

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. Бінарні Файли. Символи і Рядкові Змінні та Текстові Файли. Стандартна бібліотека та деталі/методи роботи з файлами.

Створення й використання бібліотек.»

**з дисципліни:** «Основи програмування»

до:

Практичних Робіт до блоку № 1

**Виконала:**

Студентка групи ІІІ-12  
Костак Олеся Михайлівна

Львів – 2024 р.

**Тема роботи:** Файли. Бінарні Файли. Символи і Рядкові Змінні та Текстові Файли. Стандартна бібліотека та деталі/методи роботи з файлами. Створення й використання бібліотек.

**Мета роботи:**

Освоїти роботу з файлами різних типів, зокрема текстовими та бінарними файлами, символами й рядковими змінними, а також методами стандартної бібліотеки для маніпулювання файлами. Розвинути навички створення й використання бібліотек для обробки даних із файлів, а також алгоритмічні вміння обробляти й аналізувати дані, збережені в різних форматах.

**Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:
  - Тема №\*.1: Вступ до Роботи з Файлами.
  - Тема №\*.2: Символи і Рядкові Змінні.
  - Тема №\*.3: Текстові Файли.
  - Тема №\*.4: Бінарні Файли.
  - Тема №\*.5: Стандартна бібліотека та робота з файлами.
  - Тема №\*.6: Створення й використання бібліотек.
2. Індивідуальний план опрацювання теорії:  
*Джерела Інформації:*  
Сайт [geeksforgeeks](http://geeksforgeeks.com)  
Сайт [programiz.com](http://programiz.com)  
Сайт [w3schools.com](http://w3schools.com)  
Ютуб-канал Блоган  
Лекції Пшеничного  
Сайт [cppreference.com](http://cppreference.com)

**Тема №\*.1: Вступ до Роботи з Файлами.**

- *Що опрацьовано:*
  - Відео [File Access Basics | C Programming Tutorial](#)
  - Стаття [C Library - fprintf\(\) function](#)
  - Стаття [C Library - fscanf\(\) function](#)

**Тема №\*.2: Символи і Рядкові Змінні.**

- *Що опрацьовано:*
  - Стаття [C++ gets\(\)](#)
  - Стаття [C++ strtok\(\)](#)

**Тема №\*.3: Текстові Файли.**

- *Що опрацьовано:*
  - Стаття [getline \(string\) in C++](#)
  - Стаття [cin.ignore\(\) Function in C++](#)

- Стаття [setw\(\) function in C++](#)

#### Тема №\*.4: Бінарні Файли.

- *Що опрацьовано:*
  - Стаття [Basics of File Handling in C](#)
  - Стаття [Set Position with seekg\(\) in C++ File Handling](#)

#### Тема №\*.5: Стандартна бібліотека та робота з файлами.

- *Що опрацьовано:*
  - Стаття [C++ fstream Library \(File Streams\)](#)

#### Тема №\*.6: Створення й використання бібліотек.

- *Що опрацьовано:*
  - Стаття [Static and Dynamic Libraries | Set 1](#)
  - Стаття [How Do I Create a Library in C?](#)

### Виконання роботи:

#### Завдання №1 VNS Lab 6 - Task 1

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

- *Варіант завдання: 5*

Перетворити рядок таким чином, щоб спочатку в ньому були надруковані тільки букви, а потім тільки цифри, не міняючи порядку проходження символів у рядку.

##### 3. Код програми

```

vns_lab_6_task_1_olesia_kosatk.cpp > main()
1  #include <stdio.h>
2  #include <string.h>
3  #include <iostream>
4  #include <cctype>
5
6  int main()
7  {
8      char myStr[256];
9      fgets(myStr, 256, stdin);
10     char nums_str[256] = "";
11     char res[256] = "";
12
13     char * token = strtok(myStr, " ");
14
15     while(token != NULL)
16     {
17         bool flag = true;
18
19         for(int i = 0; token[i] != '\0'; i++)
20             if(isdigit(token[i]))
21                 flag = false;
22
23         if(!flag)
24         {
25             strcat(nums_str, token);
26             strcat(nums_str, " ");
27         }
28         else
29         {
30             strcat(res, token);
31             strcat(res, " ");
32         }
33
34         token = strtok(NULL, " ");
35     }
36
37     strcat(res, nums_str);
38     std::cout << res;
39
40     return 0;
41 }

