

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

Виконала:

Студентка групи ШІ-12

Хвостова Олександра Андріївна

Тема роботи: Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та робота з масивами та структурами.

Мета роботи: Ознайомлення з основами та практичними аспектами роботи з одновимірними та двовимірними масивами, вказівниками та посиланнями, динамічними масивами, структурами даних і вкладеними структурами. Це включає в себе вивчення різних алгоритмів обробки даних, методів виділення і управління пам'яттю в мові програмування C++, а також застосування цих знань для розв'язання практичних задач.

Висновок

Теоретичні відомості:

1. Класи пам'яті у C++

- Джерела Інформації
 - Стаття. <https://acode.com.ua/urok-89-dynamichne-vydilennya-pam-yati/>
 - Стаття. <https://acode.com.ua/urok-111-stek-i-kupa/>
 - Відео. <https://youtu.be/NyOjKd5Qruk?si=mm3Q-c93h78leg0V>
- Що опрацьовано:
 - Статична пам'ять.
 - Динамічна пам'ять.
 - Поняття стеку.
 - Виділення та вивільнення пам'яті.
- Статус: Ознайомлений
- Початок опрацювання теми: 10.11.2024
- Звершення опрацювання теми: 10.11.2024

2. Вступ до Масивів і Вказівників:

- Джерела Інформації
 - Стаття. <https://acode.com.ua/urok-77-masyvy/>
 - Стаття. <https://acode.com.ua/urok-84-vkazivnyky/>
 - Стаття. <https://acode.com.ua/urok-86-vkazivnyky-i-masyvy/>
 - Стаття. <https://acode.com.ua/urok-90-dynamichni-masyvy/>
 - Стаття. <https://acode.com.ua/urok-92-posylannya/>
- Що опрацьовано:
 - 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.
- Статус: Ознайомлений
- Початок опрацювання теми: 11.11.2024
- Звершення опрацювання теми: 11.11.2024

3. Одновимірні Масиви

- Джерела Інформації
 - Стаття. <https://acode.com.ua/urok-78-fiksovani-masyvy/>
 - Відео. https://www.youtube.com/watch?v=YFLRN_Gmh4o
- Що опрацьовано:
 - 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.
- Статус: Ознайомлений
- Початок опрацювання теми: 11.11.2024
- Звершення опрацювання теми: 11.11.2024

4. Вказівники та Посилання

- Джерела Інформації
 - Стаття. <https://acode.com.ua/urok-78-fiksovani-masyvy/>
 - Відео. https://www.youtube.com/watch?v=YFLRN_Gmh4o
- Що опрацьовано:
 - 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.
- Статус: Ознайомлений
- Початок опрацювання теми: 11.11.2024
- Звершення опрацювання теми: 11.11.2024

5. Двовимірні Масиви

- Джерела Інформації
 - Стаття. <https://acode.com.ua/urok-81-bagatovymirni-masyvy/>
 - Відео. https://www.youtube.com/watch?v=YFLRN_Gmh4o
- Що опрацьовано:
 - 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.
- Статус: Ознайомлений
- Початок опрацювання теми: 12.11.2024
- Звершення опрацювання теми: 12.11.2024

6. Динамічні Масиви:

- Джерела Інформації
 - Стаття. <https://acode.com.ua/urok-89-dynamichne-vydilennya-pam-yati/>
 - Стаття. <https://acode.com.ua/urok-90-dynamichni-masyvy/>
- Що опрацьовано:
 - 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.
- Статус: Ознайомлений
- Початок опрацювання теми: 12.11.2024
- Звершення опрацювання теми: 12.11.2024

7. Структури Даних:

- Джерела Інформації
 - Стаття. <https://acode.com.ua/urok-64-strukturny/>
 - Стаття. <https://www.geeksforgeeks.org/structures-unions-and-enumerations-in-cpp/>

- Стаття. https://abitap.com/2-12-aryfmetychni-operacziyi/#google_vignette
- Що опрацьовано:
 - 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)
- Статус: Ознайомлений
- Початок опрацювання теми: 13.11.2024
- Звершення опрацювання теми: 13.11.2024

Виконання роботи:

1. Опрацювання завдання та вимог до програм та середовища:

Завдання №1. VNS. Лабораторна робота №4. Варіант 7

- Деталі завдання
 - 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.

