

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 ІІІ-11

Потапова Світлана Сергіївна

Львів 2024

**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:**

- Навчитись створювати та використовувати одновимірні, двовимірні, динамічні масиви, ознайомитися з структурами даними, з алгоритмами обробки та роботи з масивами та структурами, навчитися користуватися вказівниками та посиланнями.

### **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

- Тема №1. Класи пам'яті у C++
- Тема №2. Вступ до масивів та вказівників
- Тема №3. Одновимірні масиви
- Тема №4. Вказівники та посилання
- Тема №5. Двовимірні масиви
- Тема №6. Динамічні масиви
- Тема №7. Структури даних
- Тема №8. Вкладені структури
- Тема №9. Використання структур
- Тема №10. Алгоритми обробки та робота з масивами та структурами

2. Індивідуальний план опрацювання теорії:

Тема №1. Класи пам'яті у C++

- Джерела інформації:
  - <https://www.geeksforgeeks.org/storage-classes-in-c-with-examples/>
  - <http://cpp.dp.ua/klasy-pam-yati-u-c-builder/>
- Що опрацьовано:
  - Класи пам'яті, стек, виділення та вивільнення пам'яті
- Статус: ознайомлена

Тема №2. Вступ до масивів та вказівників

- Джерела інформації:
  - <https://acode.com.ua/urok-77-masyvy/>
- Що опрацьовано:
  - Масиви, вказівники, взаємозв'язок між ними
- Статус: ознайомлена

### Тема №3. Одновимірні масиви

- Джерела інформації:
  - <https://acode.com.ua/urok-77-masyvy/>
- Що опрацьовано:
  - Одновимірні масиви та основні операції над ними
- Статус: ознайомлена

### Тема №4. Вказівники та посилання

- Джерела інформації:
  - <https://acode.com.ua/urok-84-vkazivnyky/>
- Що опрацьовано:
  - Вказівники та посилання, різниця між ними, арифметика вказівників,
- Статус: ознайомлена

### Тема №5. Двовимірні масиви

- Джерела інформації:
  - <https://acode.com.ua/urok-81-bagatovymirni-masyvy/>
- Що опрацьовано:
  - Двовимірні масиви, вкладені цикли для обходу, передача двовимірних масивів у функцію
- Статус: ознайомлена

### Тема №6. Динамічні масиви

- Джерела інформації:
  - <https://acode.com.ua/urok-89-dynamichne-vydilennya-pam-yati/>
  - <https://acode.com.ua/urok-90-dynamichni-masyvy/>
- Що опрацьовано:
  - Динамічне виділення пам'яті, створення динамічних масивів
- Статус: ознайомлена

### Тема №7. Структури даних

- Джерела інформації:
  - [https://www.w3schools.com/cpp/cpp\\_structs.asp](https://www.w3schools.com/cpp/cpp_structs.asp)
  - [https://www.w3schools.com/cpp/cpp\\_enum.asp](https://www.w3schools.com/cpp/cpp_enum.asp)
- Що опрацьовано:
  - Оголошення та використання структур
- Статус: ознайомлена

### Тема №8. Вкладені структури

- Джерела інформації:
  - <https://www.geeksforgeeks.org/nested-structure-in-c-with-examples/>

- Що опрацьовано:
  - Вкладені структури та взаємодія з ними
- Статус: ознайомлена

#### Тема №9. Використання структур

- Джерела інформації:
  - <https://www.geeksforgeeks.org/operator-overloading-cpp/>
- Що опрацьовано:
  - Оператори у структурах
- Статус: ознайомлена

#### Тема №10. Алгоритми обробки та робота з масивами та структурами

- Джерела інформації:
  - <https://www.programiz.com/dsa/sorting-algorithm>
  - <https://www.geeksforgeeks.org/searching-algorithms/>
- Що опрацьовано:
  - Алгоритми пошуку та сортування в масивах
- Статус: ознайомлена

### Виконання роботи:

1. Опрацювання завдань та вимог до середовища:

#### **Завдання №1. VNS lab 4 variant 15.**

- 1) Реалізувати з використанням масиву двонаправлене кільце (перегляд можливий в обидва боки, від останнього елемента можна перейти до першого).
- 2) Роздрукувати отриманий масив, починаючи з К-ого елемента і до К-1 (по кільцю вліво).
- 3) Знищити з кільця перший й останній елементи.
- 4) Роздрукувати отриманий масив, починаючи з К-ого елемента (і до К+1 по кільцю вправо).

