

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



## **Звіт**

про виконання

### **Лабораторних та практичних робіт № 4**

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

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

***Виконав:***

студент групи ШІ-14

Маланій Олексій Миколайович

Львів 2023

## Тема роботи:

Розробка та впровадження програмного продукту згідно з ітерацією 4: вивчення теорії, управління вимогами, проектування, програмування лабораторних та практичних завдань, документування результатів і випуск готового продукту.

## Мета роботи:


Мета роботи полягає в успішному виконанні задач ітерації 4, зокрема, в освоєнні теоретичних аспектів, управлінні вимогами, проектуванні, програмуванні лабораторних та практичних завдань, а також документуванні результатів і успішному випуску готового програмного продукту.


## Теоретичні відомості:

1) Теоретичні відомості з переліком важливих тем:

- Epic 4 Task 1 - Lab# programming: VNS Lab 4
- Epic 4 Task 2 - Lab# programming: VNS Lab 5
- Epic 4 Task 3 - Lab# programming: Algotester Lab 2
- Epic 4 Task 4 - Lab# programming: Algotester Lab 3
- Epic 4 Task 5 - Practice# programming: Class Practice Task
- Epic 4 Task 6 - Practice# programming: Self Practice Task

2) Індивідуальний план опрацювання теорії:

- **Тема №1:** Epic 4 Task 1 - Lab# programming: VNS Lab 4
  - o Джерела Інформації
    -  Лабораторна робота № 4.pdf
    - Google
    - Одногорупники
    - Команда
  - o Що опрацьовано:
    - Код створює масив з 10 випадкових цілих чисел від 0 до 99, виводить його початковий вигляд, а потім модифікує, видаляючи елементи, які діляться на 7, та вставляючи після кожного непарного елемента нуль. Змінений масив виводиться на екран, демонструючи виконані трансформації.
  - o Статус: Ознайомлений частково
  - o Початок опрацювання теми: 25.11.2023
  - o Звершення опрацювання теми: 1.12.2023

- **Тема №2:** Epic 4 Task 2 - Lab# programming: VNS Lab 5
  - Джерела Інформації:
    -  Лабораторна робота № 5.pdf
    - Google
    - Одногопники
    - Команда
  - Що опрацьовано:
    - Код виконує видалення рядків в масиві за вказаними індексами, замінюючи їх на нулі та зсуваючи залишкові рядки. Після виконання операції видалення виводиться оновлений масив.
  - Статус: Ознайомлений частково
  - Початок опрацювання теми: 25.11.2023
  - Звершення опрацювання теми: 1.12.2023
  
- **Тема №3:** Epic 4 Task 3 - Lab# programming: Algotester Lab 2
  - Джерела Інформації:
    - <https://algotester.com/en/ContestProblem/DisplayWithEditor/134637>
    - Google
    - Одногопники
    - Команда
  - Що опрацьовано:
    - Код на C++ реалізує симуляцію змагання між двома дронами, що рухаються вліво та вправо залежно від вказаних позицій у векторі. Функція `game` визначає результат змагання та повертає пару значень: позицію переможця та статус (зіткнення, пропуск або зупинка). У головній функції користувач вводить кількість дронів та їхні початкові позиції, після чого виводиться результат гонки.
  - Статус: Ознайомлений частково
  - Початок опрацювання теми: 25.11.2023
  - Звершення опрацювання теми: 1.12.2023
  
- **Тема №4:** Epic 4 Task 4 - Lab# programming: Algotester Lab 3
  - Джерела Інформації:
    - <https://algotester.com/en/ContestProblem/DisplayWithEditor/134638>
    - Google
    - Одногопники
    - Команда