```

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

```
Hell 00 o wo 011 r1 22 d
Hell o wo r1 d 00 011 22
```

Час затрачений на виконання завдання: 20 хв

### Завдання VNS Lab 8 - Task 1

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

*Варіант завдання: 5*

*Деталі завдання:* Сформувати двійковий файл із елементів, заданої у варіанті структури, роздрукувати його вміст, виконати знищення й додавання елементів у відповідності зі своїм варіантом, використовуючи для пошуку елементів що знищуються чи додаються, функцію.

Формування, друк, додавання й знищення елементів оформити у вигляді функцій. Передбачити повідомлення про помилки при відкритті файлу й виконанні операцій вводу/виводу.

Структура "Людина":

- прізвище, ім'я, по батькові;
- рік народження;
- ріст;
- вага.

Знищити усі елементи із зазначеним ростом і вагою, додати елемент після елемента із зазначеним прізвищем.

#### 2. Код програми:

```
1 #include <iostream>
2 #include <cstring>
3 #include <cstdio>
4 using namespace std;
5
6 typedef struct
7 {
8     char name[50];
9     char surname[50];
10    char lastname[50];
11    char birthDate[11];
12    int height;
13    int weight;
14 } PERSON;
15
16 void readPersonData(PERSON* people, int len);
17 void printPersonData();
18 void deleteWithHeightAndWeight(int height, int weight);
19 void addAfterSurname(const PERSON& new_person, const char* target_surname);
20
21 int main()
22 {
23     int len = 2;
24     PERSON* people = new PERSON[len];
25     readPersonData(people, len);
26     deleteWithHeightAndWeight(170, 50);
27     std::cout << "After deleting" << std::endl << std::endl;
28     printPersonData();
29
30     std::cout << "After adding a new person" << std::endl << std::endl;
31     PERSON newPerson = {"NewSurname", "NewName", "NewLastname", "2000-01-01", 180, 70};
32     addAfterSurname(newPerson, "Kostak");
33     printPersonData();
34
35     delete[] people;
36     return 0;
37 }
38
39 void readPersonData(PERSON* people, int len)
40 {
41     FILE* fptr = fopen("people.bin", "wb");
42     if(fptr == NULL)
43     {
44         cerr << "Error opening file" << endl;
45         exit(1);
46     }
47     for(int i = 0; i < len; i++)
48     {
49         printf("Enter the information about %d person:\n", i + 1);
50         printf("Name: ");
51         fgets(people[i].name, sizeof(people[i].name), stdin);
52         people[i].name[strcspn(people[i].name, "\n")] = '\0';
53         printf("Surname: ");
54         fgets(people[i].surname, sizeof(people[i].surname), stdin);
55         people[i].surname[strcspn(people[i].surname, "\n")] = '\0';
56         printf("Last Name: ");
57         fgets(people[i].lastname, sizeof(people[i].lastname), stdin);
58         people[i].lastname[strcspn(people[i].lastname, "\n")] = '\0';
59         printf("Birth Date: ");
60         fgets(people[i].birthDate, sizeof(people[i].birthDate), stdin);
61         people[i].birthDate[strcspn(people[i].birthDate, "\n")] = '\0';
62         printf("Height: ");
