

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:

Практичних Робіт до блоку № 4

**Виконала:**

Студентка групи ІІІ-12  
Іванів Христина Вікторівна

**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:** ознайомитись з одновимірними та двовимірними масивами, вказівниками та посиланнями, структурами даних, вкладеними структурами, алгоритмами обробки та роботою з масивами та структурами в мовах C та C++.

### Теоретичні відомості:

Теоретичні відомості з переліком важливих тем:

- 1) Тема №1: Класи пам'яті у C++
- 2) Тема №2: Вступ до Масивів і Вказівників
- 3) Тема №3: Одновимірні Масиви
- 4) Тема №4: Вказівники та Посилання
- 5) Тема №5: Двовимірні Масиви
- 6) Тема №6: Динамічні Масиви
- 7) Тема №7: Структури Даних
- 8) Тема №8: Вкладені Структури
- 9) Тема №9: Використання структур
- 10) Тема №10: Алгоритми обробки та робота з масивами та структурами

#### 1) Індивідуальний план опрацювання теорії:

- Тема №1: Класи пам'яті у C++
  - Джерела Інформації:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Урок 58 з курсу C++ теорія з каналу «Блоган»
    - Урок №89. Динамічне виділення пам'яті з сайту acode
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Урок 58 з курсу C++ теорія з каналу «Блоган»
    - Урок №89. Динамічне виділення пам'яті з сайту acode
  - Статус: ознайомлена з класами пам'яті, відмінністю між ними,
- Тема №2: Вступ до Масивів і Вказівників
  - Джерела Інформації:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 57 з курсу C++ теорія з каналу «Блоган»
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 57 з курсу C++ теорія з каналу «Блоган»

Статус: напівознайомена з вказівниками та роботою з ними та  
ознайомена з масивами

- Тема №3: Одновимірні Масиви
  - Джерела Інформації:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 40, 50, 51 з курсу C++ теорія з каналу «Блоган»
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 40, 50, 51 з курсу C++ теорія з каналу «Блоган»
  - Статус: ознайомена з одновимірними масивами
- Тема №4: Вказівники та Посилання
  - Джерела Інформації:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 62 з курсу C++ теорія з каналу «Блоган»
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 62 з курсу C++ теорія з каналу «Блоган»
  - Статус: ознайомена з вказівниками та посиланнями у C++
- Тема №5: Двовимірні Масиви
  - Джерела Інформації:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 41 з курсу C++ теорія з каналу «Блоган»
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 41 з курсу C++ теорія з каналу «Блоган»
  - Статус: навчилась використовувати двовимірні масиви та виконувати деякі дії з ними
- Тема №6: Динамічні Масиви
  - Джерела Інформації:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 60 з курсу C++ теорія з каналу «Блоган»
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Уроки 60 з курсу C++ теорія з каналу «Блоган»
  - Статус: ознайомена навчилась використовувати динамічні масиви та виконувати деякі дії з ними
- Тема №7: Структури Даних
  - Джерела Інформації:
    - Лекції О. Пшеничного

- Практичні М. Фаріон
    - Урок №64. Структури з сайту acode
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Урок №64. Структури з сайту acode
  - Статус: ознайомлена з структурами даних та роботою з ними
- Тема №8: Вкладені Структури
  - Джерела Інформації:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Урок №64. Структури з сайту acode
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Урок №64. Структури з сайту acode
  - Статус: ознайомлена з вкладеними структурами, їх застосуванням
- Тема №9: Використання структур
  - Джерела Інформації:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Урок №138. Перевантаження операторів з сайту acode
    - Урок №141. Перевантаження операторів вводу і виводу з сайту acode
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
    - Урок №138. Перевантаження операторів з сайту acode
    - Урок №141. Перевантаження операторів вводу і виводу з сайту acode
  - Статус: ознайомлена з використанням структур
- Тема №10: Алгоритми обробки та робота з масивами та структурами
  - Джерела Інформації:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
  - Що опрацьовано:
    - Лекції О. Пшеничного
    - Практичні М. Фаріон
  - Статус: ознайомлена з вбудованими функціями, вмію використовувати їх на практиці

