

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра систем штучного інтелекту



Звіт

про виконання лабораторних та практичних робіт блоку № 4

На тему: «Одновимірні масиви. Двовимірні Масиви. Вказівники та
Посилання. Динамічні масиви. Структури даних. Вкладені структури.
Алгоритми обробки та робота з масивами та структурами.»

з дисципліни: «Основи програмування»

до:

ВНС Лабораторної Роботи №4

ВНС Лабораторної Роботи №5

Алготестер Лабораторної Роботи №2

Алготестер Лабораторної Роботи №3

Практичних Робіт до блоку №4

Виконав:

Студент групи ШІ-11
Маркевич Владислав

EPIC 4

“Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та робота з масивами та структурами.”

Tasks - (Задачі 4 ітерації в Trello або Any Tasks Manager):

- Task 1 - Theory Education Activities
- Task 2 - Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-8)
- Task 3 - Lab# programming: VNS Lab 4
- Task 4 - Lab# programming: VNS Lab 5
- Task 5 - Lab# programming: Algotester Lab 2
- Task 6 - Lab# programming: Algotester Lab 3
- Task 7 - Practice# programming: Class Practice Task
- Task 8 - Practice# programming: Self Practice Task
- Task 9 - Result Documentation Report and Outcomes Placement Activities (Docs and Programs on GitHub)
- Task 10 - Results Evaluation and Release

Sources:

<http://cpp.dp.ua/klasy-pam-yati-u-c-builder/>

<https://www.youtube.com/watch?v=KHK5bMp6vbo>

<https://acode.com.ua/urok-86-vkazivnyky-i-masyvy/>

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

<https://acode.com.ua/urok-64-struktury/>

<https://acode.com.ua/urok-84-vkazivnyky/>

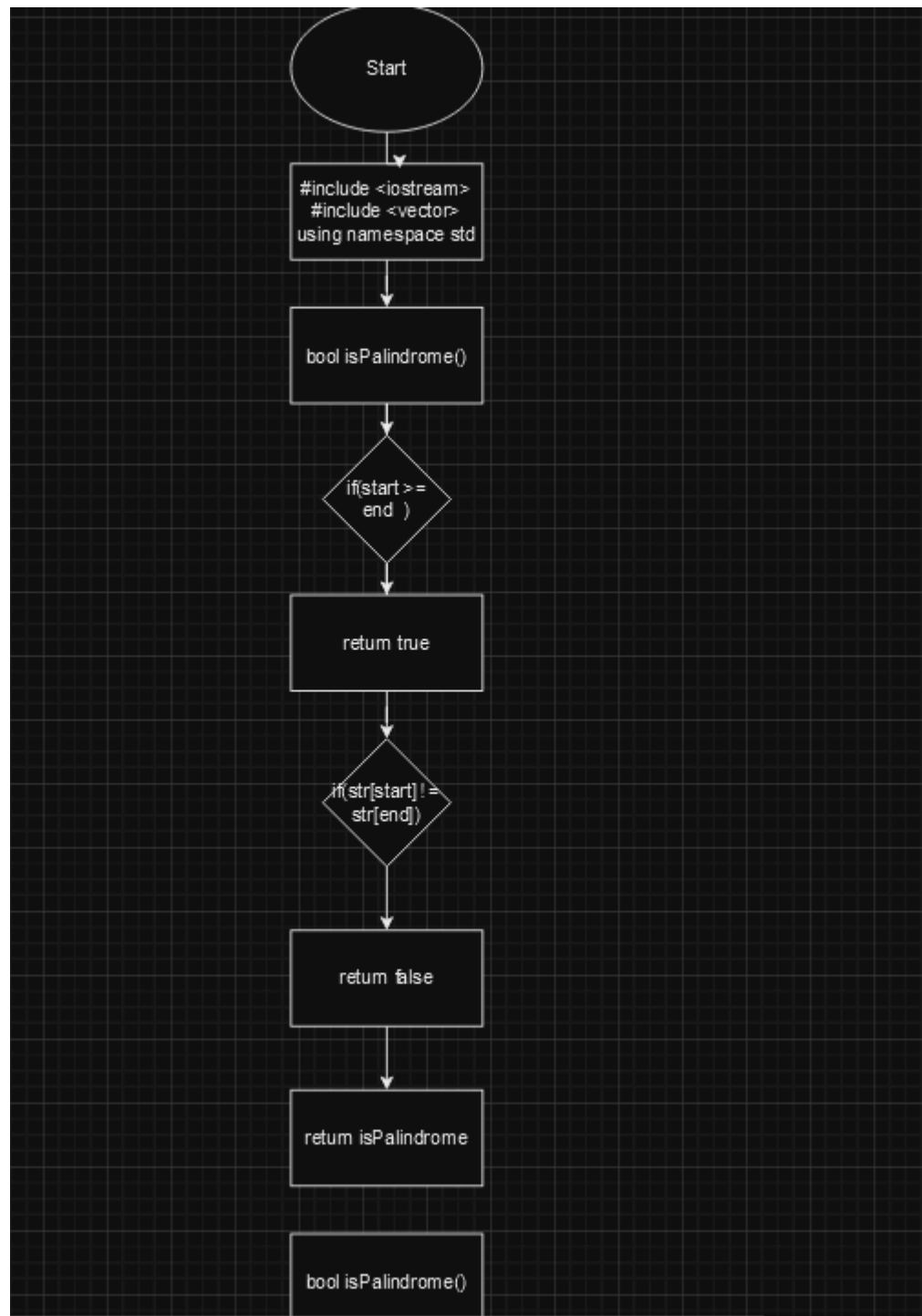
<https://acode.com.ua/urok-92-posylannya/>

