Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

**про виконання лабораторних та практичних робіт блоку № 5**

На тему:  «Файли. Системи числення. Бінарні Файли. Символи і Рядкові Змінні та Текстові Файли. Стандартна бібліотека та деталі/методи роботи з файлами. Створення й використання бібліотек.»

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

до:

ВНС Лабораторної Роботи № 6

ВНС Лабораторної Роботи № 8

ВНС Лабораторної Роботи № 9

Алготестер Лабораторної Роботи №4

Алготестер Лабораторної Роботи №6

Практичних Робіт №5

***Виконав:***

студент групи ШІ-12

Воробець Максим Володимирович

# **Тема роботи:**

Файли. Символи і рядкові змінні. Текстові Файли. Стандартна бібліотека та деталі роботи з файлами.

# **Мета роботи:**

Розглянути роботу з символьними та рядковими змінними у мові програмування. Розвинути навички опрацювання символьних та рядкових даних. Вивчити методи відкриття, читання та запису текстових файлів. Розуміти операції із текстовими файлами. Розглянути концепції створення та використання бібліотек.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Файли
* Тема №2: Символи і рядкові змінні. Текстові файли.
* Тема №3: Стандартна бібліотека та деталі роботи з файлами.

1. Індивідуальний план опрацювання теорії:

* Тема №1: Файли
  + Джерела Інформації
    - Стаття: [C++. Файлова система C++. Загальні принципи роботи. Приклади. Відкриття/закриття файлу | BestProg](https://www.bestprog.net/uk/2019/09/07/c-file-system-general-principles-of-work-examples-open-close-file-ua/)
  + Що опрацьовано:
    - Опрацьовано теорію по даній темі
  + Статус: Ознайомлений частково
  + Початок опрацювання теми: 17.12.2023
  + Звершення опрацювання теми: 18.12.2023
* Тема №2: Символи і рядкові змінні. Текстові файли.
  + Джерела Інформації:
    - Стаття: [» Основні функції обробки символьних типів C++ програмування (cpp.dp.ua)](http://cpp.dp.ua/osnovni-funktsiyi-obrobky-symvolnyh-typiv/)
  + Що опрацьовано:
    - Опрацьовані теоретичні та практичні навички по даній темі.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 19.12.2023
  + Звершення опрацювання теми: 20.12.2023
* Тема №3: Стандартна бібліотека та деталі роботи з файлами.
  + Джерела Інформації:
    - Стаття: [C++. Бібліотека стандартних шаблонів STL. Загальні поняття | BestProg](https://www.bestprog.net/uk/2021/09/26/c-standard-template-library-general-concepts-ua/)
  + Що опрацьовано:
    - Опрацьовано теорію по даній темі
  + Статус: Ознайомлений
  + Початок опрацювання теми: 15.12.2023
  + Звершення опрацювання теми: 15.12.2023

# **Виконання роботи:**

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

Завдання №1 **VNS Lab 6 - Task 1-N**

* Варіант завдання: 3
* Деталі завдання:

Надрукувати всі слова, які містять по одній цифрі.

Завдання №2 **VNS Lab 9 - Task 1-N**

* Варіант завдання: 3
* Деталі завдання:

1) Скопіювати у файл F2 тільки ті рядки з F1, які починаються й закінчуються

на ту саму букву.

2) Підрахувати кількість символів в F2.

Завдання №3(1,2) **Algotester Lab 4**

* Варіант завдання: 2
* Деталі завдання:

<https://algotester.com/en/ContestProblem/DisplayWithEditor/134641>

Завдання №4 **Algotester Lab 6**

* Варіант завдання: 2
* Деталі завдання:

<https://algotester.com/en/ContestProblem/DisplayWithEditor/134743>

Завдання №5.1 **Class Practice Work - Task 1-N**

* Варіант завдання: 2
* Деталі завдання:

-       створити файл із заданим ім’ям; якщо файл існує – перезаписати його вміст

-       написати код стійкий до різних варіантів вхідних параметрів

-       name – ім’я, може не включати шлях

-       записати у файл вміст стрічки content, прочитати content із стандартного вводу

-       повернути статус операції: Success – все пройшло успішно, Failure – файл не вдалося створити, або збій операції відкриття, запису даних, чи закриття файла.