## **Виконання роботи:**

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

## Завдання №1 VNS. Лабораторна робота №4

- **Варіант завдання: 6**
- *Деталі завдання:*
  - 1) Сформувати одновимірний масив цілих чисел, використовуючи генератор випадкових чисел.
  - 2) Роздрукувати отриманий масив.
  - 3) Знищити елемент із заданим номером.
  - 4) Додати після першого парного елемента масиву елемент зі значенням  $M[I-1]+2$ .
  - 5) Роздрукувати отриманий масив.
- *Важливі деталі для врахування:*
  - 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 завдання: 6**
- *Деталі завдання:* Використовуючи функції, розв'язати зазначене у варіанті завдання. Масив повинен передаватися у функцію як параметр. Елемент матриці є сідловою точкою, якщо він є найменшим у своєму рядку й найбільшим у своєму стовпці (або навпаки: найбільшим у своєму рядку й найменшим у своєму стовпці). Для заданої матриці визначити всі сідлові точки.

## Завдання №3 Algotester task 2. V- 3

- **Варіант завдання: 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  $a_1a_1$ , 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 опиняються 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  $a_ia_i$  та  $a_{i+1}a_{i+1}$  - виведіть **Stopped**

Врахуйте, що перевіряти треба також до взльоту.

- *Важливі деталі для врахування:*

$$1 \leq N \leq 1000 \quad 1 \leq N \leq 1000$$

$$1 \leq a_i \leq 5 \quad 1 \leq a_i \leq 5$$

Гарантується, що дрони не можуть вилетіти за межі  $[1, N][1, N]$  (тобто не існує масиву, в якому дрони вилітають за межі до того, як станеться одна з 3х вищезазначених ситуацій) та їх швидкість завжди більша за нуль.

#### Завдання №4 VNS. Algotester task 3. V- 2

- *Варіант завдання: 2*

- *Деталі завдання:* 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.
- *Важливі деталі для врахування:*

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).

Обмеження

$$0 \leq N \leq 100$$

$$0 \leq a_i \leq 100$$

$$0 \leq M \leq 100$$

$$0 \leq b_i \leq 100$$

### Завдання №5 Class Practice Work

- *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»).

- *Важливі деталі для врахування:*
  1. Визначте та реалізуйте рекурсивну функцію isPalindrome для рядків.
  2. Визначте та реалізуйте перевантажену функцію isPalindrome для цілих чисел. Використати математичний підхід щоб перевірити чи число є паліндромом.

### Завдання №6 Self Practice Algotester Task

- *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:*

Вхідні дані:

У першому рядку стрічка S

Вихідні дані:

Стрічка Scompressed

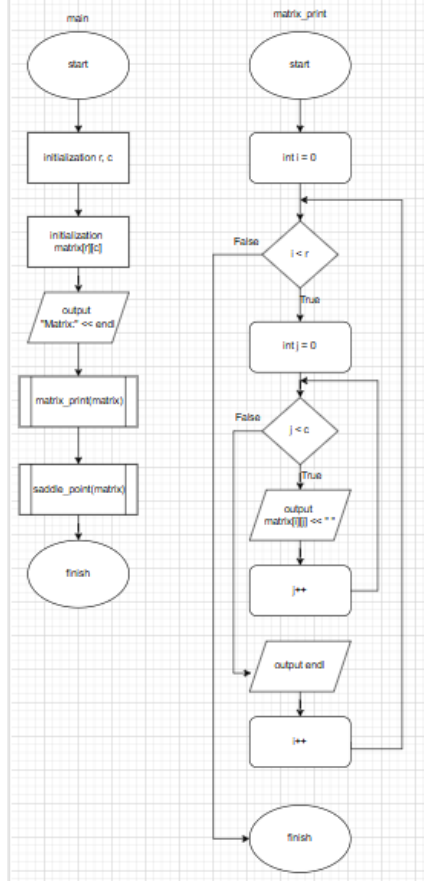
Обмеження:

$$1 \leq |S| \leq 105$$

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

**Завдання №1 VNS. Лабораторна робота №4**  
Плановий час на реалізацію: 1.5 години

**Завдання №2 VNS. Лабораторна робота №5**







Плановий час на реалізацію: 3 години

**Завдання №3 Algotester task 2 . V - 3**

Плановий час на реалізацію: 2 години

**Завдання №4 VNS . Algotester task 3. V – 2**

Плановий час на реалізацію: 2 години

**Завдання №5 Class Practice Work**

Плановий час на реалізацію: 2 години

**Завдання №6 Self Practice Algotester Task**

Плановий час на реалізацію: 1 година

***3. Код програми та фактичний час на реалізацію:***

**Завдання №1 VNS. Лабораторна робота №4**

```
1  #include <iostream>
2  #include <cstdlib>
3  #include <ctime>
4
5  using namespace std;
6
7  const int N = 100;
8
9  void array_print(int a[], int size)
10 {
11     for (int i = 0; i < size; ++i)
12     {
13         cout << a[i] << " ";
14     }
15     cout << endl;
16 }
17
18 int main()
19 {
20
21     int array[N];
22     int size;
23
24     cout << "Enter the size of the array: ";
25     cin >> size;
26
27     srand(time(0));
28
29     for (int i = 0; i < size; ++i)
30     {
31         array[i] = rand() % 100;
32     }
33
34     cout << "Array: ";
35     array_print(array, size);
36
37     int element_remove;
38     cout << "Enter the element of the array you want to remove: ";
39     cin >> element_remove;
```

```

40
41     if (element_remove >= 0 && element_remove < size)
42     {
43         for (int i = element_remove; i < size - 1; ++i)
44         {
45             array[i] = array[i + 1];
46         }
47         size--;
48     }
49     else
50     {
51         cout << "Invalid index(" << endl;
52     }
53
54     for (int i = 0; i < size; ++i)
55     {
56         if (array[i] % 2 == 0)
57         {
58             for (int j = size; j > i + 1; --j)
59             {
60                 array[j] = array[j - 1];
61             }
62             array[i + 1] = array[i] + 2;
63             size++;
64
65             break;
66         }
67     }
68     cout << "New array: " << endl;
69     array_print(array, size);
70
71     return 0;
72 }

```

Фактичний час на реалізацію: 1 година

**Завдання №2 VNS.** Лабораторна робота №4

```

#include <iostream>

using namespace std;

const int r = 4;
const int c = 4;

void matrix_print(int matrix[r][c])
{
    for (int i = 0; i < r; i++)
    {
        for (int j = 0; j < c; j++)
        {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }
}

```

```

void saddle_point(int matrix[r][c])
{
    for (int i = 0; i < r; ++i)
    {
        int min_r = matrix[i][0];
        int c_index = 0;

        for (int j = 1; j < c; ++j)
        {
            if (matrix[i][j] < min_r)
            {
                min_r = matrix[i][j];
                c_index = j;
            }
        }
    }
}

```

```

    bool is_saddle_point = true;
    for (int k = 0; k < r; ++k)
    {
        if (matrix[k][c_index] > min_r)
        {
            is_saddle_point = false;
            break;
        }
    }

    if (is_saddle_point)
    {
        cout << "Saddle point at (" << i << ", " << c_index << "), value: " << min_r << endl;
    }
}

```

```

    for (int j = 0; j < c; ++j)
    {
        int min_c = matrix[0][j];
        int r_index = 0;

        for (int i = 1; i < r; ++i)
        {
            if (matrix[i][j] < min_c)
            {
                min_c = matrix[i][j];
                r_index = i;
            }
        }
    }
}

```

```

        bool isSaddlePoint = true;
        for (int k = 0; k < c; ++k)
        {
            if (matrix[r_index][k] > min_c)
            {
                isSaddlePoint = false;
                break;
            }
        }

        if (isSaddlePoint)
        {
            cout << "Saddle point at (" << j << ", " << r_index << "), value: " << min_c << endl;
        }
    }
}

int main()
{
    int matrix[r][c] = {{6, 7, 8, 9}, {1, 2, 3, 4}, {5, 6, 7, 8}, {1, 0, 11, 12}};
    cout << "Matrix:" << endl;
    matrix_print(matrix);
    saddle_point(matrix);
    return 0;
}