- Важливі деталі для врахування в імплементації програми

1) При виконанні роботи використовуються статичні масиви. Для організації статичних масивів із псевдозмінними межами необхідно оголосити масив досить великої довжини, наприклад, 100 елементів:

```
int N=100;
int a[N];
```

Потім користувач вводить реальну довжину масиву (не більше N) і працює з масивом тієї довжини, що він сам вказав. Інші елементи (хоча пам'ять під них і буде виділена) не розглядаються.

2) При зменшенні або збільшенні довжини масиву необхідно змінювати його реальну довжину.

Завдання №2. VNS. Лабораторна робота №5. Варіант 7

- Деталі завдання

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

- Важливі деталі для врахування в імплементації програми

Використовуючи функції, розв'язати зазначене у варіанті завдання. Масив повинен передаватися у функцію як параметр.

Завдання №3. Algotester. Лабораторна робота №2. Варіант 3

- Деталі завдання

Вам дано масив цілих чисел розміром NN, на першій та останній клітинці розміщено по дрону.

Вони одночасно взлітають.

На початку кожного ходу швидкість дрону стає рівною значенню клітинки, у якій він знаходиться.

Тобто лівий дрон у першу секунду з клітинки з індексом 11 перелетить у клітинку з індексом $a[1]$, тобто його наступна позиція рахується як поточна позиція + число у поточній позиції (перегляньте пояснення для візуалізації) Правий робить аналогічно в протилежну сторону.

Вони це роблять до моменту, коли трапиться одна з зазначених подій:

Якщо 2 дрони опиняться в одній клітинці - ви виводите Collision.

Якщо лівий дрон опиниться справа від правого - це Miss

У випадку якщо вони зупиняться один навпроти одного, тобто у клітинках $a[i]$ та $a[i+1]$ - виведіть Stopped

Врахуйте, що перевіряти треба також до взльоту.

- Важливі деталі для врахування в імплементації програми
 - Вхідні дані
 - У першому рядку ціле число NN - розмір масиву
 - У другому рядку NN цілих чисел - елементи масиву
 - Вихідні дані
 - У першому рядку фінальна позиція першого та другого дрона.
 - У другому рядку одне зі слів:
Collision
Miss
Stopped

Завдання №4. Algotester. Лабораторна робота №3. Варіант 3

- Деталі завдання

Вам дана стрічка s .

Ваше завдання зробити компресію стрічки, тобто якщо якась буква йде більше одного разу підряд у стрічці замінити її на букву + кількість входжень підряд.

- Важливі деталі для врахування в імплементації програми

Великі та маленькі букви вважаються різними.

Завдання №5. Практична робота

- Деталі завдання

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

Паліндром — це послідовність символів, яка однаково читається вперед і назад (наприклад, «radar», «level», «12321»).

Мета Задачі

Навчитися користуватися механізмами перевантаження функції та використовувати рекурсію для вирішення задач обчислення.

Визначте та реалізуйте рекурсивну функцію isPalindrome для рядків.

Визначте та реалізуйте перевантажену функцію isPalindrome для цілих чисел.

Використати математичний підхід щоб перевірити чи число є паліндромом.

- Важливі деталі для врахування в імплементації програми

Вимоги:

- Визначення функції:

Реалізуйте рекурсивну функцію isPalindrome, яка перевіряє, чи заданий рядок є паліндромом.

- Приклад визначення функції:

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

- Перевантаження функції:

Перевантажте функцію isPalindrome для роботи з цілими значеннями.

bool isPalindrome(ціле число);

- Рекурсія:

- Планований час на реалізацію: 2 години
- Важливі деталі для врахування в імплементації
 - 1) При виконанні роботи використовуються статичні масиви. Для організації статичних масивів із псевдозмінними межами необхідно оголосити масив досить великої довжини, наприклад, 100 елементів:
 - `int N=100;`
 - `int a[N];`
 - Потім користувач вводить реальну довжину масиву (не більше N) і працює з масивом тієї довжини, що він сам вказав. Інші елементи (хоча пам'ять під них і буде виділена) не розглядаються.
 - 2) При зменшенні або збільшенні довжини масиву необхідно змінювати його реальну довжину.