Завдання №5.2 **Class Practice Work - Task 2-N**

* Варіант завдання: 2
* Деталі завдання:

-       копіювати вміст файла з ім’ям file\_from у файл з ім’ям file\_to; написати код стійкий до різних варіантів вхідних параметрів, обробити всі можливі варіанти відсутності одного з файлів

-       file\_from, file\_to – можуть бути повним або відносним шляхом

-       повернути статус операції: Success – все пройшло успішно, Failure – файл не вдалося створити, або збій операції відкриття, читання чи запису даних, закриття файла.

Завдання №6 **Self Practice Work - Task 1-N**

* Варіант завдання: 2
* Деталі завдання:

<https://algotester.com/en/ArchiveProblem/DisplayWithEditor/13>

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

Програма №1 **VNS Lab 6 - Task 1-N**

* Блок-схема :

Зображення, що містить текст, знімок екрана, Шрифт, графічний дизайн

Автоматично згенерований опис

Рисунок . Блок-схема до програми №1

* Планований час на реалізацію: 45хв

Програма №2 **VNS Lab 9 - Task 1-N**

* Планований час на реалізацію: 1год 10хв

Програма №3.1 **Algotester Lab 4**

* Планований час на реалізацію: 40хв

Програма №3.2 **Algotester Lab 4**

* Планований час на реалізацію: 1год

Програма №4 **Algotester Lab 6**

* Планований час на реалізацію: 2год 10хв

Програма №5.1 **Class Practice Work - Task 1-N**

* Планований час на реалізацію: 40хв

Програма №5.2

* Планований час на реалізацію: 1год

Програма №6

* Планований час на реалізацію: 30хв

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

Завдання №1 Програма влаштована так, щоб користувач ввів рядок слів за допомогою функції gets.Я використав функцію strtok для розділення рядка на слова за роздільником пробілу. Поки рядок не буде дорівнювати значенню 0, що означає закінчення рядка, програма за допомогою циклу for та умовного оператора if програма дізнається чи слово містить лише одну цифру. Якщо так, то виводимо це слово, якщо ні, тоді переходим до наступного слова та повторюємо цикл.

#include <iostream>

#include <cstring>

using namespace std;

int main() {

    char rad[255];

    cout<<"Введіть рядок: ";

    gets(rad);

    char\* scape;

    scape=strtok(rad, " ");

    while(scape!=0){

        int num=0;

        for(int i=0; scape[i]!=0; i++){

            if(isdigit(scape[i])){

                num++;

            }

        }

        if(num==1){

            puts(scape);

        }

        scape=strtok(0, " ");

    }

    return 0;

}

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/4731e868128a7099463f0dffb1cbf1805eec402b/ai_12/maksym_vorobets/Epic5/vns_lab_6_task_maksym_vorobets.cpp>

Завдання №2 Я створюю файл F1, відкриваю його для запису за допомогою циклу for записую туди 10 рядків та закриваю його. Далі я відкриваю F1 для зчитування та створюю файл F2 для запису, щоб за допомогою циклу while(зупиниться коли рядок буде пустий) та умовного оператора if записати у файл F2 рядки, які починаються і закінчуються тою самою буквою. Тоді відкриваю F2 для зчитування, щоб порахувати скільки символів є у файлі F2.

#include <iostream>

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

int main() {

    ofstream f1("F1.txt");

    if (!f1.is\_open()) {

        cout << "Помилка" << endl;

    }

    for (int i = 0; i < 10; ++i) {

        string rad;

        getline(cin, rad);

        f1 << rad << '\n';

    }

    f1.close();

    ifstream f1\_read;

    ofstream f2;

    f1\_read.open("F1.txt");

    f2.open("F2.txt");

    if (!f1\_read.is\_open() && !f2.is\_open()) {

        cout<<"Помилка"<<endl;

    }else{

        string line;

        while (getline(f1\_read, line)) {

            if (!line.empty() && line.front() == line.back()) {

                f2 << line << '\n';

            }

        }}

        f1\_read.close();

        f2.close();

        ifstream f2\_read;

        f2\_read.open("F2.txt");

        if (!f2\_read.is\_open()) {

            cout<<"Помилка"<<endl;

        }else{

            f2\_read.seekg(0, ios::end);

            int r = f2\_read.tellg();

            cout << "Кількість символів в F2: " << r << endl;

        }

        f2\_read.close();

    return 0;

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/2625e1ee317b85aedd2f60196c3b9458748d04fe/ai_12/maksym_vorobets/Epic5/vns_lab_9_task_2_variant_3_maksym_vorobets.cpp>

Завдання №3.1 Для того щоб посортувати масив та видалити однакові значення з нього, я використав функцію множини set, так як множині властиві послідовність та неповторність. Потім цю множину перекидаю у масив за допомогою функції push\_back. Тоді за допомогою циклу обертаю підсортований масив згідно з K і виводжу на екран кількість елементів нового масиву та сам масив.