#### **Завдання №2. VNS lab 5 variant 15.**

Задано двовимірний масив. Знайти суму елементів першого стовпця без одного останнього елемента, суму елементів другого стовпця без двох останніх, суму елементів третього стовпця без трьох останніх і т.д. Останній стовпець не обробляється. Серед знайдених сум знайти максимальну.

#### **Завдання №3. Algotester lab 2 variant 1**

У вас є дорога, яка виглядає як  $N$  чисел. Після того як ви по ній пройдете - вашу втому можна визначити як різницю максимального та мінімального елементу.

Ви хочете мінімізувати втому, але все що ви можете зробити - викинути одне число з дороги, тобто забрати його з масиву.

В результаті цієї дії, яку мінімальну втому ви можете отримати в кінці дороги?

#### **Вхідні дані**

У першому рядку ціле число  $N$  - кількість чисел

У другому рядку масив  $r$ , який складається з  $N$  цілих чисел

#### **Вихідні дані**

Єдине ціле число  $m$  - мінімальна втома, яку можна отримати

### **Завдання №4. Algotester lab 3 variant 2**

Вам дано 2 масиви розміром  $N$  та  $M$ . Значення у цих масивах унікальні. Ваше завдання вивести у першому рядку кількість елементів, які наявні в обох масивах одночасно, у другому кількість унікальних елементів в обох масивах разом.

#### **Вхідні дані**

У першому рядку ціле число  $N$ , у другому рядку  $N$  цілих чисел  $a_1..a_n$

У третьому рядку ціле число  $M$ , у четвертому рядку  $M$  цілих чисел  $b_1..b_n$

#### **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).

### **Завдання №5. Practice task**

Реалізувати програму, яка перевіряє, чи дане слово чи число паліндромом за допомогою рекурсії. Паліндром — це послідовність символів, яка однаково читається вперед і назад (наприклад, «radar», «level», «12321»).

Вимоги:

1. Визначення функції:

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

2. Приклад визначення функції:

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

3. Перевантаження функцій:

Перевантажте функцію *isPalindrome* для роботи з цілими значеннями.

*bool isPalindrome(ціле число);*

#### 4. 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

- 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.

#### **Завдання №6. Self-practice alogtester lab 3 variant 1**

Ви з'явилися у світі під назвою Атод посеред Пустелі Безправ'я. Так сталося, що Ви попали саме в той час і місце, де ведеться битва між чаклункою Ліною і темними силами, які хочуть знищити цей світ. На жаль, трапилась халепа, бо деякі слова із книги чар були пошкоджені під час битви. Одне таке слово можна відновити виконавши ритуал зцілення над пошкодженими буквами. Ритуал зцілення можна виконати на всіх **підряд** розташованих **пошкоджених** буквах. Вам не залишається нічого іншого як допомогти Ліні відновити ці слова і сказати скільки мінімально треба провести таких ритуалів, щоб прочитати одне з наймогутніших у цьому світі заклять - Поневолення Дракона!

##### **Вхідні дані**

У першому рядку N - кількість рядків у заклятті.

В наступних N рядках - набір слів w1,...,wM, розділених пробілами, де кожне слово може містити малі латинські літери та символ #, який позначає пошкоджену букву.