```

Фактичний час на реалізацію: 2 години

### Завдання №3 Algotester task 2. V- 3

```

1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int N;
8      cin >> N;
9      int array[N];
10
11     for (int i = 0; i < N; i++)
12     {
13         cin >> array[i];
14     }
15
16     int left = 0;
17     int right = N - 1;
18
19     for (int i = 0; i < N; i++)
20     {
21
22         if (left == right)
23         {
24             cout << left + 1 << " " << right + 1 << endl;
25             cout << "Collision";
26             break;
27         }
28         if (left + 1 == right)
29         {
30             cout << left + 1 << " " << right + 1 << endl;
31             cout << "Stopped";
32             break;
33         }
34

```

```

34
35     if (left > right)
36     {
37         cout << left + 1 << " " << right + 1 << endl;
38         cout << "Miss";
39         break;
40     }
41     left += array[left];
42     right -= array[right];
43
44
45     return 0;
46 }

```

Фактичний час на реалізацію: 2 години

#### Завдання №4 VNS. Algotester task 3. V- 2

```

1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5
6  int main()
7  {
8      int N;
9      cin >> N;
10     vector<int> array1(N);
11     for (int i = 0; i < N; i++)
12     {
13         cin >> array1[i];
14     }
15     int M;
16     cin >> M;
17     vector<int> array2(M);
18     for (int i = 0; i < M; i++)
19     {
20         cin >> array2[i];
21     }
22     int common = 0;
23     int unique = 0;
24     for (int i = 0; i < N; i++)
25     {
26         for (int j = 0; j < M; j++)
27         {
28             if (array1[i] == array2[j])
29             {
30                 common++;
31                 break;
32             }
33         }
34     }
35     unique = N + M - common;
36     cout << common << endl;
37     cout << unique;
38     return 0;
39 }

```

Фактичний час на реалізацію: 1 година

## Завдання №5 Class Practice Work

```
1  #include <iostream>
2  #include <string>
3  #include <algorithm>
4
5  using namespace std;
6
7  bool is_palindrome(const string &str, int start, int end)
8  {
9      if (start > end)
10     {
11         return true;
12     }
13     if (str[start] != str[end])
14     {
15         return false;
16     }
17     return is_palindrome(str, start + 1, end - 1);
18 }
19
20 bool is_palindrome(int number)
21 {
22     int original = number;
23     int reversed = 0;
24     if (number < 0)
25     {
26         return false;
27     }
28     while (number > 0)
29     {
30         int new_number = number % 10;
31         reversed = reversed * 10 + new_number;
32         number /= 10;
33     }
34     return original == reversed;
35 }
36
```



```

37  int main()
38  {
39      string str;
40      cout << "\nEnter a word: ";
41      cin >> str;
42
43      if (is_palindrome(str, 0, str.length() - 1))
44      {
45          cout << "Your word is a palindrome";
46      }
47      else
48      {
49          cout << "Your word is not a palindrome";
50      }
51
52      int number;
53      cout << "\n Enter a number: ";
54      cin >> number;
55      if (is_palindrome(number))
56      {
57          cout << "Your number is a palindrome";
58      }
59      else
60      {
61          cout << "Your number is not a palindrome";
62      }
63
64      return 0;
65  }

```

Фактичний час на реалізацію: 2 години

**Завдання №6 Self Practice Algotester Task**

```

1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  int main()
7  {
8      string string;
9      cin >> string;
10     int counter = 1;
11     for (int i = 0; i < string.length(); i++)
12     {
13         if (string[i] == string[i + 1])
14             counter++;
15         else
16         {
17             if (counter == 1)
18                 cout << string[i];
19             else
20                 cout << string[i] << counter;
21             counter = 1;
22         }
23     }
24     return 0;
25 }
26