#include<iostream>

#include<vector>

#include<set>

using namespace std;

int main(){

    int N;

    cin>>N;

    int k;

    cin>>k;

    set<int>st;

    for (int i = 0; i < N; i++) {

        int x;

        cin>>x;

        st.insert(x);

    }

    int len = st.size();

    vector<int>a;

    for (auto p : st) {

        a.push\_back(p);

    }

    cout<<len<<endl;

    k %= len;

    for (int i = k; i < len; i++) {

        cout<<a[i]<<' ';

    }

    for (int i = 0; i < k; i++) {

        cout<<a[i]<<' ';

    }

    return 0;

}

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/2625e1ee317b85aedd2f60196c3b9458748d04fe/ai_12/maksym_vorobets/Epic5/algotester_lab_4_task_maksym_vorobets.cpp>

Завдання №3.2 Я використав функцію unordered\_set, щоб видалити елементи, які повторюються, з вектору a. Потім, щоб відсортувати матрицю, я використав функцію sort, а також фунцію rotate для перестановки значень матриці згідно K. Тоді програма виводить отриману матрицю.

#include <iostream>

#include <algorithm>

#include <vector>

#include <unordered\_set>

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];

    }

    unordered\_set<int> uniqueSet(a.begin(), a.end());

    a.assign(uniqueSet.begin(), uniqueSet.end());

    sort(a.begin(), a.end());

    int len = a.size();

    cout << len << endl;

    rotate(a.begin(), a.begin() + k%len, a.end());

    for (auto i : a) {

        cout << i <<' ';

    }

    return 0;

}

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/2625e1ee317b85aedd2f60196c3b9458748d04fe/ai_12/maksym_vorobets/Epic5/algotester_lab_4_variant_2_maksym_vorobets.cpp>

Завдання №4 За допомогою циклу for вводимо значення масиву a(шахова дошка розміром 8×8). Програма за допомогою циклів та умовних операторів показує чи атакує фігура певну клітинку. І тоді користувач вводить координати. Програма виводить чим зайнята ця координата.

#include<iostream>

#include<vector>

#include<string>

#include<algorithm>

#include<set>

using namespace std;

int main(){

    int x,y;

    vector<vector<char>>a(8,vector<char>(8));

    vector<vector<string>>b(8,vector<string>(8));

    for (int i = 0; i < 8; i++) {

        for (int j = 0; j < 8; j++) {

            cin>>a[i][j];

        }

    }

    for (int i = 0; i < 8; i++) {

        for (int j = 0; j < 8; j++) {

            if (a[i][j] == 'P') {

                if (i+1 != 8) {

                    if(j+1!=8){

                        b[i+1][j+1]+='P';

                    }

                    if(j-1>=0){

                        b[i+1][j-1]+='P';

                    }

                }

            } else if (a[i][j] == 'R') {

                x=i-1;

                y=j;

                while (x>=0 ){

                    b[x][y]+='R';

                    x--;

                }

                x=i+1;

                y=j;

                while (x<8 ){

                    b[x][y]+='R';

                    x++;

                }

                x=i;

                y=j-1;

                while (y>=0 ){

                    b[x][y]+='R';

                    y--;

                }

                x=i;

                y=j+1;

                while (y<8 ){

                    b[x][y]+='R';

                    y++;

                }

            } else if (a[i][j]=='N') {

                x=i+1;

                y=j+2;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='N';

                }

                x=i+2;

                y=j+1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='N';

                }

                x=i+1;

                y=j-2;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='N';

                }

                x=i-1;

                y=j+2;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='N';

                }

                x=i-2;

                y=j+1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='N';

                }

                x=i-2;

                y=j-1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='N';

                }

                x=i-1;

                y=j-2;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='N';

                }

                x=i+2;

                y=j-1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='N';

                }

            } else if (a[i][j]=='B'){

                x=i-1;

                y=j-1;

                while (x>=0 && y>=0 && x<8 && y<8 ) {

                    b[x][y]+='B';

                    x--;

                    y--;

                }

                x=i-1;

                y=j+1;

                while (x>=0 && y>=0 && x<8 && y<8 ) {

                    b[x][y]+='B';

                    x--;

                    y++;

                }

                x=i+1;

                y=j-1;

                while (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='B';

                    x++;

                    y--;