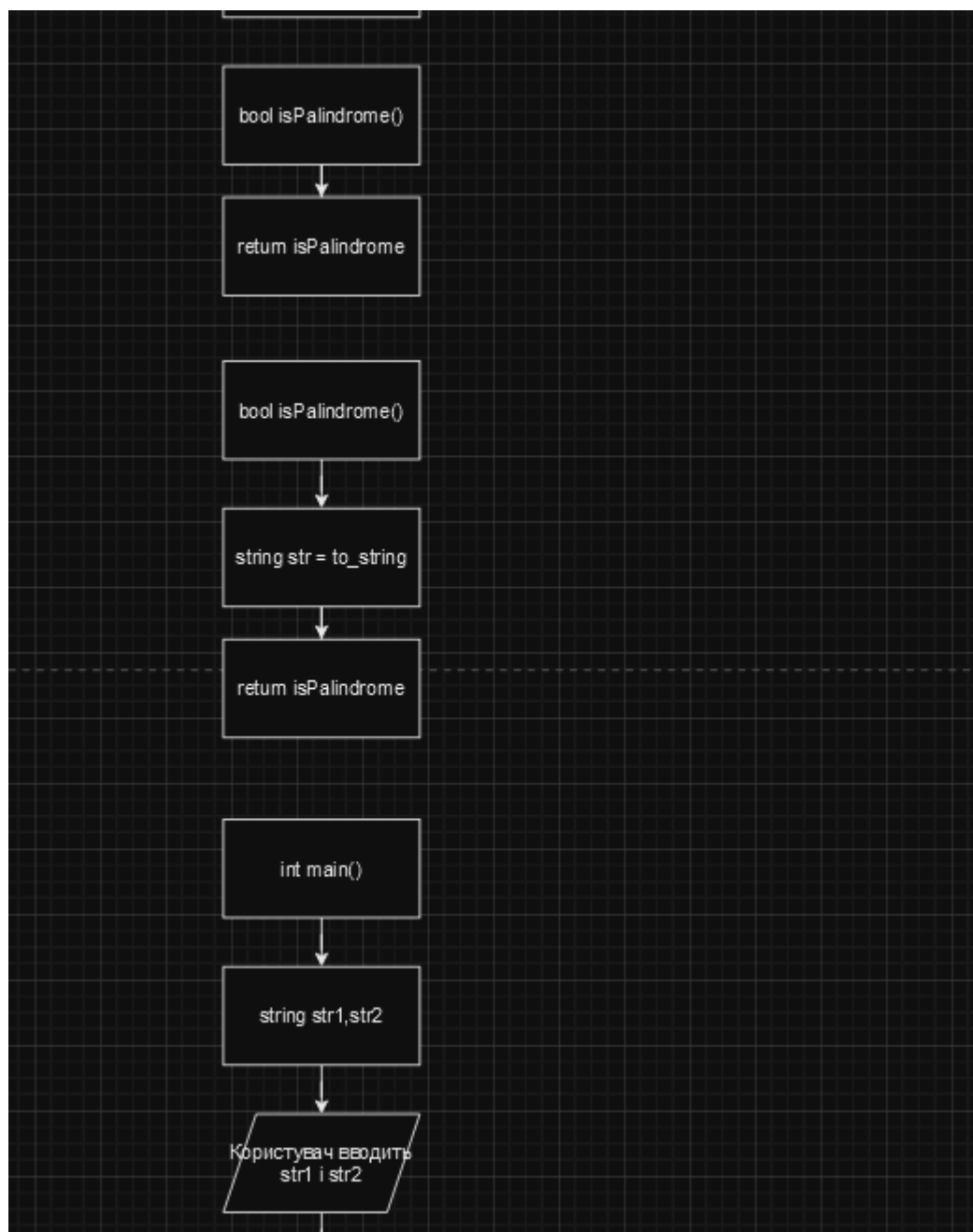
Task 1 - Theory Education Activities

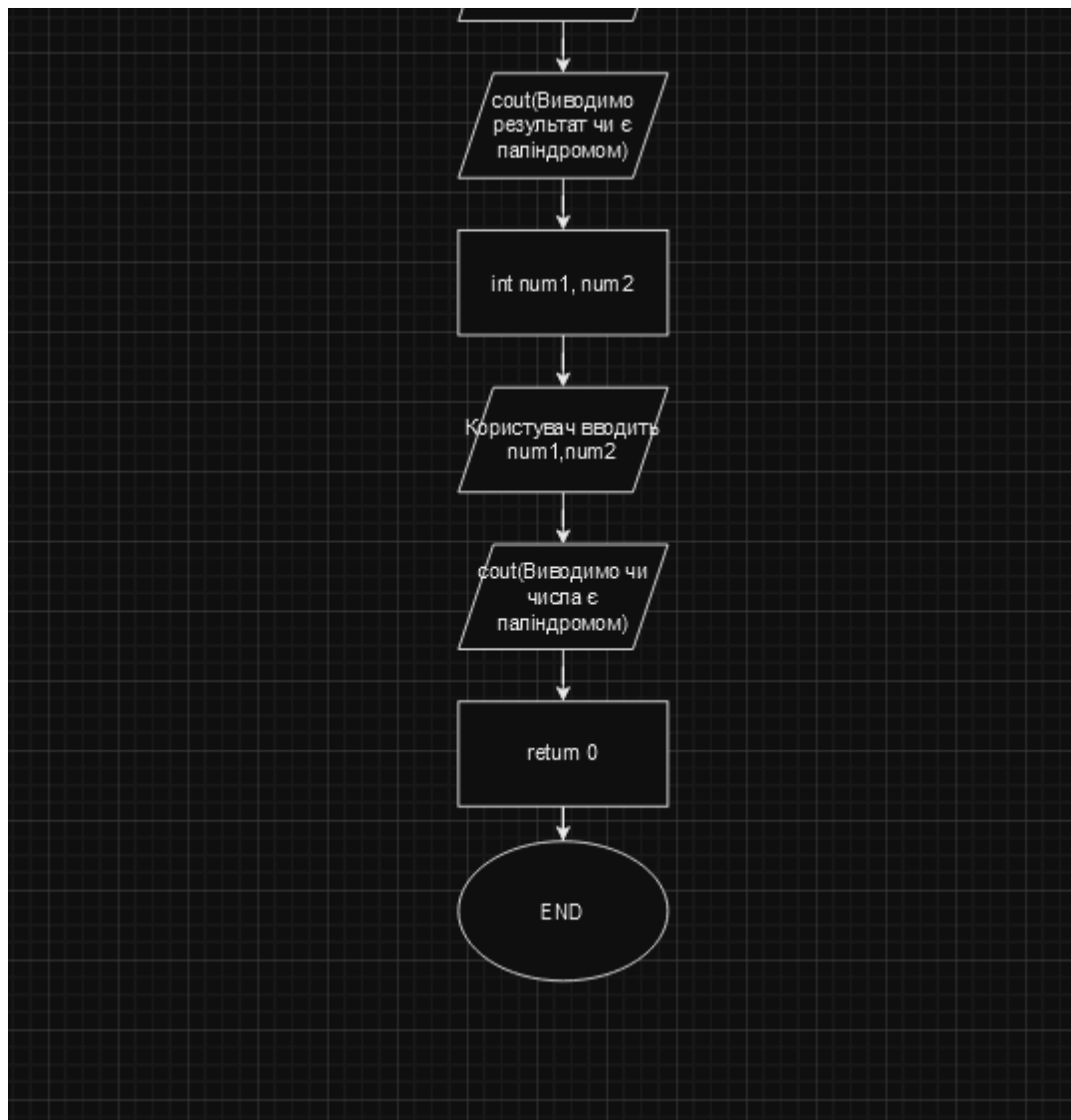
1. Класи пам'яті у C++
 - Статична пам'ять.
 - Динамічна пам'ять.
 - Поняття стеку.
 - Виділення та вивільнення пам'яті.
0. Вступ до Масивів і Вказівників:
 - Основи масивів: визначення, важливість, приклади використання.
 - Різниця між статичними та динамічними масивами.
 - Основи вказівників: що це таке, як вони працюють.
 - Взаємозв'язок між масивами та вказівниками.
 - Вступ до посилань: основні концепції та відмінності від вказівників.
0. Одновимірні Масиви:
 - Створення та ініціалізація одновимірних масивів.
 - Основні операції: індексація, присвоєння, читання.
 - Цикли та обхід масивів.
 - Використання функцій для роботи з масивами.