- Що опрацьовано:
    - Код приймає два вектори, підраховує кількість спільних елементів та виводить цю кількість, а також загальну кількість унікальних елементів обох векторів після їх об'єднання та сортування. Результати виводяться на екран.
  - Статус: Ознайомлений частково
  - Початок опрацювання теми: 24.11.2023
  - Звершення опрацювання теми: 1.12.2023
- **Тема №5:** Epic 4 Task 5 - Practice# programming: Class Practice TaskM
- Джерела Інформації:
    - 
    - Google
    - Одногопники
    - Команда
  - Що опрацьовано:
    -
  - Статус: Частково Ознайомлений
  - Початок опрацювання теми: 25.11.2023
  - Звершення опрацювання теми: 30.11.2023
- **Тема №6:** Epic 4 Task 6 - Practice# programming: Self Practice Task
- Джерела Інформації:
    - 
    - Google
    - Одногопники
    - Команда
  - Що опрацьовано:
    -
  - Статус: Частково Ознайомлений
  - Початок опрацювання теми: 25.11.2023
  - Звершення опрацювання теми: 30.11.2023

## Виконання роботи:

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

#### Завдання №1 VNS Lab 4

- 5
- У цьому завданні створюється масив з 10 випадкових цілих чисел в межах від 0 до 99. Потім програма видаляє елементи, які є кратними 7, та вставляє нулі після непарних елементів. Результат виводиться на екран у вигляді відредагованого масиву.
- Введення та валідація даних, Обчислення, Вивід результатів.

#### Завдання №2 VNS Lab 5

- 5
- У цьому завданні програма працює з двовимірним масивом розміром 4x4, виконуючи операції зміни та видалення рядків. Спочатку виводиться початковий вигляд масиву, а потім видаляються рядки від 1 до 2, включно, замінюючи їх на нулі. Результат виводиться на екран.
- Введення та валідація даних, Обчислення, Вивід результатів.

#### Завдання №3 Algotester Lab 2

- 2
- У цьому завданні програма працює з двовимірним масивом розміром 4x4, виконуючи операції зміни та видалення рядків. Спочатку виводиться початковий вигляд масиву, а потім видаляються рядки від 1 до 2, включно, замінюючи їх на нулі. Результат виводиться на екран.
- Правильність введених даних, Порядок операцій, Вивід результатів.

#### Завдання №4 Algotester Lab 3

- 3
- Програма на C++, яка приймає два масиви цілих чисел, обчислює кількість спільних елементів у них та виводить цю кількість, а також загальну кількість унікальних елементів у обох масивах.
- Порядок операцій, Вивід результатів.

#### Завдання №5 Class Practice Task

- ...
- У цьому завданні визначається функція для перевірки, чи є задане слово або число паліндромом. Потім використовуються ці функції для перевірки, чи є задане слово "stats" та число 45454 паліндромами, і виводите відповідне повідомлення про результат у консоль.
- Порядок операцій, Вивід результатів.