##### **Вихідні дані**

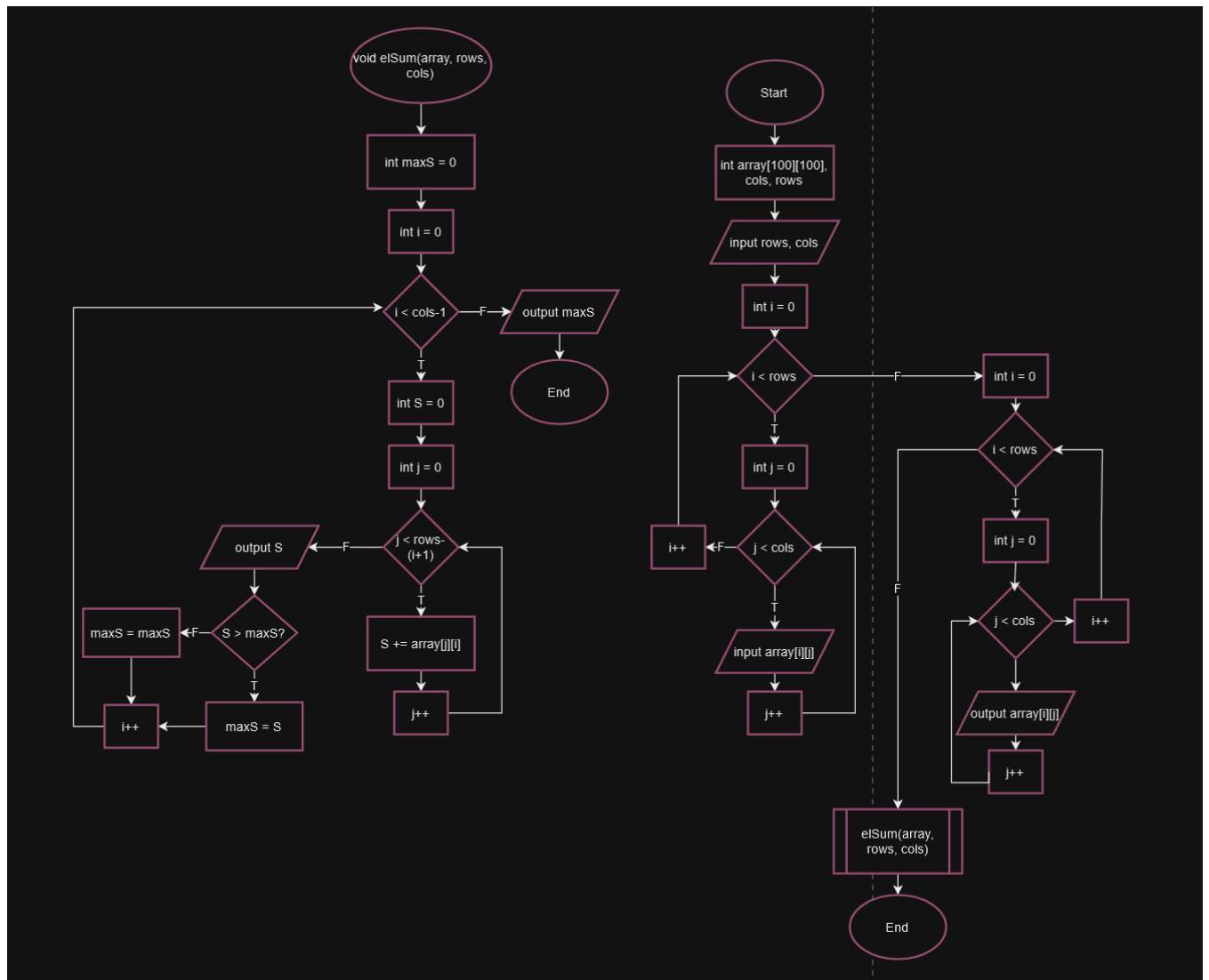
Єдине ціле число - мінімальна кількість ритуалів, які потрібно провести, щоб відновити закляття.

#### **Завдання №7. Self-practice Structures** <https://www.w3resource.com/c-programming-exercises/structure/index.php>

Create a structure named "Employee" to store employee details such as employee ID, name, and salary. Write a program to input data for employees, find the highest salary employee, and display their information.