 - Приклади алгоритмів сортування та пошуку.
0. Вказівники та Посилання:
 - Використання вказівників для доступу до елементів масиву.
 - Арифметика вказівників.
 - Різниця між вказівниками та посиланнями в контексті функцій.
 - Динамічне виділення пам'яті з використанням вказівників.
 - Використання вказівників для створення складних структур даних.
0. Двовимірні Масиви:
 - Оголошення та ініціалізація двовимірних масивів.
 - Вкладені цикли для обходу двовимірних масивів.
 - Практичні приклади використання двовимірних масивів.
 - Передача двовимірних масивів у функції.
 - Застосування двовимірних масивів для розв'язання задач.
0. Динамічні Масиви:
 - Основи динамічного виділення пам'яті.
 - Створення та управління динамічними масивами.
 - Використання операторів new та delete для управління пам'яттю.
 - Реалізація змінної розмірності масивів.
 - Передача динамічних масивів у функції.
0. Структури Даних:
 - Оголошення та використання структур.
 - Використання масивів та вказівників у структурах.
 - Функції для обробки даних у структурах.
 - Використання структур для представлення складних даних.
 - Вкладені структури та їх використання.
 - Об'єднання (Union)
 - Переліки (enumerations)
0. Вкладені Структури:
 - Поняття вкладених структур та їх оголошення.
 - Взаємодія з вкладеними структурами.
 - Використання вкладених структур для моделювання складних даних.
 - Передача вкладених структур у функції.
 - Приклади реального використання вкладених структур.
9. Використання структур
 - Перевантаження операторів у структурі.