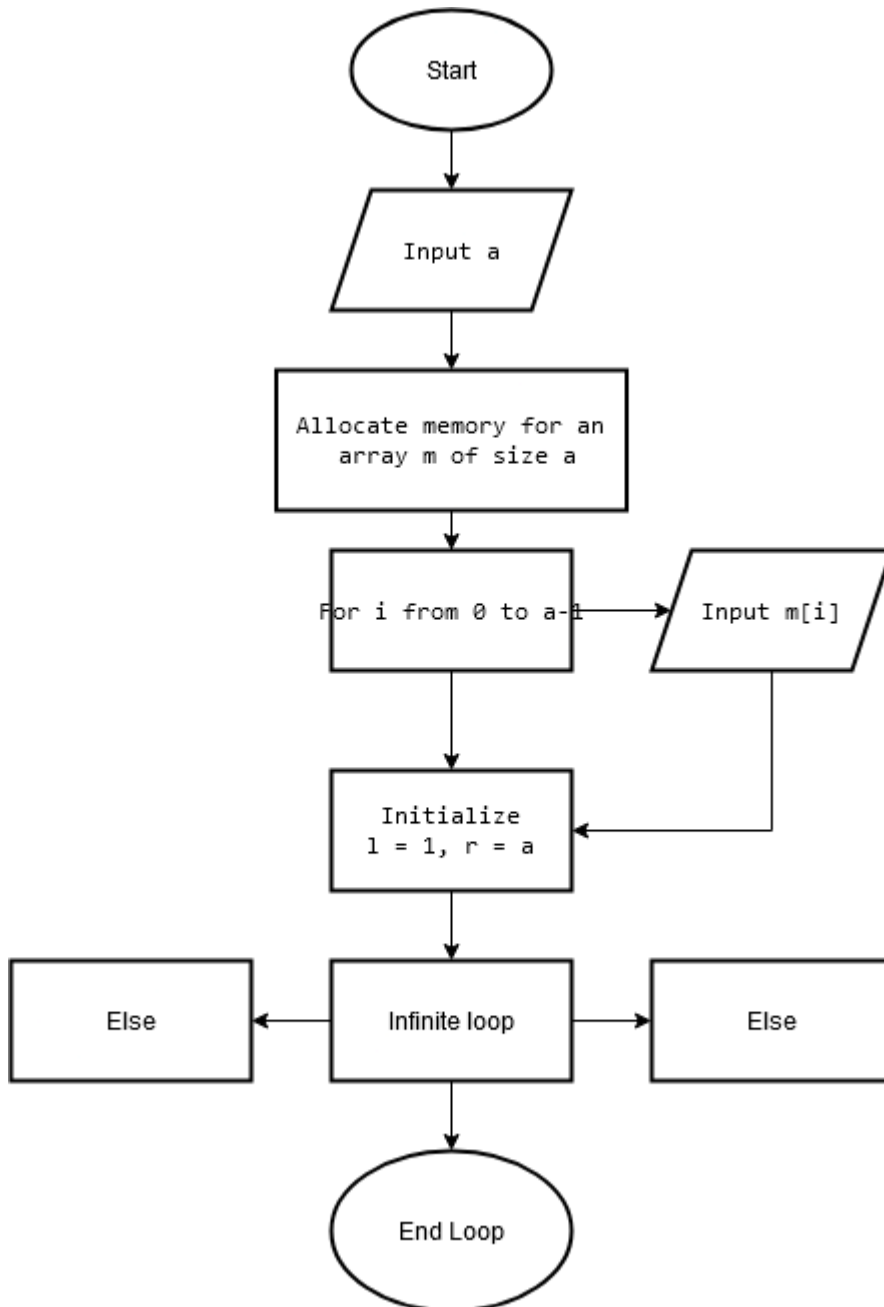
#### Завдання №6 Self Practice Task

- ...
- Ця програма призначена для розміщення офісів компаній на вулиці в порядку зростання їхніх довжин з метою мінімізації загальної відстані від точки 0 до усіх офісів. Введіть кількість компаній та довжини їхніх офісів, а програма виведе порядок побудови офісів.
- Порядок операцій, Вивід результатів.

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

### Програма №1

- Блок-схема:



- Планований час на реалізацію: 2-3 год
- Динамічне виділення пам'яті, Введення та обробка масиву, Цикл та умови, Управління циклом, Вивід результатів, Індикація масиву, Управління пам'яттю, Валідація введених даних

### 3. Конфігурація середовища до виконання завдань:

...