Завдання №2. VNS. Лабораторна робота №5. Варіант 7

- Планований час на реалізацію: 2 години
- Важливі деталі для врахування в імплементації

Використовуючи функції, розв'язати зазначене у варіанті завдання. Масив повинен передаватися у функцію як параметр.

Завдання №3. Algotester. Лабораторна робота №2. Варіант 3

- Планований час на реалізацію: 2 години
- Важливі деталі для врахування в імплементації:

Вхідні дані

У першому рядку ціле число NN - розмір масиву

У другому рядку NN цілих чисел - елементи масиву

Вихідні дані

У першому рядку фінальна позиція першого та другого дрона.

У другому рядку одне зі слів:

Collision

Miss

Stopped

Завдання №4. Algotester. Лабораторна робота №3. Варіант 3

- Планований час на реалізацію: 2 години
- Важливі деталі для врахування в імплементації:

Великі та маленькі букви вважаються різними.

Завдання №5. Практична робота

- Планований час на реалізацію: 2 години
- Важливі деталі для врахування в імплементації:

Вимоги:

Визначення функції:

Реалізуйте рекурсивну функцію `isPalindrome`, яка перевіряє, чи заданий рядок є паліндромом.

Приклад визначення функції:

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

Перевантаження функцій:

Перевантажте функцію `isPalindrome` для роботи з цілими значеннями.

`bool isPalindrome(ціле число);`

Рекурсія:

Рекурсивна функція для рядків перевірить символи в поточній початковій і кінцевій позиціях. Якщо вони збігаються, він буде рекурсивно перевіряти наступні позиції, поки початок не перевищить кінець, після чого рядок буде визначено як паліндром.
Кроки реалізації

Завдання №6. Algotester. Self practice

- Планований час на реалізацію: 2 години
- Важливі деталі для врахування в імплементації:
 - Вихідні дані
 - У першому рядку одне ціле число - кількість елементів, які наявні в обох масивах одночасно.
 - У другому рядку кількість унікальних елементів в обох масивах (тобто кількість унікальних елементів у масиві, який буде об'єднанням двох даних).

4. Код програм з посиланням на зовнішні ресурси:

Завдання №1. VNS. Лабораторна робота №4. Варіант 7

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/300b0675814de3e1b2e5ea778a775a599460a569/ai_12/oleksandra_khvostova/epic_4/vns_lab_4_task_oleksandra_khvostova.cpp

```
1  #include <iostream>
2  #include <cstdlib>
3  #include <ctime>
4
5  using namespace std;
6
7  void generateRandomArray(int arr[], int size, int minValue, int maxValue) {
8      for (int i = 0; i < size; ++i) {
9          arr[i] = minValue + rand() % (maxValue - minValue + 1);
10     }
11 }
12
13 void printArray(const int arr[], int size) {
14     for (int i = 0; i < size; ++i) {
15         cout << arr[i] << " ";
16     }
17     cout << endl;
18 }
19
```



```

20 void removeLastZero(int arr[], int& size) {
21     for (int i = size - 1; i >= 0; --i) {
22         if (arr[i] == 0) {
23             for (int j = i; j < size - 1; j++) {
24                 arr[j] = arr[j + 1];
25             }
26             size--;
27             break;
28         }
29     }
30 }
31
32 void addElementAfterIndex(int arr[], int& size, int index, int value) {
33     if (index >= 0 && index < size) {
34         for (int i = size-1; i > index; i--) {
35             arr[i] = arr[i - 1];
36         }
37         arr[index + 1] = value;
38         size++;
39     }
40 }
41
42 int main() {
43     srand(time(0));
44
45     const int MAX_SIZE = 100;
46     int arr[MAX_SIZE];
47     int size = 10;
48
49     generateRandomArray(arr, size, -2, 2);
50     cout << "Generated array: ";
51     printArray(arr, size);
52     removeLastZero(arr, size);
53     printArray(arr, size);
54
55     int index = 3;
56     addElementAfterIndex(arr, size, index, 100);
57
58     cout << "Modified array: ";
59     printArray(arr, size);
60
61     return 0;
62 }

