

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:

ВНС Лабораторної Роботи № 6

ВНС Лабораторної Роботи № 8

ВНС Лабораторної Роботи № 9

Алготестер Лабораторної Роботи №4

Алготестер Лабораторної Роботи №6

Практичних Робіт до блоку №5

Виконав:

Студент групи ІІІ-11

Орєньчук Юрій Миколайович

Львів 2024

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

Мета: Навчитися створювати та зчитувати файли у C++, створювати бінарні файли, працювати з символьними та рядковими змінними, дізнатися про стандартну бібліотеку та створити свої бібліотеки.

Теоретичні відомості:

Тема №1: Вступ до роботи з файлами

Тема №2: Символи і рядкові змінні

Тема №3: Створення та використання бібліотек

Індивідуальний план опрацювання теорії:

Тема №1: https://www.youtube.com/watch?v=EaHFhms_Shw

Тема №2: <https://www.youtube.com/watch?v=-TkoO8Z07hI>

Тема №3: <https://www.youtube.com/watch?v=Wt4dxDNmDA8>

Виконання роботи:

Завдання №1: VNS Lab 6 Variant 19

2. Постановка завдання

Задано рядок, що складається із символів. Символи поєднуються в слова. Слова одне від одного відокремлюються одним або декількома пробілами. Наприкінці тексту ставиться крапка. Текст містить не більше 255 символів. Виконати ввід рядка, використовуючи функцію gets(s) і здійснити обробку рядка у відповідності зі своїм варіантом.

19. Знищити з рядка всі слова, які не є ідентифікаторами.

Завдання №2: VNS Lab 8 Variant 19

2. Постановка 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.

19. Структура "Фільм":

- назва;
- режисер;
- країна;
- прибуток.

Знищити 2 елементи з кінця файлу, додати елемент після елемента із зазначеною назвою.

Завдання №3: VNS Lab 9 Variant 19

19.

- 1) Скопіювати з файлу F1 у файл F2 всі рядки, у яких немає слів, що збігаються з першим словом.
- 2) Визначити кількість приголосних букв у першому рядку файлу F2.

Завдання №4: Algotester Lab 4 Variant 1

Lab 4v1

Limits: 1 sec., 256 MiB

Вам дано 2 цілих чисел масиви, розміром N та M .

Ваше завдання вивести:

1. Різницю $N-M$
2. Різницю $M-N$
3. Їх перетин
4. Їх об'єднання
5. Їх симетричну різницю

Input

У першому рядку ціле число N - розмір масиву 1

У другому рядку N цілих чисел - елементи масиву 1

У третьому рядку ціле число M - розмір масиву 2

У четвертому рядку M цілих чисел - елементи масиву 2

Output

Вивести результат виконання 5 вищезазначених операцій у форматі:

У першому рядку ціле число N - розмір множини

У наступному рядку N цілих чисел - посортована у порядку зростання множина

Завдання №5: Algotester Lab 6 Variant 1

Lab 6v1

Limits: 2 sec., 256 MiB

Вам дано N слів та число K .

Ваше завдання перерахувати букви в словах, які зустрічаються в тексті більше-рівне ніж K разів (саме слово, не буква!).

Великі та маленькі букви вважаються однаковими, виводити необхідно малі, посортовані від останньої до першої у алфавіті. Букву потрібно виводити лише один раз.

У випадку якщо таких букв немає - вивести "Empty!".

Input

Цілі числа N та K - загальна кількість слів та мінімальна кількість слів щоб враховувати букви цього слова в результаті.

N стрічок s

Output

У першому рядку ціле число M - кількість унікальних букв

У другому рядку унікальні букви через пробіли

Завдання №6: Practice Work Task

Задача №2 – Копіювання вмісту файла у інший файл

Реалізувати функцію створення файла і запису в нього даних:

```
enum FileOpResult { Success, Failure, ... };  
FileOpResult copy_file(char *file_from, char *file_to);
```

Умови задачі:

- 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.