### 4. Код програм з посиланням на зовнішні ресурси:

Завдання №1 VNS Lab 4

```
1  #include <iostream>
2  #include <cstdlib>
3  #include <ctime>
4  #define size 10
5  using namespace std;
6  int main() {
7      int mas[size];
8      srand(time(NULL));
9      for (int i = 0; i < size; i++) {
10         mas[i] = rand() % 100;
11     }
12     cout << "Original: ";
13     for (int i = 0; i < size; i++) {
14         cout << mas[i] << " ";
15     }
16     cout << endl;
17     int nsize = 0;
18     for (int i = 0; i < size; i++) {
19         if (mas[i] % 7 != 0) {
20             mas[nsize] = mas[i];
21             nsize++;
22         }
23     }
24     int upsize = nsize;
25     for (int i = 0; i < size; i++) {
26         if (mas[i] % 2 != 0) {
27             upsize++;
28
29             for (int j = upsize - 1; j > i + 1; j--) {
30                 mas[j] = mas[j - 1];
31             }
32
33             mas[i + 1] = 0;
34             i++;
35         }
36     }
37     cout << "Edited: ";
38     for (int i = 0; i < upsize; i++) {
39         cout << mas[i] << " ";
40     }
41     cout << endl;
42     return 0;
43 }
44
```

Опис завдання:

У цьому завданні створюється масив з 10 випадкових цілих чисел в межах від 0 до 99. Потім програма видаляє елементи, які є кратними 7, та вставляє нулі після непарних елементів. Результат виводиться на екран у вигляді відредагованого масиву.

Завдання №2 VNS Lab 5



▼ Click here to ask blackbox to help you code faster

```
1  #include <iostream>
2  using namespace std;
3  const int NUM_ROWS = 6;
4  const int NUM_COLS = 4;
5  int** deleteRow(int** matrix, int rowToDelete) {
6      if (rowToDelete < 0 || rowToDelete >= NUM_ROWS) {
7          cout << "Invalid row index\n";
8          return matrix;
9      }
10
11     for (int i = rowToDelete; i < NUM_ROWS - 1; ++i) {
12         for (int j = 0; j < NUM_COLS; ++j) {
13             matrix[i][j] = matrix[i + 1][j];
14         }
15     }
16
17     for (int j = 0; j < NUM_COLS; ++j) {
18         matrix[NUM_ROWS - 1][j] = 0;
19     }
20
21     return matrix;
22 }
23 int** deleteRowsInRange(int** matrix, int A, int B) {
24     for (int i = A; i <= B; ++i) {
25         matrix = deleteRow(matrix, A);
26     }
27     return matrix;
28 }
29 void printMatrix(int** matrix) {
30     for (int i = 0; i < NUM_ROWS; ++i) {
31         for (int j = 0; j < NUM_COLS; ++j) {
32             cout << matrix[i][j] << '\t';
33         }
34         cout << '\n';
35     }
36 }
37 int main() {
38     int A = 1, B = 3;
39     if (A > B) {
40         return 1;
41     }
42     int** matrix = new int*[NUM_ROWS];
43     for (int i = 0; i < NUM_ROWS; ++i)
44         matrix[i] = new int[NUM_COLS];
45
46     int s = 0;
47     for (int i = 0; i < NUM_ROWS; ++i) {
```

```

47     for (int i = 0; i < NUM_ROWS; ++i) {
48         for (int j = 0; j < NUM_COLS; ++j) {
49             matrix[i][j] = ++s;
50         }
51     }
52     cout << "Initial matrix:\n";
53     printMatrix(matrix);
54     cout << "Delete " << A << "-" << B << " rows" << endl;
55     matrix = deleteRowsInRange(matrix, A, B);
56     cout << "Result matrix:\n";
57     printMatrix(matrix);
58     for (int i = 0; i < NUM_ROWS; ++i)
59         delete[] matrix[i];
60     delete[] matrix;
61     return 0;
62 }
63

```

Опис завдання:

У цьому завданні програма працює з двовимірним масивом розміром 4x4, виконуючи операції зміни та видалення рядків. Спочатку виводиться початковий вигляд масиву, а потім видаляються рядки від 1 до 2, включно, замінюючи їх на нулі. Результат виводиться на екран.