63         scanf("%d", &people[i].height);
64         printf("Weight: ");
65         scanf("%d", &people[i].weight);
66         getchar(); // To consume the newline character left by scanf
67     }
68     fwrite(people, sizeof(PERSON), len, fptr);
69     fclose(fptr);
70 }
71
72 void printPersonData()
73 {
74     FILE* fptr = fopen("people.bin", "rb");
75     if(fptr == NULL)
76     {
77         cerr << "Error opening file" << endl;
78         exit(1);
79     }
80     PERSON person;
81     int i = 1;
82     while (fread(&person, sizeof(PERSON), 1, fptr))
83     {
84         printf("Information about %d person:\n", i++);
85         printf("Surname: %s\n", person.surname);
86         printf("Name: %s\n", person.name);
87         printf("Last Name: %s\n", person.lastname);
88         printf("Birth Date: %s\n", person.birthDate);
89         printf("Height: %i\n", person.height);
90         printf("Weight: %i\n", person.weight);
91     }
92 }
```

```

104     }
105     fclose(fptr);
106 }
107
108 void deleteWithHeightAndWeight(int height, int weight)
109 {
110     FILE* fptr = fopen("people.bin", "rb");
111     if (fptr == NULL)
112     {
113         cerr << "Error opening file" << endl;
114         exit(1);
115     }
116
117     FILE* fptr_temp = fopen("temp.bin", "wb");
118     if (fptr_temp == NULL)
119     {
120         cerr << "Error opening file" << endl;
121         fclose(fptr);
122         exit(1);
123     }
124
125     PERSON person;
126     int newlen = 0;
127     while (fread(&person, sizeof(PERSON), 1, fptr))
128     {
129         if (person.height != height || person.weight != weight)
130         {
131             fwrite(&person, sizeof(PERSON), 1, fptr_temp);
132             newlen++;
133         }
134     }
135     fclose(fptr);
136     fclose(fptr_temp);
137
138     if (remove("people.bin") != 0 || rename("temp.bin", "people.bin"))
139     {
140         std::cerr << "Error deleting objects" << std::endl;
141     }
142 }
143
144 void addAfterSurname(const PERSON& new_person, const char* target_surname)
145 {
146     FILE* fptr = fopen("people.bin", "rb");
147     if (fptr == NULL)
148     {
149         std::cerr << "Error opening file" << std::endl;
150         exit(1);
151     }
152
153     FILE* fptr_temp = fopen("temp.bin", "wb");
154     if (fptr_temp == NULL)
155     {
156         cerr << "Error opening file" << endl;
157         fclose(fptr);
158         exit(1);
159     }
160
161     PERSON person;
162     while (fread(&person, sizeof(PERSON), 1, fptr))
163     {
164         fwrite(&person, sizeof(PERSON), 1, fptr_temp);
165         if (strcmp(person.surname, target_surname) == 0)
166         {
167             fwrite(&new_person, sizeof(PERSON), 1, fptr_temp);
168         }
169     }
170
171     fclose(fptr);
172     fclose(fptr_temp);
173
174     if (remove("people.bin") != 0 || rename("temp.bin", "people.bin"))
175     {
176         std::cerr << "Error deleting objects" << std::endl;
177     }
178 }