```

Завдання №2. VNS. Лабораторна робота №5. Варіант 7

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/300b0675814de3e1b2e5ea778a775a599460a569/ai_12/oleksandra_khvostova/epic_4/vns_lab_5_task_oleksandra_khvostova.cpp

```
1  #include <iostream>
2  using namespace std;
3
4  const int SIZE = 5;
5
6  void printMatrix(int matrix[SIZE][SIZE]) {
7      for (int i = 0; i < SIZE; ++i) {
8          for (int j = 0; j < SIZE; ++j) {
9              cout << matrix[i][j] << " ";
10         }
11         cout << endl;
12     }
13 }
14
15 void swapRowAndColumn(int matrix[SIZE][SIZE], int rowIndex, int colIndex) {
16     for (int i = 0; i < SIZE; ++i) {
17         swap(matrix[rowIndex][i], matrix[i][colIndex]);
18     }
19 }
20
21 int main() {
22     int matrix[SIZE][SIZE] = {
23         {1, 7, 3, 0, 8},
24         {5, 6, 7, 8, 9},
25         {3, 0, 1, 2, 3},
26         {4, 4, 5, 6, 7},
27         {2, 1, 0, 3, 4}
28     };
29
30     cout << "Original Matrix:" << endl;
31     printMatrix(matrix);
32
33     for (int i = 0; i < SIZE; ++i) {
34         for (int j = 0; j < SIZE; ++j) {
35             if (matrix[i][0] == matrix[0][j]) {
36                 swapRowAndColumn(matrix, i, j);
37             }
38         }
39     }
40
41     cout << "Modified Matrix:" << endl;
42     printMatrix(matrix);
43
44     return 0;
45 }
46
```

Завдання №3. Algotester. Лабораторна робота №2. Варіант 3

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/300b0675814de3e1b2e5ea778a775a599460a569/ai_12/oleksandra_khvostova/epic_4/algotester_lab_2_variant_3_oleksandra_khvostova.cpp

```
1  #include <iostream>
2  using namespace std;
3
4  int main(){
5      int N;
6      cin>>N;
7      int* array = new int[N];
8      for(int i=0; i<N; i++){
9          cin>>array[i];
10     }
11     int IndexL=0;
12     int IndexR=N-1;
13     int valL=0;
14     int valR=0;
15     for (IndexL, IndexR; valL=array[IndexL], valR=array[IndexR]; IndexL+=valL, IndexR-=valR){
16         if(IndexL==IndexR){
17             cout<<IndexL+1<<" "<<IndexR+1<<endl<<"Collision";
18             break;
19         }
20         if(IndexL+1==IndexR){
21             cout<<IndexL+1<<" "<<IndexR+1<<endl<<"Stopped";
22             break;
23         }
24         if(IndexL>IndexR){
25             cout<<IndexL+1<<" "<<IndexR+1<<endl<<"Miss";
26             break;
27         }
28     }
29     delete[]array;
30     return 0;
31 }
```

Завдання №4. Algotester. Лабораторна робота №3. Варіант 3

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/300b0675814de3e1b2e5ea778a775a599460a569/ai_12/oleksandra_khvostova/epic_4/algotester_lab_3_variant_3_oleksandra_khvostova.cpp

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  int main() {
6      string String, cString;
7      cin >> String;
8      for (int i = 0; i < String.size(); i++) {
9          int sum = 1;
10         while (i + 1 < String.size() && String[i] == String[i + 1]) {
11             sum++;
12             i++;
13         }
14         cString += String[i];
15         if (sum > 1) {
16             cString += to_string(sum);
17         }
18     }
19     cout << cString << endl;
20     return 0;
21 }
```

Завдання №5. Практична робота

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/300b0675814de3e1b2e5ea778a775a599460a569/ai_12/oleksandra_khvostova/epic_4/practice_work_task_1_oleksandra_khvostova.cpp