Завдання №3 Algotester Lab 2

```

Click here to ask Blackbox to help you code taster
1  #include <iostream>
2  using namespace std;
3  int main(){
4      int a;
5      cin >> a;
6      int* m = new int[a];
7      for (int i = 0; i < a; i++){
8          cin >> m[i];
9      }
10     int l = 1;
11     int r = a;
12     while(true){
13         if (r == l){
14             cout << l << " " << r << endl;
15             cout << "Collision";
16             break;
17         }
18         else if (r == l + 1){
19             cout << l << " " << r << endl;
20             cout << "Stopped";
21             break;
22         }
23         else if (r < l){
24             cout << l << " " << r << endl;
25             cout << "Miss";
26             break;
27         }
28         l = l + m[l - 1];
29         r = r - m[r - 1];
30     }
31 }
32

```

Опис завдання:

У цьому завданні програма працює з двовимірним масивом розміром 4x4, виконуючи операції зміни та видалення рядків. Спочатку виводиться початковий вигляд масиву, а потім видаляються рядки від 1 до 2, включно, замінюючи їх на нулі. Результат виводиться на екран.

Завдання №4 Algotester Lab 3

```
1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4  using namespace std;
5  int main() {
6      int n, m;
7      cin >> n;
8      vector<int> a(n);
9      for (int i = 0; i < n; ++i) {
10         cin >> a[i];
11     }
12     cin >> m;
13     vector<int> b(m);
14     for (int i = 0; i < m; ++i) {
15         cin >> b[i];
16     }
17     int element = 0;
18     for (int i = 0; i < n; ++i) {
19         if (find(b.begin(), b.end(), a[i]) != b.end()) {
20             element++;
21         }
22     }
23     vector<int> com;
24     com.insert(com.end(), a.begin(), a.end());
25     com.insert(com.end(), b.begin(), b.end());
26     sort(com.begin(), com.end());
27     com.erase(unique(com.begin(), com.end()), com.end());
28     cout << element << endl;
29     cout << com.size() << endl;
30     return 0;
31 }
32
```

Опис завдання:

- Програма на C++, яка приймає два масиви цілих чисел, обчислює кількість спільних елементів у них та виводить цю кількість, а також загальну кількість унікальних елементів у обох масивах.

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4  bool ispalindrome(const string& str, int start, int end) {
5      if (start >= end) {
6          return true;
7      }
8      if (str[start] == str[end]) {
9          return ispalindrome(str, start + 1, end - 1);
10     } else {
11         return false;
12     }
13 }
14 bool ispalindrome(int num) {
15     string numStr = to_string(num);
16     return ispalindrome(numStr, 0, numStr.length() - 1);
17 }
18 int main() {
19     string word = "stats";
20     int number = 45454;
21     if (ispalindrome(word, 0, word.length() - 1)) {
22         cout << word << " - palindrome." << endl;
23     } else {
24         cout << word << " - not palindrome." << endl;
25     }
26     if (ispalindrome(number)) {
27         cout << number << " - palindrome." << endl;
28     } else {
29         cout << number << " - not palindrome." << endl;
30     }
31     return 0;
32 }
```

Опис завдання:

У цьому завданні визначається функція для перевірки, чи є задане слово або число паліндромом. Потім використовуються ці функції для перевірки, чи є задане слово "stats" та число 45454 паліндромами, і виводите відповідне повідомлення про результат у консоль.

## Завдання №6 Self Practice Task

```
1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4  using namespace std;
5  struct office {
6      int index;
7      int length;
8  };
9  bool compOffices(const office& a, const office& b) {
10     return a.length < b.length;
11 }
12 int main() {
13     int n;
14     cin >> n;
15     vector<office> offices(n);
16     for (int i = 0; i < n; ++i) {
17         cin >> offices[i].length;
18         offices[i].index = i + 1;
19     }
20     sort(offices.begin(), offices.end(), compOffices);
21     for (int i = 0; i < n; ++i) {
22         cout << offices[i].index << " ";
23     }
24     return 0;
25 }
26
```

Опис завдання:

Ця програма призначена для розміщення офісів компаній на вулиці в порядку зростання їхніх довжин з метою мінімізації загальної відстані від точки 0 до усіх офісів. Введіть кількість компаній та довжини їхніх офісів, а програма виведе порядок побудови офісів.

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

Завдання №1 VNS Lab4