- Вивід/ввід структури (operator<<);
 - Арифметичні операції з структурами (operator+, operator-);
 - Практичні задачі на виведення структур та операції з ними
0. Алгоритми обробки та робота з масивами та структурами:
- Алгоритми пошуку та сортування в масивах.
 - Обробка та маніпуляції з даними у структурах.
 - Використання циклів та умовних операторів для роботи з масивами та структурами.
 - Інтеграція масивів та структур у алгоритми.
 - Розв'язання практичних задач з використанням масивів та структур.

Task 2 - Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-8)







Task 3 - VNS Lab 4

V – 7

7.

- 1) Сформувати одновимірний масив цілих чисел, використовуючи генератор випадкових чисел.
- 2) Роздрукувати отриманий масив.
- 3) Знищити останній елемент, який дорівнює 0.
- 4) Додати після елемента масиву із заданим індексом елемент зі значенням 100.
- 5) Роздрукувати отриманий масив.

```

#include <iostream>
#include <vector>
#include <cstdlib>
#include <ctime>

using namespace std;

int main() {
    srand(time(0));
    vector<int> array;
    int n = 10;
    for (int i = 0; i < n; i++) {
        array.push_back(rand() % 11);
    }

    cout << "Початковий масив: ";
    for (int num : array) {
        cout << num << " ";
    }
    cout << endl;

    for (int i = array.size() - 1; i >= 0; i--) {
        if (array[i] == 0) {
            array.erase(array.begin() + i);
            break;
        }
    }

    cout << "Після видалення останнього 0: ";
    for (int num : array) {
        cout << num << " ";
    }
    cout << endl;

    int index;
    cout << "Введіть індекс для додавання елемента після: ";
    cin >> index;

```

```

    cout << endl;

    int index;
    cout << "Введіть індекс для додавання елемента після: ";
    cin >> index;

    if (index >= 0 && index < array.size()) {
        array.insert(array.begin() + index + 1, 100);
    } else {
        cout << "Неправильний індекс!" << endl;
    }

    cout << "Остаточний масив: ";
    for (int num : array) {
        cout << num << " ";
    }
    cout << endl;

    return 0;
}

```

```

Початковий масив: 4 5 10 8 5 0 3 4 8 3
Після видалення останнього 0: 4 5 10 8 5 3 4 8 3
Введіть індекс для додавання елемента після: 8
Остаточний масив: 4 5 10 8 5 3 4 8 3 100

```


Task 4 - VNS Lab 5

V – 7

7. Написати процедуру обміну стовпця й рядка двовимірного масиву. З її допомогою поміняти місцями ті рядки й стовпці, перші елементи яких збігаються.

```
#include <iostream>
using namespace std;

void swapRowAndColumn(int matrix[][4], int size, int row, int col) {
    for (int i = 0; i < size; i++) {
        swap(matrix[row][i], matrix[i][col]);
    }
}

void printMatrix(int matrix[][4], int rows, int cols) {
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }
}

int main() {
    const int size = 4;
    int matrix[size][size] = {
        {2, 1, 3, 4},
        {5, 2, 6, 7},
        {8, 9, 2, 1},
        {2, 4, 5, 6}
    };

    cout << "Original matrix:" << endl;
    printMatrix(matrix, size, size);

    for (int i = 0; i < size; i++) {
        for (int j = 0; j < size; j++) {
            if (matrix[i][0] == matrix[0][j]) {
                swapRowAndColumn(matrix, size, i, j);
            }
        }
    }
}
```

```
    cout << "\nMatrix after swapping rows and columns:" << endl;
    printMatrix(matrix, size, size);

    return 0;
}
```

```
ERROR -14p1nvx1.43y --pid=Microsoft-MEngine-Pid-qsqkykcl.c
Original matrix:
2 1 3 4
5 2 6 7
8 9 2 1
2 4 5 6

Matrix after swapping rows and columns:
2 7 1 6
1 6 2 5
3 9 2 8
4 4 5 2
PS C:\ai_programming_playground_2024>
```