```
1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  // Рекурсивна функція для перевірки паліндрома для рядків
7  bool isPalindrome(const string& str, int start, int end) {
8      if (start >= end) {
9          return true; // Порожній рядок або один символ - паліндром
10     }
11     if (str[start] != str[end]) {
12         return false; // Якщо перший і останній символи різні, то це не паліндром
13     }
14     return isPalindrome(str, start + 1, end - 1); // Рекурсивний виклик для підрядка
15 }
16
17 // Перевантажена функція для перевірки паліндрома для цілих чисел
18 bool isPalindrome(int num) {
19     if (num < 0) {
20         return false; // Від'ємні числа не можуть бути паліндромами
21     }
22     if (num == 0) {
23         return true; // Нуль є паліндромом
24     }
25     int reversedNum = 0;
26     int originalNum = num;
27     while (num != 0) {
28         int remainder = num % 10;
29         reversedNum = reversedNum * 10 + remainder;
30         num /= 10;
31     }
32     return originalNum == reversedNum;
33 }
34
35 int main() {
36     string str;
37     int num;
38     cout<<"Enter the string: ";
39     cin>>str;
40     cout<<"Enter the integer: ";
41     cin>>num;
42
43     if (isPalindrome(str, 0, str.length() - 1)) {
44         cout << str << " is a palindrome" << endl;
45     } else {
46         cout << str << " is not a palindrome" << endl;
47     }
48
49     if (isPalindrome(num)) {
50         cout << num << " is a palindrome" << endl;
51     } else {
52         cout << num << " is not a palindrome" << endl;
53     }
54
55     return 0;
56 }
```

Завдання №6. Algotester. Self practice

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/300b0675814de3e1b2e5ea778a775a599460a569/ai_12/oleksandra_khvostova/epic_4/practice_work_self_algotester_tasks_oleksandra_khvo
stova.cpp

```
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int N, M;
6      cin >> N;
7      int* a = new int[N];
8      for (int i = 0; i < N; ++i) {
9          cin >> a[i];
10     }
11
12     cin >> M;
13     int* b = new int[M];
14     for (int i = 0; i < M; ++i) {
15         cin >> b[i];
16     }
17
18     // Знаходження елементів, які наявні в обох масивах
19     int count = 0;
20     for (int i = 0; i < N; ++i) {
21         for (int j = 0; j < M; ++j) {
22             if (a[i] == b[j]) {
23                 ++count;
24                 break;
25             }
26         }
27     }
28
29     // Створюємо об'єднання двох масивів з унікальними елементами
30     int* union_array = new int[N + M];
31     int union_size = 0;
32
33     // Додаємо всі елементи з першого масиву
34     for (int i = 0; i < N; ++i) {
35         union_array[union_size++] = a[i];
36     }
37
38     // Додаємо тільки ті елементи з другого масиву, яких ще немає в об'єднаному масиві
39     for (int i = 0; i < M; ++i) {
40         bool found = false;
41         for (int j = 0; j < N; ++j) {
42             if (b[i] == a[j]) {
43                 found = true;
44                 break;
45             }
46         }
47         if (!found) {
48             union_array[union_size++] = b[i];
49         }
50     }
51
52     cout << count << endl;
53     cout << union_size << endl;
54
55     delete[] a;
56     delete[] b;
57     delete[] union_array;
58
59     return 0;
60 }
```

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

Завдання №1. VNS. Лабораторна робота №4. Варіант 7

```
Generated array: 2 2 0 1 1 2 -1 2 -2 -1
2 2 1 1 2 -1 2 -2 -1
Modified array: 2 2 1 1 100 2 -1 2 -2 -1
```

Час затрачений на виконання завдання: 40 хвилин

Завдання №2. VNS. Лабораторна робота №5. Варіант 7

- Деталі завдання
- Важливі деталі для врахування в імплементації програми

```
Original Matrix:
1 7 3 0 8
5 6 7 8 9
3 0 1 2 3
4 4 5 6 7
2 1 0 3 4
Modified Matrix:
1 5 3 4 2
7 6 0 8 9
3 7 1 5 0
0 4 2 6 7
8 1 3 3 4
```

Час затрачений на виконання завдання: 1.5 години

Завдання №3. Algotester. Лабораторна робота №2. Варіант 3

- Деталі завдання
- Важливі деталі для врахування в імплементації програми

```
10
1 3 1 1 5 1 1 2 1 2
5 6
Stopped
PS C:\Users\User> ^C
PS C:\Users\User>
PS C:\Users\User> & 'c:\Users\
-MIEngine-In-vszoekge.ycn' '--s
patwygy.lxc' '--dbgExe=C:\msys6
10
1 3 1 1 5 1 1 3 1 2
5 5
Collision
PS C:\Users\User> ^C
PS C:\Users\User>
PS C:\Users\User> & 'c:\Users\
-MIEngine-In-rvgxbh5q.1ro' '--s
1tv5aab.ba4' '--dbgExe=C:\msys6
10
1 3 1 1 5 1 1 5 1 2
5 3
Miss
```

Час затрачений на виконання завдання: 2 години

Завдання №4. Algotester. Лабораторна робота №3. Варіант 3

- Деталі завдання

- Важливі деталі для врахування в імплементації програми