```

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

```

Enter the information about 1 person:
Name: Olesia
Surname: Kostak
Last Name: Muh
Birth Date: 23-12-2006
Height: 171
Weight: 53
Enter the information about 2 person:
Name: Clone
Surname: Of
Last Name: Olesia
Birth Date: 23-12-2006
Height: 170
Weight: 50
After deliting

Information about 1 person:
Surname: Kostak
Name: Olesia
Last Name: Muh
Birth Date: 23-12-2006
Height: 171
Weight: 53
After adding a new person

Information about 1 person:
Surname: Kostak
Name: Olesia
Last Name: Muh
Birth Date: 23-12-2006
Height: 171
Weight: 53
Information about 2 person:
Surname: NewName
Name: NewSurname
Last Name: NewLastname
Birth Date: 2000-01-01
Height: 180
Weight: 70

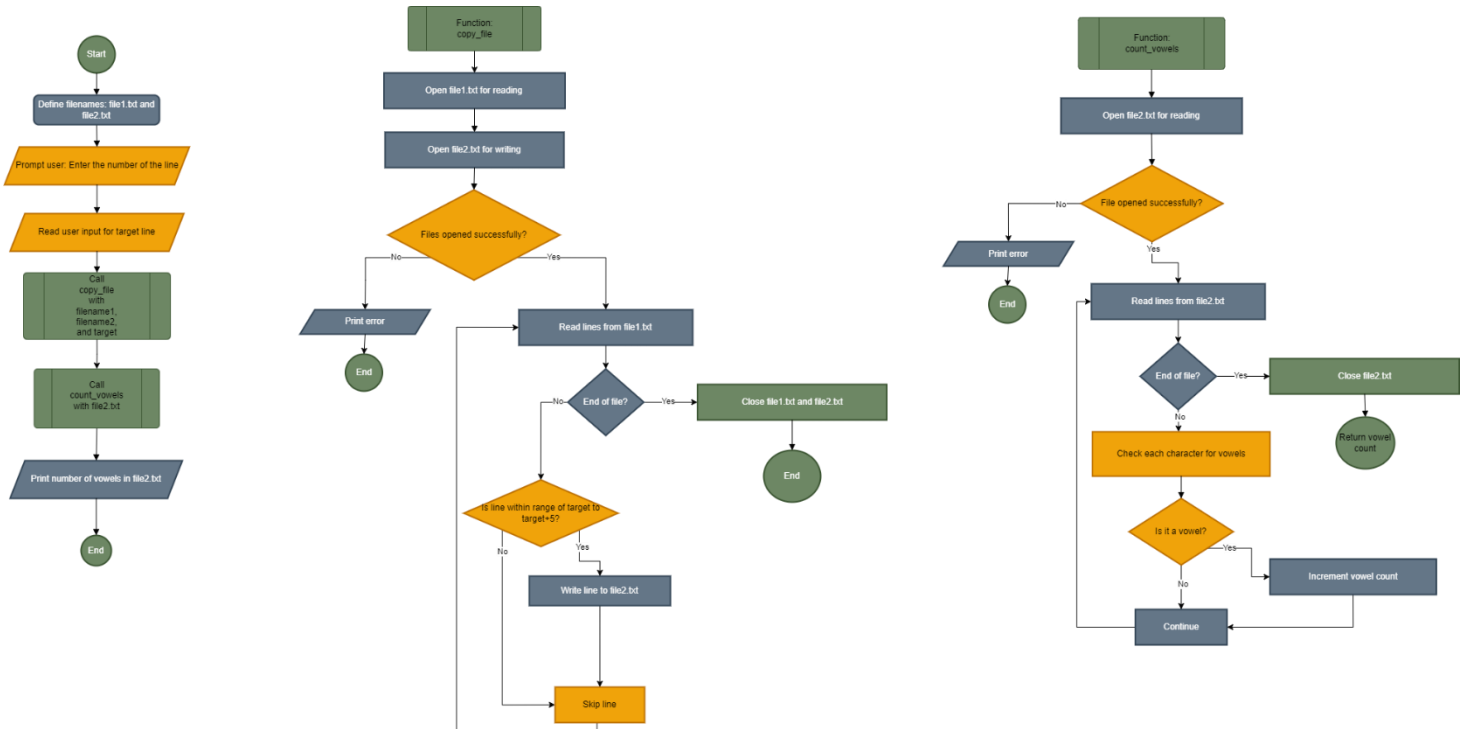
```

Затрачений час: 2 години

## Завдання VNS Lab 9 - Task 1

- Варіант завдання: 5
- Деталі завдання:
  - Скопіювати з файлу F1 у файл F2 рядки, починаючи з K до K+5.
  - Підрахувати кількість голосних букв у файлі F2.