                }

                x=i+1;

                y=j+1;

                while (x>=0 && y>=0 && x<8 && y<8 ) {

                    b[x][y]+='B';

                    x++;

                    y++;

                }

            } else if (a[i][j]=='K'){

                x=i-1;

                y=j-1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='K';

                }

                x=i-1;

                y=j;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='K';

                }

                x=i-1;

                y=j+1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='K';

                }

                x=i;

                y=j-1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='K';

                }

                x=i;

                y=j+1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='K';

                }

                x=i+1;

                y=j-1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='K';

                }

                x=i+1;

                y=j;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='K';

                }

                x=i+1;

                y=j+1;

                if (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='K';

                }

            } else if (a[i][j]=='Q'){

                x=i-1;

                y=j;

                while (x>=0 ){

                    b[x][y]+='Q';

                    x--;

                }

                x=i+1;

                y=j;

                while (x<8 ){

                    b[x][y]+='Q';

                    x++;

                }

                x=i;

                y=j-1;

                while (y>=0){

                    b[x][y]+='Q';

                    y--;

                }

                x=i;

                y=j+1;

                while (y<8){

                    b[x][y]+='Q';

                    y++;

                }

                x=i-1;

                y=j-1;

                while (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='Q';

                    x--;

                    y--;

                }

                x=i-1;

                y=j+1;

                while (x>=0 && y>=0 && x<8 && y<8 ) {

                    b[x][y]+='Q';

                    x--;

                    y++;

                }

                x=i+1;

                y=j-1;

                while (x>=0 && y>=0 && x<8 && y<8 ) {

                    b[x][y]+='Q';

                    x++;

                    y--;

                }

                x=i+1;

                y=j+1;

                while (x>=0 && y>=0 && x<8 && y<8) {

                    b[x][y]+='Q';

                    x++;

                    y++;

                }

            }

        }

    }

    int test;

    cin>>test;

    while (test--){

        cin>>x>>y;

        x--;

        y--;

        if (a[x][y]!='O'){

            cout<<'X'<<endl;

        }else if (b[x][y].length()==0){

            cout<<'O'<<endl;

        }else{

            set<char>st;

            for (auto p : b[x][y]){

                st.insert(p);

            }

            for (auto p : st){

                cout<<p;

            }

            cout<<endl;

        }

    }

    return 0;

}

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/2625e1ee317b85aedd2f60196c3b9458748d04fe/ai_12/maksym_vorobets/Epic5/algotester_lab_6_variant_2_maksym_vorobets.cpp>

Завдання №5.1 Програма створює файл із заданим ім’ям, у який вносимо стрічку content та повертає статус операції: Success – все пройшло успішно, Failure – файл не вдалося створити, або збій операції відкриття, читання чи запису даних, закриття файла.

#include<iostream>

#include<fstream>

#include<string>

using namespace std;

enum FileOpResult {Success, Failure};

FileOpResult write\_to\_file(string name, string content)

{

    ofstream file;

    file.open(name);

    if(!file.is\_open()){

        return Failure;

    }else{

        file<<content<<endl;

        }

    file.close();

    return  Success;

}

int main()

{

    string content;

    string name="file.txt";

    cout<<"Введіть текст: "<<endl;

    getline(cin, content);

    FileOpResult res=write\_to\_file(name, content);

    if (res==Success){

        cout<<"Файл успішно заповнений"<<endl;

    }

    else if(res==Failure){

        cout<<"Помилка"<<endl;

    }

    return 0;

}

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/2625e1ee317b85aedd2f60196c3b9458748d04fe/ai_12/maksym_vorobets/Epic5/practice_work_task_1_maksym_vorobets.cpp>

Завдання №5.2  Програма копіює вміст файла з ім’ям file\_from у файл з ім’ям file\_to та повертає статус операції: Success – все пройшло успішно, Failure – файл не вдалося створити, або збій операції відкриття, читання чи запису даних, закриття файла.