```

Фактичний час на реалізацію: 1 година

#### 4. Результати виконання завдань, тестування:

Завдання №1 VNS. Лабораторна робота №4

```

Enter the size of the array: 5
Array: 67 86 92 56 74
Enter the element of the array you want to remove: 3
67 86 92 74
New array:
67 86 88 92 74

```

Завдання №2 VNS. Лабораторна робота №5

```

C:\C64\bin\gdb.exe -i interpreter -u
Matrix:
6 7 8 9
1 2 3 4
5 6 7 8
1 0 11 12
Saddle point at (0, 0), value: 6
Saddle point at (3, 1), value: 4

```

Завдання №3 Algotester task 2. V - 3

6 1 1 2 2 1 1 3 4 Stopped	6 2 5 2 3 2 1 5 3 Miss
------------------------------------	---------------------------------

### Завдання №4 Algotester task 3. V- 2

```

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

```

### Завдання №5 Class Practice Work

Enter a word: uiio Your word is not a palindrome Enter a number: 99998 Your number is not a palindrome	Enter a word: pop Your word is a palindrome Enter a number: 989 Your number is a palindrome
---	--

### Завдання №6 Self Practice Algotester Task

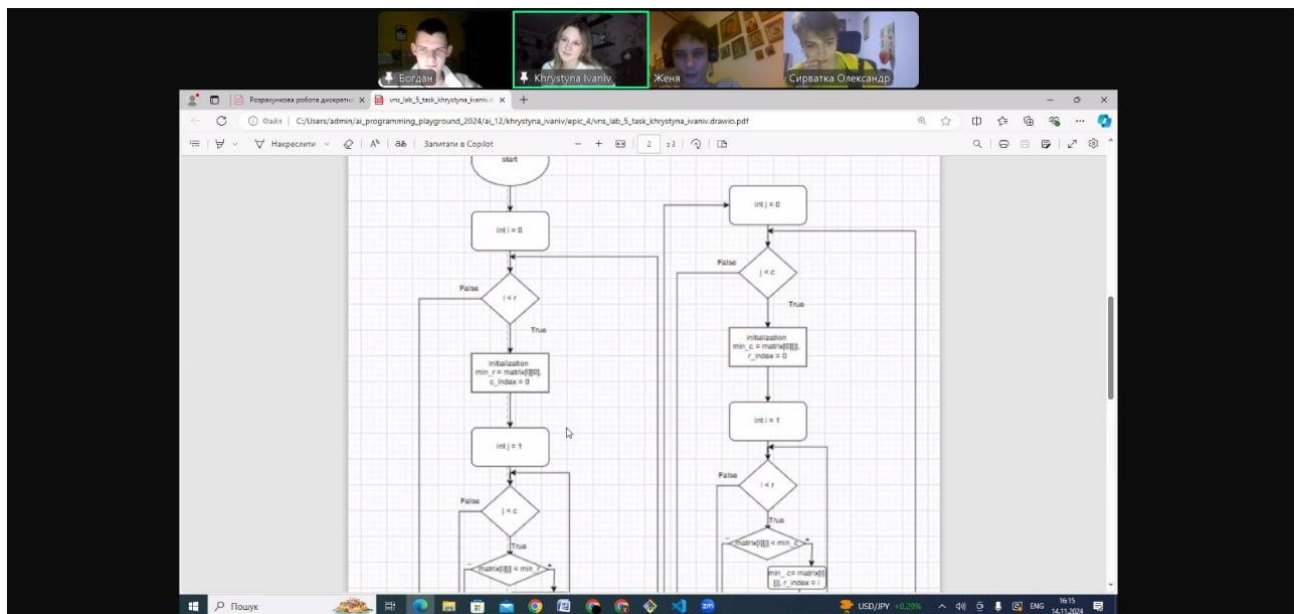
```

strrring
str3ing

```

### 5. Кооперація з командою:

*Провели зустріч у зумі, обговорили деталі виконання завдань*



**Висновок:** Під час виконання епіку я ознайомилась з одновимірними та двовимірними масивами, вказівниками та посиланнями, структурами даних, вкладеними структурами, алгоритмами обробки та роботою з масивами, структурами в мовах C та C++