### 1. Блоксхема:



### 2. Код програми:

```
1 #include <stdio.h>
2 #include <stdlib.h> //for exit
3
4 void copy_file(const char* from_file, const char* to_file, int target);
5 int count_vowels(const char* filename2);
6
7 int main()
8 {
9     const char* filename1 = "file1.txt";
10    const char* filename2 = "file2.txt";
11    printf("Enter the number of the line:");
12    int target;
13    scanf("%d", &target);
14    copy_file(filename1, filename2, 2);
15    printf("Number of vowel in file 2: %d", count_vowels(filename2));
16
17    return 0;
18 }
19
20 void copy_file(const char* from_file, const char* to_file, int target)
21 {
22     FILE* in = fopen(from_file, "r");
23     FILE* out = fopen(to_file, "w");
24
25     if (in == nullptr || out == nullptr)
26     {
27         perror("Error opening files");
28         exit(1);
29     }
30
31     int current_point = 1;
32     char buffer[256];
33     while (fgets(buffer, sizeof(buffer), in) != NULL)
34     {
35         if (current_point >= target && current_point <= target + 5)
36             fputs(buffer, out);
37         current_point++;
38     }
39     if (fclose(in) != 0 || fclose(out) != 0)
40     {
41         perror("Error opening files");
42         exit(1);
43     }
44 }
```

```
44 }
45
46 int count_vowels(const char* filename2)
47 {
48     FILE* fptr = fopen(filename2, "r");
49     if (fptr == NULL)
50     {
51         perror("Error opening file");
52         exit(1);
53     }
54
55     char buffer[256];
56     int counter = 0;
57     while (fgets(buffer, sizeof(buffer), fptr) != NULL)
58     {
59         for (int i = 0; buffer[i] != '\0'; i++)
60         {
61             char ch = buffer[i];
62             if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')
63                 counter++;
64         }
65     }
66     if (fclose(fptr) != 0)
67     {
68         perror("Error closing a file");
69         exit(1);
70     }
71     return counter;
72 }
```

### 3. Результат:

```
file1.txt
1 That
2 makes
3 sence
4 some
5 inf
6 and
7 this
8 text
9 must
10 be
11 in
12 english
13 because
14 ukrainian
15 doesn't
16 work
17 here

file2.txt
1 makes
2 sence
3 some
4 inf
5 and
6 this
7
```

```
Enter the number of the line: 2
Number of vowel in file 2: 9
```

## Завдання №4 Algotester Lab 4.1

- Варіант завдання: 1
- Деталі завдання (з використанням STL)

### 1. Код програми:

```
algotester_lab_4_task_olesia_kostak.cpp > ...
1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4
5
6 void print(const std::vector<int>& set, int len);
7
8
9 int main()
10 {
11     int N;
12     std::cin >> N;
13     std::vector<int> set_1(N);
14
15     for(int i = 0; i < N; i++)
16         std::cin >> set_1[i];
17
18     int M;
19     std::cin >> M;
20     std::vector<int> set_2(M);
21
22     for(int i = 0; i < M; i++)
23         std::cin >> set_2[i];
24
25
26     std::sort(set_1.begin(), set_1.end());
27     std::sort(set_2.begin(), set_2.end());
28
29     //N-N
30     std::vector<int> NDdif;
31     std::set_difference(set_1.begin(), set_1.end(), set_2.begin(), set_2.end(), std::back_inserter(NDdif));
32     print(NDdif, NDdif.size());
33     std::cout << std::endl;
34
35     //M-N
36     std::vector<int> MDdif;
37     std::set_difference(set_2.begin(), set_2.end(), set_1.begin(), set_1.end(), std::back_inserter(MDdif));
38     print(MDdif, MDdif.size());
39     std::cout << std::endl;
40
41     //Intersection
42     std::vector<int> NDinters;
43     std::set_intersection(set_1.begin(), set_1.end(), set_2.begin(), set_2.end(), std::back_inserter(NDinters));
44     print(NDinters, NDinters.size());
45     std::cout << std::endl;
46
47     //Union
48     std::vector<int> NDunion;
49     std::set_union(set_1.begin(), set_1.end(), set_2.begin(), set_2.end(), std::back_inserter(NDunion));
50     print(NDunion, NDunion.size());
51     std::cout << std::endl;
52
53     //Symetric difference
54     std::vector<int> NDSymdiff;
55     std::set_symmetric_difference(set_1.begin(), set_1.end(), set_2.begin(), set_2.end(), std::back_inserter(NDSymdiff));
56     print(NDSymdiff, NDSymdiff.size());
57     std::cout << std::endl;
58
59
60     return 0;
61
62
63
64 }
65
```

## 2. Результат:

a few seconds ago	C++ 23	Accepted	0.003	1.195	<a href="#">View</a>
-------------------	--------	----------	-------	-------	----------------------

## Завдання №5 Algotester Lab 4.2

- Варіант завдання: 2
- Деталі завдання (без STL)