#include<iostream>

#include<fstream>

#include<string>

using namespace std;

enum FileOpResult{Success, Failure};

FileOpResult copy\_file(string file\_from, string file\_to){

    ifstream file1;

    ofstream file2;

    file1.open(file\_from);

    file2.open(file\_to);

    if(!file1.is\_open() || !file2.is\_open()){

        return Failure;

    }else{

        string content;

        while(getline(file1, content)){

            file2<<content<<endl;

        }}

    file1.close();

    file2.close();

    return Success;

}

int main(){

    string file\_from="from.txt";

    string file\_to="to.txt";

    string text;

    cout<<"Введіть текст:"<<endl;

    getline(cin, text);

    ofstream file;

    file.open(file\_from);

    if(file.is\_open()){

        file<<text;

    }

    file.close();

    FileOpResult res=copy\_file(file\_from, file\_to);

    if(res==Success){

        cout<<"Ваш текст успішно скопійований"<<endl;

    }

    else if (res==Failure) {

        cout<<"Помилка"<<endl;

    }

    return 0;

}

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/2625e1ee317b85aedd2f60196c3b9458748d04fe/ai_12/maksym_vorobets/Epic5/practice_work_task_2_maksym_vorobets.cpp>

Завдання №6 Програма приймає вхідні дані щодо кількості раундів, а потім зчитує результат кожного раунду. Вона перевіряє, чи одна з команд набрала 11 очок з різницею принаймні 2, і оновлює загальний рахунок відповідно. Завершений рахунок виводиться, а також будь-який залишковий рахунок, якщо гра не завершилася.

#include <iostream>

#include <cmath>

using namespace std;

int main(){

    int n, k=0, v=0, resk=0, resv=0;

    cin>>n;

    char r;

    for(int i=0; i<n; i++){

        cin>>r;

        if(r=='K'){

            k++;

        }else if(r=='V'){

            v++;

        }

         if((k>= 11 || v>= 11) && abs(k-v) >= 2){

            if(r=='K'){

                resk++;

            }else if(r=='V'){

                resv++;

            }

        k=0;

        v=0;

         }

    }

    cout<<resk<<":"<<resv<<endl;

    if(k>0 || v>0){

        cout<<k<<":"<<v<<endl;

    }

   return 0;

}

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/2625e1ee317b85aedd2f60196c3b9458748d04fe/ai_12/maksym_vorobets/Epic5/self_practice_work_algotester_task_1_maksym_vorobets.cpp>

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

Завдання №1

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Рисунок . Результат програми №1

Час затрачений на виконання завдання: 45 хв

Завдання №2

Зображення, що містить текст, знімок екрана

Автоматично згенерований опис

Рисунок . Результат програми №2

Час затрачений на виконання завдання: 1год 10хв

Завдання №3.1 та №3.2

Зображення, що містить текст, Шрифт, число, ряд

Автоматично згенерований опис

Рисунок . Результат програми №3(1,2)

Час затрачений на виконання завдання: 1год 40хв

Завдання №4

Зображення, що містить текст, Шрифт, число, ряд

Автоматично згенерований опис

Рисунок . Результат програми №4

Час затрачений на виконання завдання: 2год 10хв

Завдання №5.1

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Рисунок . Результат програми №5.1

Час затрачений на виконання завдання: 40 хв

Завдання №5.2

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Рисунок . Результат програми №5.2

Час затрачений на виконання завдання: 1 год

Завдання №6

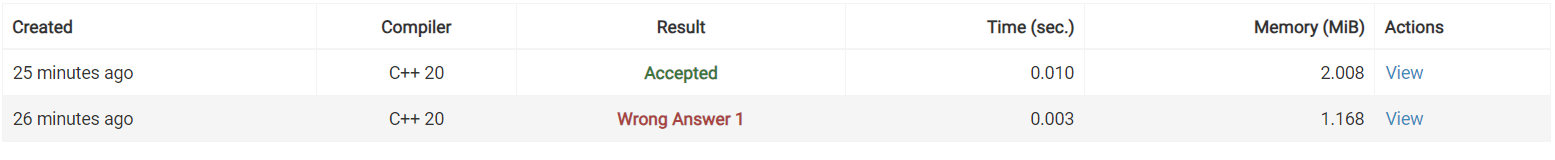


Рисунок . Результат програми №6

Час затрачений на виконання завдання: 30 хв

## **6. Кооперація з командою:**

Зображення, що містить знімок екрана, Обличчя людини, Мультимедійне програмне забезпечення, програмне забезпечення

Автоматично згенерований опис

Рисунок . Обговорення з командою

Зображення, що містить текст, знімок екрана, програмне забезпечення, Мультимедійне програмне забезпечення

Автоматично згенерований опис

Рисунок . Коментарі від двох членів команди

# **Висновки:**

У процесі вивчення теми я розвинув свої навички по опрацюванню символьних та рядкових даних у програмах, розглянув метод відкриття, читання та запису текстових файлів, яке дозволило зрозуміти особливості роботи з інформацією у текстовому форматі. Також я отримав важливі знання щодо концепцій створення та використання бібліотек.