```
AaaAaaaAAaaaaaaAaaAAAAA
Aa2Aa3A2a8Aa2A6
```

Час затрачений на виконання завдання: 1,5 години

Завдання №5. Практична робота

- Деталі завдання
- Важливі деталі для врахування в імплементації програми

```
Enter the string: radar
Enter the integer: 1234321
radar is a palindrome
1234321 is a palindrome
```

Час затрачений на виконання завдання: 1,5 години

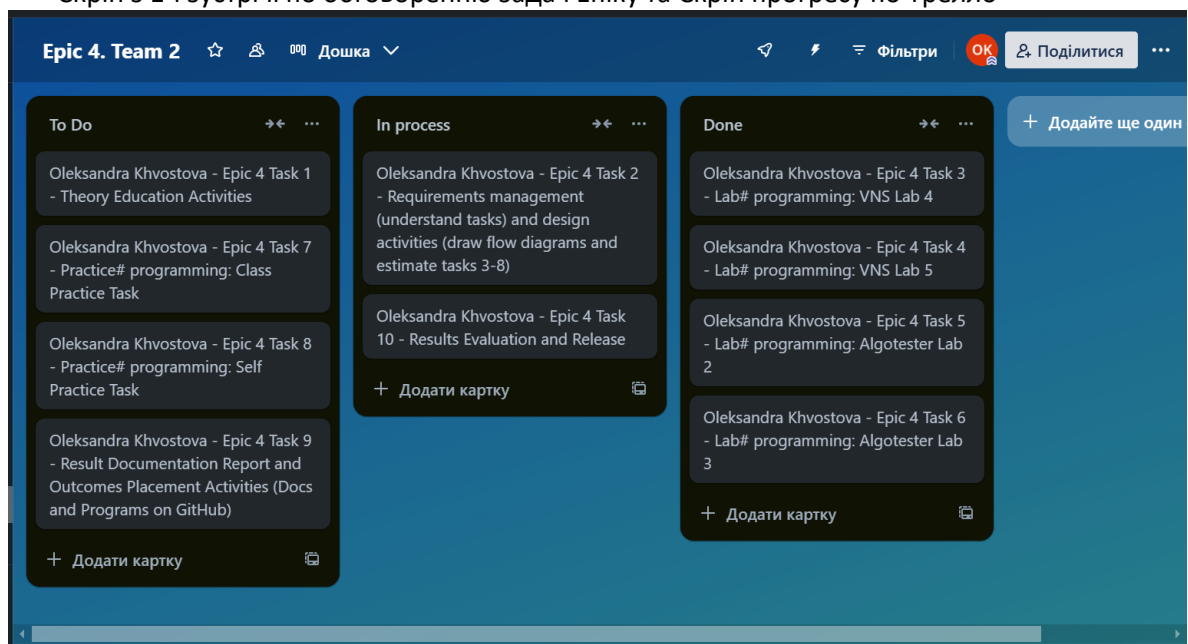
Завдання №6. Algotester. Self practice

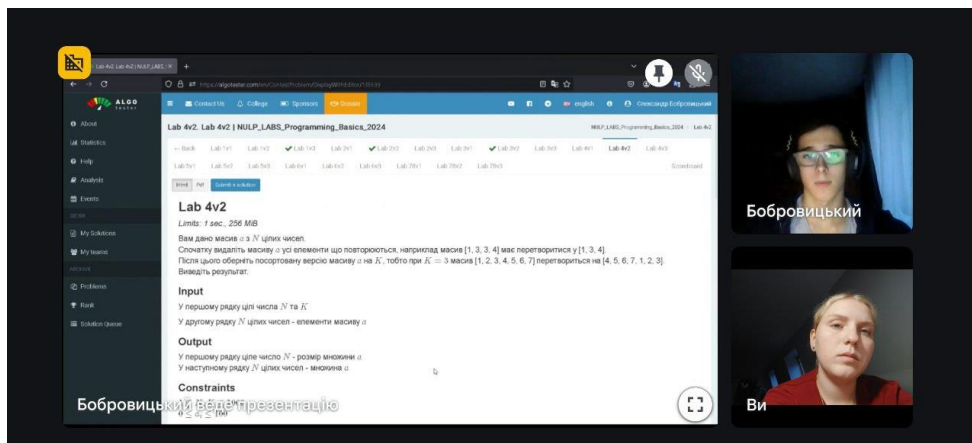
```
5
1 2 3 1 1
3
6 7 1
3
7
```

Час затрачений на виконання завдання: 1,5 години

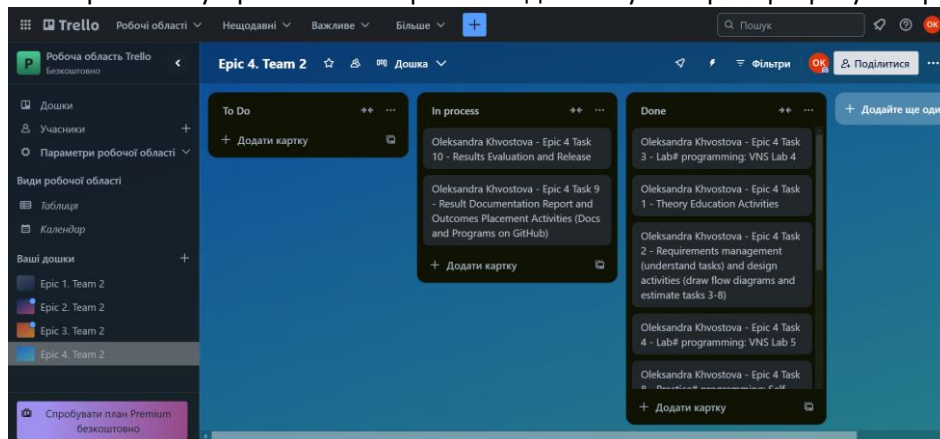
6. Кооперація з командою:

- Скрін з 1-ї зустрічі по обговоренню задач Епіку та Скрін прогресу по Тrello