Task 5 - Algotester Lab 2

V - 3

```

#include <iostream>
using namespace std;

int main() {
    int N;
    cin >> N;
    vector<int> a(N);

    for (int i = 0; i < N; ++i) {
        cin >> a[i];
    }

    int left = 0;
    int right = N - 1;

    while (true) {
        if (left == right) {
            cout << left + 1 << " " << right + 1 << endl;
            cout << "Collision" << endl;
            break;
        }
        else if (left + 1 == right) {
            cout << left + 1 << " " << right + 1 << endl;
            cout << "Stopped" << endl;
            break;
        }
        else if (left > right) {
            cout << left + 1 << " " << right + 1 << endl;
            cout << "Miss" << endl;
            break;
        }

        left += a[left];
        right -= a[right];
    }

    return 0;
}

```

```

10
1 3 1 1 5 1 1 2 1 2
5 6
Stopped

```

Task 6 - Algotester Lab 3

V – 3

```
#include <iostream>
#include <string>
using namespace std;

string compressString(const string& s) {
    string compressed;
    int n = s.length();

    for (int i = 0; i < n; ) {
        char currentChar = s[i];
        int count = 0;

        while (i < n && s[i] == currentChar) {
            count++;
            i++;
        }

        compressed += currentChar;
        if (count > 1) {
            compressed += to_string(count);
        }
    }

    return compressed;
}

int main() {
    string s;
    cin >> s;

    string compressed = compressString(s);
    cout << compressed << endl;

    return 0;
}
```

SSSSSFFFFAAAFAAAAA
S5F5A3FA6

Task 7 - Class Practice Task

Задача

Реалізувати програму, яка перевіряє, чи дане слово чи число є паліндромом за допомогою рекурсії.

```
#include <iostream>
#include <vector>

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(const string& str){
    return isPalindrome(str, 0, str.length() - 1);
}

bool isPalindrome(int numbers){
    string str = to_string(numbers);
    return isPalindrome(str);
}

int main(){
    string str1;
    string str2;
    cout<<"Введіть два слова: "<<endl;
    cin>>str1;
    cin>>str2;

    cout<<"Чи "<<str1<<" є паліндромом?\n"<<(isPalindrome(str1) ? "Так" : "Ні")<<endl;;
    cout<<"Чи "<<str2<<" є паліндромом?\n"<<(isPalindrome(str2) ? "Так" : "Ні")<<endl;

    int num1;
    int num2;

    cout<<"Введіть два числа:"<<endl;
    cin>>num1;
    cin>>num2;

    cout<<"Чи "<<num1<<" є паліндромом?\n"<<(isPalindrome(num1) ? "Так" : "Ні")<<endl;;
    cout<<"Чи "<<num2<<" є паліндромом?\n"<<(isPalindrome(num2) ? "Так" : "Ні");

    return 0;
}
```

```
Введіть два слова:  
level  
ex  
Чи level є паліндромом?  
Так  
Чи ex є паліндромом?  
Ні  
Введіть два числа:  
120021  
1212121  
Чи 120021 є паліндромом?  
Так  
Чи 1212121 є паліндромом?  
Так
```