## Код:

```
1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4
5 std::vector<int> merge(const std::vector<int>& left, const std::vector<int>& right);
6 std::vector<int> mergeSort(const std::vector<int>& vec);
7
8
9 int main()
10 {
11     int N, K;
12     std::cin >> N >> K;
13     std::vector<int> set(0);
14
15     for(int i = 0; i < N; i++)
16         std::cin >> set[i];
17
18     for(int i = 0; i < N; i++)
19     {
20         if(std::count(set.begin(), set.end(), set[i]) != 1)
21         {
22             set.erase(find(set.begin(), set.end(), set[i]));
23             N--;
24             i--;
25         }
26     }
27     set = mergeSort(set);
28
29     for(int i = 0; i < K; i++)
30     {
31         int index = 0;
32         int temp = set[index];
33         set.push_back(temp);
34         set.erase(find(set.begin(), set.end(), temp));
35     }
36
37     std::cout << set.size() << std::endl;
38     for(int i = 0; i < N; i++)
39         std::cout << set[i] << " ";
40
41     return 0;
42 }
43
44
45 std::vector<int> merge(const std::vector<int>& left, const std::vector<int>& right)
46 {
47     std::vector<int> result;
48     size_t i = 0, j = 0;
49
50     while (i < left.size() && j < right.size())
51     {
52         if (left[i] < right[j])
53
54         {
55             result.push_back(left[i]);
56             i++;
57         } else
58         {
59             result.push_back(right[j]);
60             j++;
61         }
62     }
63
64     while (i < left.size())
65     {
66         result.push_back(left[i]);
67         i++;
68     }
69     while (j < right.size())
70     {
71         result.push_back(right[j]);
72         j++;
73     }
74
75     return result;
76 }
77
78
79 std::vector<int> mergeSort(const std::vector<int>& vec)
80 {
81     if (vec.size() <= 1) return vec;
82
83     size_t mid = vec.size() / 2;
84     std::vector<int> left(vec.begin(), vec.begin() + mid);
85     std::vector<int> right(vec.begin() + mid, vec.end());
86
87     return merge(mergeSort(left), mergeSort(right));
88 }
```

## Результат:

a few seconds ago	C++ 23	Accepted	0.003	1.219	<a href="#">View</a>
8 days ago	C++ 23	Accepted	0.003	1.207	<a href="#">View</a>



## Завдання №6 Algotester Lab 6

- *Варіант завдання: 3*
- *Деталі завдання:*

Код:

```
g vns_lab_6_task_1_olesia_kosatk.cpp > main()
1  #include <stdio.h>
2  #include <string.h>
3  #include <iostream>
4  #include <cctype>
5
6  int main()
7  {
8      char myStr[256];
9      fgets(myStr, 256, stdin);
10     myStr[strcspn(myStr, "\n")] = '\0';
11     char nums_str[256] = "";
12     char res[256] = "";
13
14     char * token = strtok(myStr, " ");
15
16     while(token != NULL)
17     {
18         bool flag = true;
19
20         for(int i = 0; token[i] != '\0'; i++)
21             if(!isdigit(token[i]))
22                 flag = false;
23
24         if(flag)
25         {
26             strcat(nums_str, token);
27             strcat(nums_str, " ");
28         }
29         else
30         {
31             strcat(res, token);
32             strcat(res, " ");
33         }
34         token = strtok(NULL, " ");
35     }
36
37     strcat(res, nums_str);
38     std::cout << res;
39
40     return 0;
41 }
42
43
```

Результат:

a few seconds ago	C++ 23	Accepted	0.003	1.418	<a href="#">View</a>
-------------------	--------	----------	-------	-------	----------------------

## Завдання №7 Class Practice Work

- *Варіант завдання: -*
- *Деталі завдання:*

1. Реалізувати функцію створення файлу і запису в нього даних:

```
enum FileOpResult { Success, Failure, ... };
```

```
FileOpResult write_to_file(char *name, char *content);
```

2. Реалізувати функцію створення файлу і запису в нього даних:

```
enum FileOpResult { Success, Failure, ... };
```

```
FileOpResult copy_file(char *file_from, char *file_to);
```