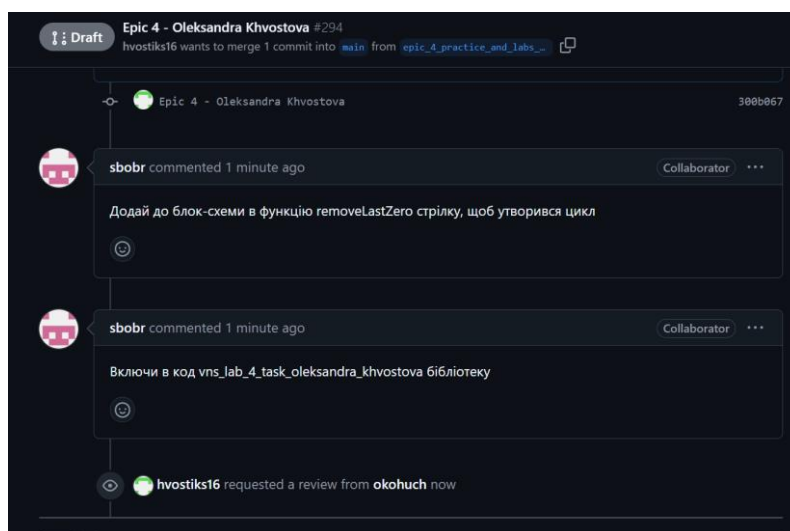




- Скрін з 2-ї зустрічі по обговоренню задач Епіку та Скрін прогресу по Тrello



- Скрін з 2-му коментарями від учасників команди на пул реквестів з Ревью Роботи



Висновки:

В результаті виконання роботи було детально опрацьовано теоретичні та практичні аспекти роботи з масивами та структурами даних у мові C++. Студенти ознайомилися з класами пам'яті, динамічним виділенням пам'яті, основами масивів, вказівниками, посиланнями, а також алгоритмами обробки даних. Було вивчено, як використовувати одновимірні та двовимірні масиви, динамічні масиви і структури даних для розв'язання складних задач. Вкладені структури та їх застосування додали глибину розумінню структури даних.