#### **2. Дизайн виконання завдань**

##### **Завдання №2. VNS lab 5 variant 15**



### 3. Код програм з посиланням на зовнішні ресурси

#### Завдання №1. VNS lab 4 variant 15.

[https://github.com/artificial-intelligence-department/ai\\_programming\\_playground\\_2024/pull/300/files#diff-b85ee76f773532471c743db81621b9efb29cc35318098de52c07fe1b239fb9c](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/300/files#diff-b85ee76f773532471c743db81621b9efb29cc35318098de52c07fe1b239fb9c)

```

1  #include <iostream>
2  using namespace std;
3
4  void print_left(int* array, int n, int k){
5      for(int i=0; i<n; i++){
6          int index = k-i;
7          if (index<0){
8              index +=n;
9          }
10         cout << array[index] << " ";
11     }
12     cout << endl;
13 }
14
15 void print_right(int* array, int n, int k){
16     for (int i=0; i<n; i++){
17         int index = k+i;
18         if(index>n-1){
19             index -= n;
20         }
21         cout << array[index] << " ";
22     }
23     cout << endl;
24 }
25
26 void deleteEl(int* array, int& n){
27     for(int i=0; i<n-1; i++){
28         array[i] = array[i+1];
29     }
30     n -= 2;
31 }
32
33 int main(){
34     int array[100];
35     int n, k;
36     cout << "Введіть розмір масиву:" << endl;
37     cin >> n;
38     cout << "Введіть елементи масиву:" << endl;
39     for (int i=0; i<n; i++){
40         cin >> array[i];
41     }
42     cout << "Введіть k номер елемента:" << endl;
43     cin >> k;
44     cout << "Масив по кільцю вліво від елемента k:" << endl;
45     print_left(array, n, k);
46     cout << "Масив(з видаленням першим та останнім елементом) по кільцю вправо від елемента k:" << endl;
47     deleteEl(array,n);
48     print_right(array, n, k);
49
50     return 0;
51 }

```

## Завдання №2. VNS lab 5 variant 15.

[https://github.com/artificial-intelligence-department/ai\\_programming\\_playground\\_2024/pull/300/files#diff-7a3830d02e6c3d2a573bcece8719895fff640f962767689fbf001b5f53f8b655](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/300/files#diff-7a3830d02e6c3d2a573bcece8719895fff640f962767689fbf001b5f53f8b655)



```

1  #include <iostream>
2  using namespace std;
3
4  void elSum(int array[100][100], int rows, int cols){
5      int maxS = 0;
6      for(int i=0; i<cols-1; i++){
7          int S = 0;
8          for(int j=0; j<rows-(i+1); j++){
9              S += array[j][i];
10         }
11         cout << S << " ";
12         maxS = (S > maxS) ? S : maxS;
13     }
14     cout << endl << "Максимальна сума: " << maxS;
15 }
16 int main(){
17     int array[100][100];
18     int rows, cols;
19     cin >> rows >> cols;
20     for (int i=0; i<rows; i++){
21         for (int j=0; j<cols; j++){
22             cin >> array[i][j];
23         }
24     }
25     cout << "Матриця " << rows << "x" << cols << ":\n";
26     for (int i = 0; i < rows; i++) {
27         for (int j = 0; j < cols; j++) {
28             cout << array[i][j] << " ";
29         }
30         cout << endl;
31     }
32     elSum(array, rows, cols);
33     return 0;
34 }

```

### Завдання №3. Algotester lab 2 variant 1

[https://github.com/artificial-intelligence-department/ai\\_programming\\_playground\\_2024/pull/300/files#diff-c0a8ff45dbf5db83e26eba7b9d557099dc8dc02177b988e1ea24350a3190adb](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/300/files#diff-c0a8ff45dbf5db83e26eba7b9d557099dc8dc02177b988e1ea24350a3190adb)

c

```

1  #include <iostream>
2  using namespace std;
3
4  void bubbleSort(int* array, int N){
5      for (int step = 0; step < (N-1); ++step) {
6          bool swapped = false;
7          for (int i = 0; i < (N-step-1); ++i) {
8              if (array[i] > array[i + 1]) {
9                  int temp = array[i];
10                 array[i] = array[i + 1];
11                 array[i + 1] = temp;
12                 swapped = true;
13             }
14         }
15         if (!swapped)
16             break;
17     }
18 }