```
Original: 18 59 99 88 26 94 10 21 38 89  
Edited: 18 59 0 99 0 88 26 94 10 38 89  
PS C:\Users\alexw\OneDrive\Робочий стіл\LPNU\EPIC4>
```

Завдання зайняло 3 год.

Завдання №2 VNS Lab 5

```
Initial matrix:  
1      2      3      4  
5      6      7      8  
9      10     11     12  
13     14     15     16  
17     18     19     20  
21     22     23     24  
Delete 1-3 rows  
Result matrix:  
1      2      3      4  
17     18     19     20  
21     22     23     24  
0      0      0      0  
0      0      0      0  
0      0      0      0  
PS C:\Users\alexw\OneDrive\Робочий стіл\LPNU\EPIC4>
```

Завдання зайняло 4 год.

### Завдання №3 Algotester Lab 2

```
10
1 3 1 1 5 1 1 2 1 2
5 6
Stopped
PS C:\Users\alexw\OneDrive\Робочий стіл\LPNU\EPIC4>
sk_3_Oleksii_Malanii }
10
1 3 1 1 5 1 1 3 1 2
5 5
Collision
PS C:\Users\alexw\OneDrive\Робочий стіл\LPNU\EPIC4>
sk_3_Oleksii_Malanii }
10
1 3 1 1 5 1 1 5 1 2
5 3
Miss
```

Завдання зайняло 3 год.

### Завдання №4 Algotester Lab 3

```
5
1 2 3 4 5
5
4 5 6 7 8
2
8
PS C:\Users\alexw\OneDrive\Робочий стіл\LPNU\EPIC4>
nii }
5
1 2 3 4 5
5
6 7 8 9 10
0
10
PS C:\Users\alexw\OneDrive\Робочий стіл\LPNU\EPIC4>
```

Завдання зайняло 3 год.

## Завдання №5 Class Practice Task

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4  bool ispalindrome(const string& str, int start, int end) {
5      if (start >= end) {
6          return true;
7      }
8      if (str[start] == str[end]) {
9          return ispalindrome(str, start + 1, end - 1);
10     } else {
11         return false;
12     }
13 }
14 bool ispalindrome(int num) {
15     string numStr = to_string(num);
16     return ispalindrome(numStr, 0, numStr.length() - 1);
17 }
18 int main() {
19     string word = "stats";
20     int number = 45454;
21     if (ispalindrome(word, 0, word.length() - 1)) {
22         cout << word << " - palindrome." << endl;
23     } else {
24         cout << word << " - not palinrome." << endl;
25     }
26     if (ispalindrome(number)) {
27         cout << number << " - palindrome." << endl;
28     } else {
29         cout << number << " - not palinrome." << endl;
30     }
31     return 0;
32 }
33
```

Завдання зайняло 3 год.



## Завдання №5 Self Practice Task

```
1  ✓ #include <iostream>
2  ✓ #include <vector>
3  ✓ #include <algorithm>
4  ✓ using namespace std;
5  ✓ struct office {
6  |     int index;
7  |     int length;
8  | };
9  ✓ bool compOffices(const office& a, const office& b) {
10 |     return a.length < b.length;
11 | }
12 ✓ int main() {
13 |     int n;
14 |     cin >> n;
15 |     vector<office> offices(n);
16 ✓ |     for (int i = 0; i < n; ++i) {
17 |         cin >> offices[i].length;
18 |         offices[i].index = i + 1;
19 |     }
20 |     sort(offices.begin(), offices.end(), compOffices);
21 ✓ |     for (int i = 0; i < n; ++i) {
22 |         cout << offices[i].index << " ";
23 |     }
24 |     return 0;
25 | }
26
```

Завдання зайняло 2 год.

## Висновки:

У процесі виконання роботи було успішно опрацьовано всі завдання, пов'язані з виконанням лабораторних робіт.