Task 8 - Self Practice Task

```

#include <iostream>
#include <vector>
#include <algorithm>

using namespace std;

int main() {
    int n, k;
    cin >> n >> k;

    vector<int> a(n);
    for (int i = 0; i < n; i++) {
        cin >> a[i];
    }

    int maxLength = 0;
    int currentLength = 0;

    for (int i = 0; i < n; i++) {
        if (a[i] >= k) {
            currentLength++;
        } else {
            maxLength = max(maxLength, currentLength);
            currentLength = 0;
        }
    }

    maxLength = max(maxLength, currentLength);

    cout << maxLength << endl;

    return 0;
}

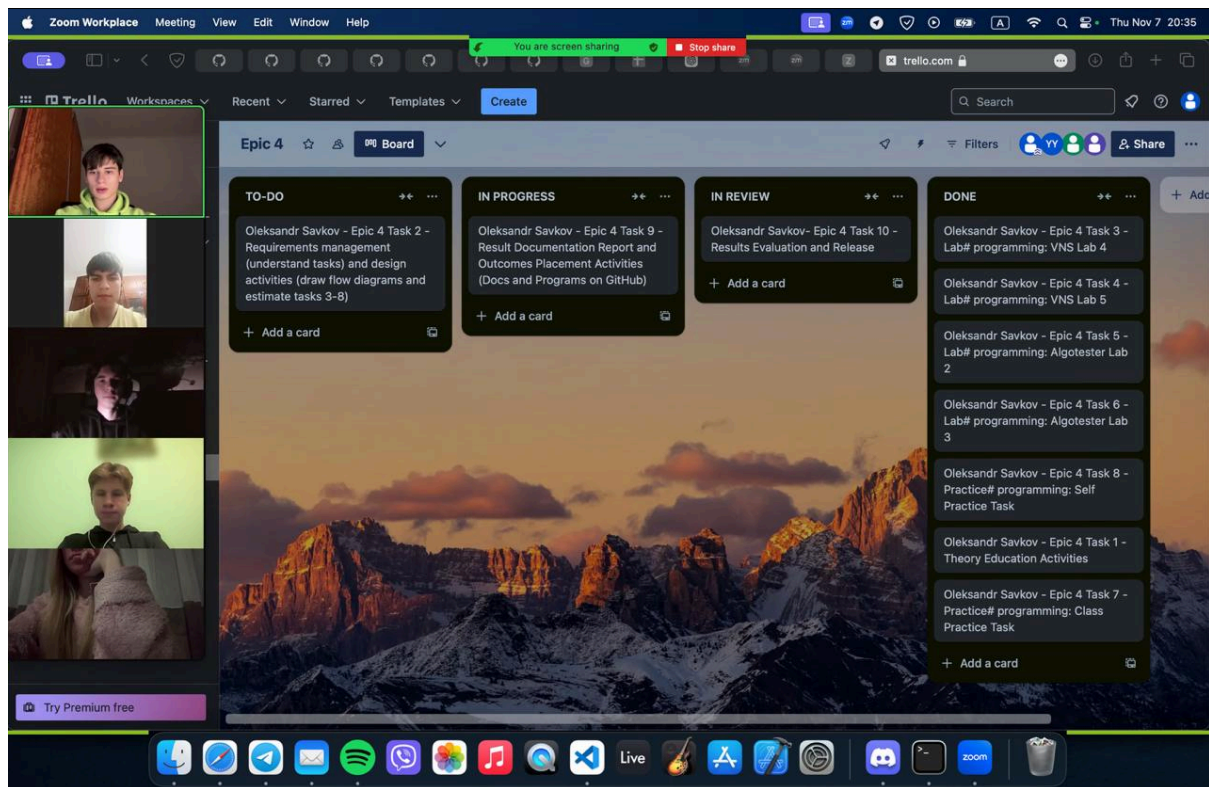
```

```

7 4
7 1 4 7 6 3 4
3

```

Робота з командою:



Висновок: у цій лабораторній роботі я ознайомився з основами роботи з масивами, вказівниками, посиланнями, динамічними масивами та структурами даних у C++. Навчився використовувати одновимірні й двовимірні масиви для зберігання та обробки даних, працювати з вказівниками для управління пам'яттю та створювати динамічні масиви невідомого наперед розміру. Робота зі структурами та вкладеними структурами допомогла зрозуміти, як організовувати складніші дані, а застосування алгоритмів обробки — як ефективно працювати з ними.