```

```

19
20  int main(){
21      int N;
22      cin >> N;
23      int *array = new int[N];
24      for(int i = 0; i<N; i++){
25          cin >> array[i];
26      }
27
28      if(N==1){
29          cout << 0;
30          return 0;
31      }
32
33      bubbleSort(array, N);
34
35      int tiredness = array[N-1] - array[0];
36      tiredness = min(tiredness, array[N-2] - array[0]);
37      tiredness = min(tiredness, array[N-1]-array[1]);
38
39      cout << tiredness;
40      return 0;
41  }

```

#### Завдання №4. Algotester lab 3 variant 2

[https://github.com/artificial-intelligence-department/ai\\_programming\\_playground\\_2024/pull/300/files#diff-6080a96955fe005ecff4b8eaecl72a73484de5db0f377c8876cf3d1d6980b92f](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/300/files#diff-6080a96955fe005ecff4b8eaecl72a73484de5db0f377c8876cf3d1d6980b92f)

```

1  #include <iostream>
2  using namespace std;
3
4  int main(){
5
6      int N, M;
7      cin >> N;
8      int *array1 = new int[N];
9      for(int i=0; i<N; i++){
10         cin >> array1[i];
11     }
12     cin >> M;
13     int *array2 = new int[M];
14     for(int i=0; i<M; i++){
15         cin >> array2[i];
16     }
17
18     int both_arrays = 0, unique = 0;

```

```

19
20     for(int i=0; i<N; i++){
21         bool isUnique = true;
22         for(int j=0; j<M; j++){
23             if (array1[i]==array2[j]){
24                 isUnique = false;
25                 both_arrays++;
26                 break;
27             }
28         }
29     }
30     unique = N + M - both_arrays;
31     cout << both_arrays << endl << unique;
32
33     delete[] array1;
34     delete[] array2;
35
36     return 0;
37 }

```

### Завдання №5. Practice task

[https://github.com/artificial-intelligence-department/ai\\_programming\\_playground\\_2024/pull/300/files#diff-f20a976533b2897d9790883ee962893dd80a0932743a449de67fc25d49c00b9e](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/300/files#diff-f20a976533b2897d9790883ee962893dd80a0932743a449de67fc25d49c00b9e)

```

1  #include <iostream>
2  using namespace std;
3
4  bool isPalindrome(const string& str, int start, int end){
5      if (start >= end) return true;
6
7      if (str[start] != str[end]) return false;
8
9      return isPalindrome(str, ++start, --end);
10 }
11
12 bool isPalindrome(int num){
13     int reversed = 0;
14
15     int temp = num;
16     while (temp != 0) {
17         reversed = reversed * 10 + temp % 10;
18         temp /= 10;
19     }
20     return (reversed == num);
21 }
22
23 int main(){
24     int num;
25     string str;
26
27     cout << "Enter a string: " << endl;
28     getline(cin, str);
29

```

```

30     if(isPalindrome(str, 0, str.length() - 1)){
31         cout << str << " is a palindrome" << endl;
32     }
33     else{
34         cout << str << " isn't a palindrome" << endl;
35     }
36
37     cout << "Enter a number: " << endl;
38     cin >> num;
39
40     if(isPalindrome(num)){
41         cout << num << " is a palindrome" << endl;
42     }
43     else{
44         cout << num << " isn't a palindrome" << endl;
45     }
46
47     return 0;
48 }

```

## Завдання №6. Self-practice alogtester lab 3 variant 1

[https://github.com/artificial-intelligence-](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/300/files#diff-d8bf9d2ed5007f90adcbd4df0c1f0a2336a15034b39ec78355dfe408de4eb11e)

[department/ai\\_programming\\_playground\\_2024/pull/300/files#diff-](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/300/files#diff-d8bf9d2ed5007f90adcbd4df0c1f0a2336a15034b39ec78355dfe408de4eb11e)