Код:

```

1 #include <iostream>
2 #include <string.h>
3 #include <stdio.h>
4
5 enum FileOpResult {Success, Failure};
6 FileOpResult write_to_file(const char *name, char *content);
7 FileOpResult copy_file(const char *file_from, const char *file_to);
8 void print_data(const char* name);
9
10 int main()
11 {
12     const char* file_name1 = "filename1.txt";
13     const char* file_name2 = "filename2.txt";
14     int size_of_content = 256;
15     char content[size_of_content];
16     std::cout << "Enter a string to be written in a file: " << std::endl;
17     std::cin.getline(content, size_of_content);
18
19     if(write_to_file(file_name1, content) == Failure)
20         std::cout << "Failed to write content to file" << std::endl;
21     else
22         std::cout << "Content was successfully written!" << std::endl;
23
24     if(copy_file(file_name1, file_name2) == Failure)
25         std::cout << "Failed to copy content to file" << std::endl;
26     else
27         std::cout << "Content was successfully copied!" << std::endl;
28
29     return 0;
30 }
31
32 FileOpResult write_to_file(const char *name, char *content)
33 {
34     if(name == nullptr || strlen(name) == 0)
35         return Failure;
36
37     FILE* fptr = fopen(name, "w");
38     if(fptr == NULL)
39         return Failure;
40
41     fwrite(content, sizeof(char), strlen(content), fptr);
42
43     if(fclose(fptr) != 0)
44         return Failure;
45     return Success;
46 }
47
48 FileOpResult copy_file(const char *file_from, const char *file_to)
49 {
50     if(file_from == nullptr || strlen(file_from) == 0 || file_to == nullptr || strlen(file_to) == 0)
51         return Failure;
52
53

```

```

54     FILE* fptr1 = fopen(file_from, "r");
55     FILE* fptr2 = fopen(file_to, "w");
56
57     if (fptr1 == NULL || fptr2 == NULL)
58         return Failure;
59
60     char buffer[256];
61     while((fgets(buffer, sizeof(buffer), fptr1)) != NULL)
62     {
63         if(fputs(buffer, fptr2) == EOF)
64         {
65             fclose(fptr1);
66             fclose(fptr2);
67             return Failure;
68         }
69     }
70     if (fclose(fptr1) != 0 || fclose(fptr2) != 0)
71         return Failure;
72     return Success;
73 }
74
75 void print_data(const char* name)
76 {
77     FILE* fptr = fopen(name, "r");
78     if(fptr == NULL)
79     {
80         perror("Error opening file");
81         exit(1);
82     }
83     char ch;
84     while((ch = fgetc(fptr)) != EOF)
85         putchar(ch);
86     fclose(fptr);
87 }

```

## Результат:

```

Enter a string to be written in a file:
Hello world
Content was successfully written!
Content was successfully copied!

```

filename1.txt	filename2.txt
1 Hello world	1 Hello world

## Кооперація з командою:

Спільна борда в трелло по епіку:

The screenshot shows a Trello board titled 'Epic\_5' with a background image of a blue nebula. The board has four columns: 'To do', 'Doing', 'Done', and 'Committed'. The 'To do' column contains one card: 'Task 9 - Practice# programming: Self Practice Task'. The 'Doing' column is empty. The 'Done' column contains three cards: 'Nazarii Styk - Epic 5 Task 1 - Theory Education Activities', 'Nazarii Styk - Epic 5 Task 2 - Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-9)', and 'Nazarii Styk - Epic 5 Task 3 - Lab# programming: VNS Lab 6'. The 'Committed' column is empty. The board interface includes a top bar with search, filters, and user avatars, and a bottom bar with a 'Board' tab selected.

## **Висновки:**

По завершенню завдань Епіка №5, я освоїла роботу з файлами різних типів, зокрема текстовими та бінарними файлами, символами й рядковими змінними, а також методами стандартної бібліотеки для маніпулювання файлами. Розвинула навички створення й використання бібліотек для обробки даних із файлів, а також алгоритмічні вміння обробляти й аналізувати дані, збережені в різних форматах.