.size() - i;

if (s1[i] == s2[j]) return minDeletions(s1, s2, i + 1, j + 1); // Якщо символи співпадають, переходимо до наступного символу в обох рядках

return min(minDeletions(s1, s2, i + 1, j), minDeletions(s1, s2, i, j + 1)) + 1;// Якщо символи не співпадають, видаляємо символ з s1 або s2 і рекурсивно викликаємо функцію

}

int main() {

string s1 = "pinguin";

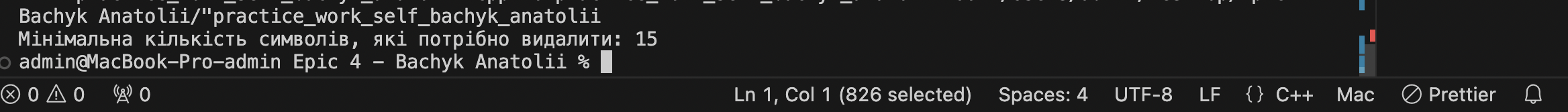
string s2 = "madagascar";

cout << "Мінімальна кількість символів, які потрібно видалити: " << minDeletions(s1, s2, 0, 0) << endl;

return 0;

}

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



Потрачено часу: *2 години*

**Завдання №5**

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

-Варіант 25,VNS Lab 5 Task 1

- Із двовимірного масиву в одновимірний записали спочатку рядки в

довільному порядку, потім стовпці в довільному порядку. Написати

програму, яка відновлює вихідний двовимірний масив маючи одновимірний,

якщо відомо розмірність двовимірного масиву й елементи в ньому не

повторюються.

1. *Код програми*

#include <iostream>

#include <vector>

using namespace std;

// Функція для відновлення двовимірного масиву з одновимірного

vector<vector<int>> restoreMatrix(vector<int>& arr, int rows, int cols) {

vector<vector<int>> matrix(rows, vector<int>(cols)); // Створюємо двовимірний масив

int index = 0;

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

for (int j = 0; j < cols; ++j) {

matrix[i][j] = arr[index++]; // Копіюємо елементи з одновимірного масиву в двовимірний

}

}

for (int j = 0; j < cols; ++j) {

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

matrix[i][j] = arr[index++]; // Копіюємо елементи з одновимірного масиву в двовимірний

}

}

return matrix;

}

int main() {

vector<int> arr = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}; // Одновимірний масив

int rows = 3;

int cols = 4;

vector<vector<int>> matrix = restoreMatrix(arr, rows, cols); // Відновлюємо двовимірний масив

for (const auto& row : matrix) {

for (const auto& elem : row) {

cout << elem << ' ';

}

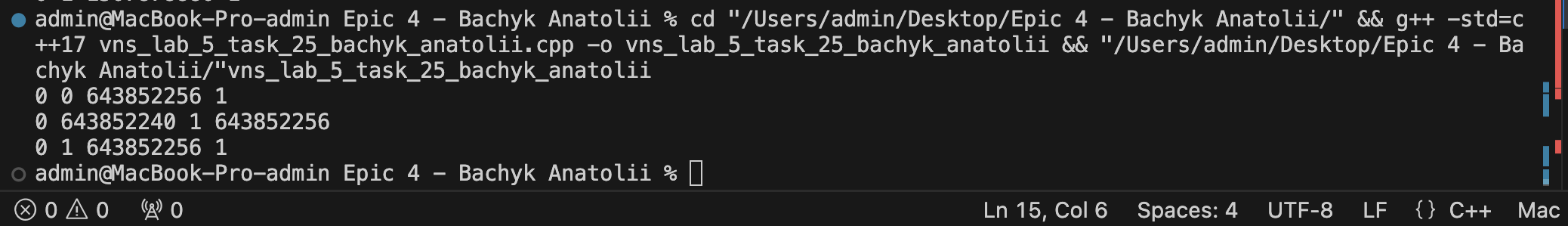
cout << '\n';

}

return 0;

}

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



Потрачено часу: *2 дня*

### Висновки:

Ознайомленений з простими структурами даних,одновимірними масивами,двовимиріними масивами,алгоритмами обробки.Часткове використання їх в поставлених завданнях