[d8bf9d2ed5007f90adcbd4df0c1f0a2336a15034b39ec78355dfe408de4eb11e](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/300/files#diff-d8bf9d2ed5007f90adcbd4df0c1f0a2336a15034b39ec78355dfe408de4eb11e)

```

1  #include <iostream>
2  using namespace std;
3
4  int countChar(const string& str, char ch){
5      int count = 0;
6      bool counted = false;
7      for (char c : str){
8          if(c==ch){
9              if (!counted){
10                 count++;
11                 counted = true;
12             }
13         }
14         else counted = false;
15     }
16
17     return count;
18 }
19
20 int main(){
21     int N, result = 0;
22     string str;
23     char target = '#';
24     cin >> N;
25     cin.ignore();
26
27     string *array = new string[N];
28     for(int i=0; i<N; i++){
29         getline(cin, str);
30         array[i] = str;
31     }

```

```

32
33     for(int i=0; i<N; i++){
34         result += countChar(array[i], target);
35     }
36
37     cout << result;
38     delete[] array;
39     return 0;
40 }

```

## Завдання №7. Self-practice Structures

[https://github.com/artificial-intelligence-department/ai\\_programming\\_playground\\_2024/pull/300/files#diff-59d45e6cdf6bd1b4ea8153d8124aca21cf7ef33985e949f06dd150fda4b42180](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/300/files#diff-59d45e6cdf6bd1b4ea8153d8124aca21cf7ef33985e949f06dd150fda4b42180)

```

1  #include <iostream>
2  using namespace std;
3
4  struct employee{
5      int id;
6      string name;
7      int salary;
8
9      friend ostream& operator<<(ostream& os, const employee emp){
10         os << "Employee ID: " << emp.id << ", name: " << emp.name << ", salary: " << emp.salary;
11         return os;
12     }
13     friend istream& operator>>(istream& is, employee& emp){
14         cout << "Enter ID: ";
15         is >> emp.id;
16         cout << "Enter name: ";
17         is >> emp.name;
18         cout << "Enter salary: ";
19         is >> emp.salary;
20         return is;
21     }
22 };
23
24 int main(){
25     int N;
26     cout << "Enter amount of employees: " << endl;
27     cin >> N;
28
29     employee* array = new employee[N];
30
31     for(int i=0; i<N; i++){
32         cout << "Enter data for employee №" << i+1 << endl;
33         cin >> array[i];

```

```

34     }
35
36     int maxSalary = 0;
37     for(int i=0; i<N; i++){
38         if(array[i].salary > maxSalary){
39             maxSalary = array[i].salary;
40         }
41         cout << array[i] << endl;
42     }
43     cout << "The highest salary is: " << maxSalary;
44     delete[] array;
45     return 0;
46 }

```

#### 4. Результати виконання завдань та фактично затрачений час

##### Завдання №1. VNS lab 4 variant 15.

```
Введіть розмір масиву:
8
Введіть елементи масиву:
1 2 3 4 5 6 7 8
Введіть k номер елемента:
4
Масив по кільцю вліво від елемента k:
5 4 3 2 1 8 7 6
Масив(з видаленим першим та останнім елементом) по кільцю вправо від елемента k:
6 7 2 3 4 5
```

Планований час: 40 хв, фактично: 40 хв

##### Завдання №2. VNS lab 5 variant 15.

```
4 4
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Матриця 4x4:
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
15 8 3
Максимальна сума: 15
```

Планований час: 30 хв, фактично: 40 хв

##### Завдання №3. Algotester lab 2 variant 1

```
5
1 3 3 4 4
1
```

2 дні тому	Lab 2v1 - Lab 2v1	C++ 23	Зараховано	0.006	1.063	1862503
------------	-------------------	--------	------------	-------	-------	---------

Планований час: 40 хв, фактично: 50 хв

##### Завдання №4. Algotester lab 3 variant 2

```
5
1 2 3 4 5
5
4 5 6 7 8
2
8
```

4 дні тому	Lab 3v2 - Lab 3v2	C++ 23	Зараховано	0.003	1.266	18618
------------	-------------------	--------	------------	-------	-------	-------

Планований час: 40 хв, фактично: 30 хв

### Завдання №5. Practice task

```
Enter a string:
hello
hello isn't a palindrome
Enter a number:
12721
12721 is a palindrome
```

Планований час: 40 хв, фактично: 40 хв

### Завдання №6. Self-practice alogtester lab 3 variant 1

```
8
d#rkness b#yond twilight
crimson beyond blood that flows
buried in the strea# of t### is where your power grow
i pledge myself to conquer all the foes who stand
before the mighty gift bestowed in my unworthy hand
let the fools who stand before me be destroyed
by the power you and i possess
dr#gon sl#ve
6
```

2 дні тому	Lab 3v1 - Lab 3v1	C++ 23	Зараховано	0.017	1.691	18625
------------	-------------------	--------	------------	-------	-------	-------

Планований час: 30 хв, фактично: 30 хв

### Завдання №7. Self-practice Structures

```
Enter amount of employees:
2
Enter data for employee №1
Enter ID: 101
Enter name: James
Enter salary: 2500
Enter data for employee №2
Enter ID: 102
Enter name: Mary
Enter salary: 5000
Employee ID: 101, name: James, salary: 2500
Employee ID: 102, name: Mary, salary: 5000
The highest salary is: 5000
```

Планований час: 20 хв, фактично: 20 хв

## 5. Кооперація з командою

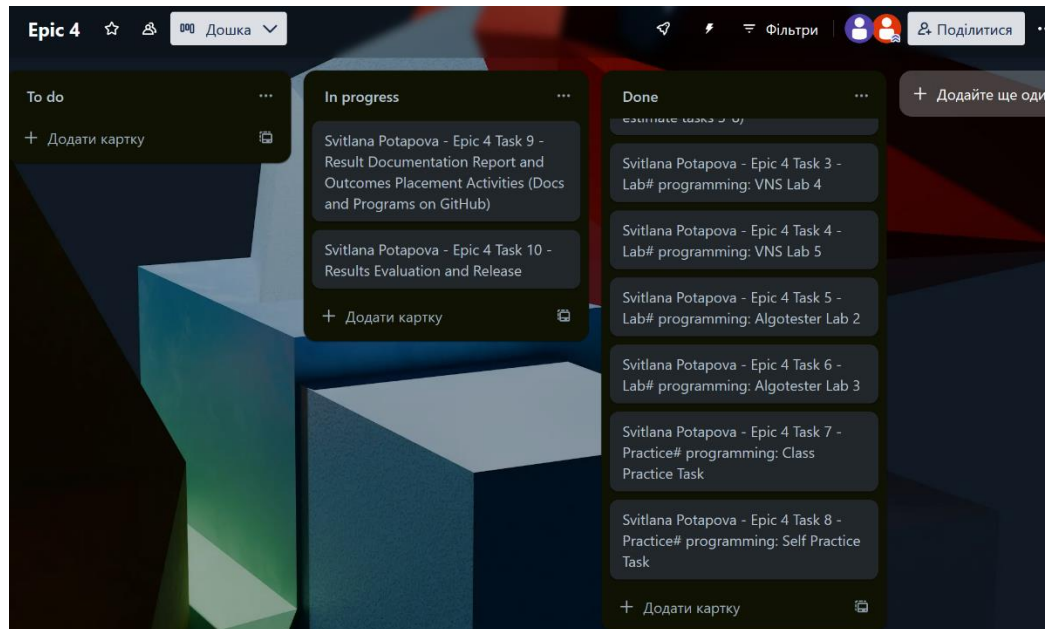


Рисунок 1. Trello



**Висновок:** Під час виконання роботи я на практиці закріпила знання про одновимірні, двовимірні, динамічні масиви, їх застосування, про використання вказівників та посилань, про структури даних, вкладені структури, їх використання.