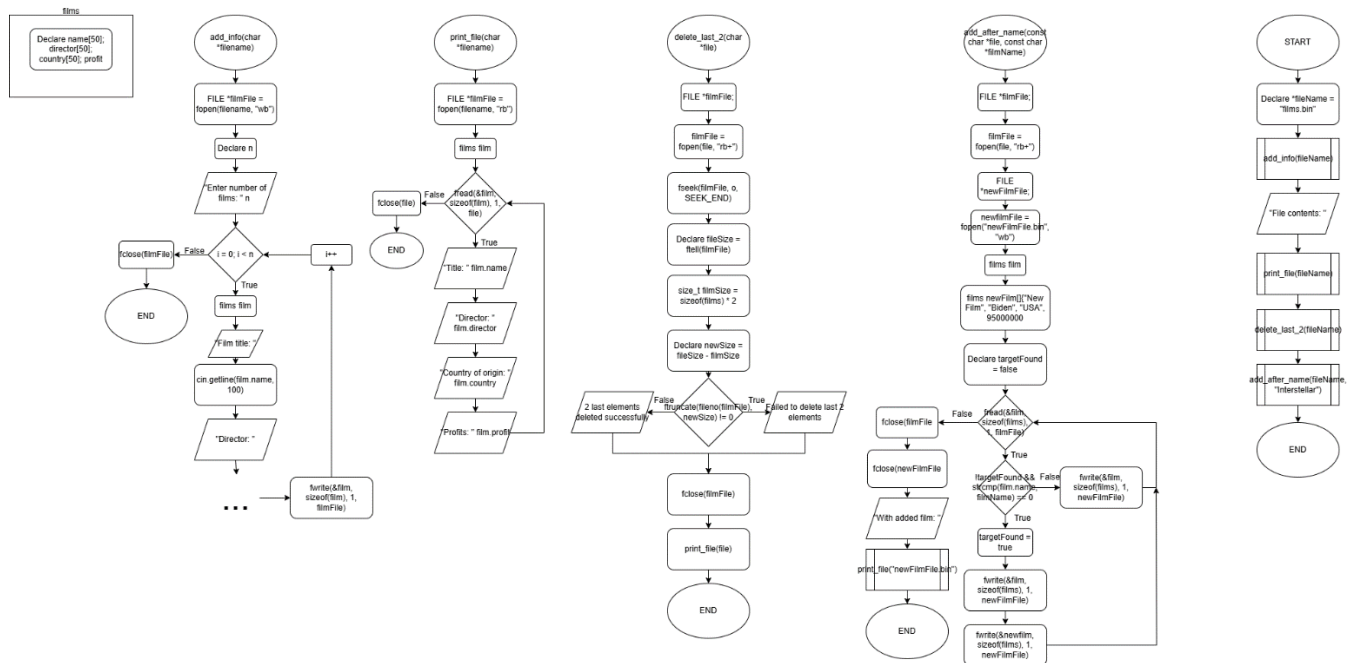
Дизайн та планована оцінка часу виконання завдань:

Завдання №1: VNS Lab 6 Variant 19

Планований час: 30 хв

Завдання №2: VNS Lab 8 Variant 19

Планований час: 1.5 год



Завдання №3: VNS Lab 9 Variant 19

Планований час: 1 год

Завдання №4: Algotester Lab 4 Variant 1

Планований час: 1.5 год

Завдання №5: Algotester Lab 6 Variant 1

Планований час: 1 год

Завдання №6: Practice Work Task

Планований час: 1 год

Код програм з посиланням на зовнішні ресурси:

Завдання №1: VNS Lab 6 Variant 19

```

1  #include <iostream>
2  #include <cstring>
3  #include <cctype>
4  using namespace std;
5
6  bool ident_check(char* word) {
7      if (word[0] == '\0') return false;
8
9      if (!isalpha(word[0]) && word[0] != '_') return 0;
10
11     for (int i = 1; word[i] != '\0'; i++) {
12         if (!isalnum(word[i]) && word[i] != '_')
13             return false;
14     }
15
16     return true;
17 }
18
19 int main() {
20     char str[256];
21     cout << endl << "Enter a sentence: \n";
22     gets(str);
23
24     char res[256] = "";
25     char* word = strtok(str, " ");
26
27     while (word != nullptr) {
28         if (ident_check(word)) {
29             strcat(res, word);
30             strcat(res, " ");
31         }
32         word = strtok(nullptr, " ");
33     }
34
35     cout << "Result: " << res;
36
37     return 0;

```

Завдання №2: VNS Lab 8 Variant 19

```

78     size_t filmSize = sizeof(films) * 2;
79
80     long newSize = fileSize - filmSize;
81     if (ftruncate(fileno(filmFile), newSize) != 0) {
82         cout << "Failed to delete last 2 elements\n";
83     }
84     else cout << "2 last elements deleted successfully\n\n";
85
86     fclose(filmFile);
87
88     print_file(file);
89 }
90
91 void add_after_name(const char *file, const char *filmName)
92 {
93     FILE *filmFile;
94     filmFile = fopen(file, "rb+");
95     FILE *newFilmFile;
96     newFilmFile = fopen("newFilmFile.bin", "wb");
97     if (filmFile == NULL) {
98         cout << "Failed to open a file";
99         exit(0);
100     }
101     if (newFilmFile == NULL) {
102         cout << "Failed to open new file";
103         exit(0);
104     }
105
106     films film;
107     films newFilm[] {
108         {"New Film", "Biden", "USA", 950000000}};
109
110     bool targetFound = false;
111     while (fread(&film, sizeof(films), 1, filmFile)) {
112         if (!targetFound && strcmp(film.name, filmName) == 0) {
113             targetFound = true;
114             fwrite(&film, sizeof(films), 1, newFilmFile);
115             fwrite(&newFilm, sizeof(films), 1, newFilmFile);
116         } else {
117             fwrite(&film, sizeof(films), 1, newFilmFile);
118         }
119     }

```



```

120
121     fclose(filmFile);
122     fclose(newFilmFile);
123     cout << "\nWith added film: \n\n";
124     print_file("newFilmFile.bin");
125 }
126
127 int main(){
128     char *fileName = "films.bin";
129     add_info(fileName);
130
131     cout << "\nFile contents:\n";
132     print_file(fileName);
133
134     delete_last_2(fileName);
135
136     add_after_name(fileName, "Interstellar");
137
138     return 0;
139 }

```

Завдання №3: VNS Lab 9 Variant 19

```

43     if (getline(readFile, line)) {
44         istringstream iss(line);
45         if (iss >> firstWord) {
46             for (char c : line) {
47                 if (consonant_check(c)) {
48                     consonantCount++;
49                 }
50             }
51         }
52         writeFile << line << endl;
53     }
54
55     while (getline(readFile, line)) {
56         if (line.empty()) continue;
57
58         if (line.find(firstWord) != string::npos) {
59             continue;
60         }
61
62         writeFile << line << endl;
63     }
64
65     writeFile << "Amount of consonants in 1st line: " << consonantCount << endl;
66
67     readFile.close();
68     writeFile.close();
69
70     return 0;
71 }

```

Завдання №4: Algotester Lab 4 Variant 1

3 STL:

```
1  #include <iostream>
2  #include <set>
3  #include <vector>
4  #include <algorithm>
5  using namespace std;
6
7  int main() {
8      int N, M;
9
10     cin >> N;
11     vector<int> n(N);
12     for (int i = 0; i < N; i++) {
13         cin >> n[i];
14     }
15
16     cin >> M;
17     vector<int> m(M);
18     for (int i = 0; i < M; i++) {
19         cin >> m[i];
20     }
21
22     sort(n.begin(), n.end());
23     sort(m.begin(), m.end());
24
25     vector<int> diff_N_M;
26     set_difference(n.begin(), n.end(), m.begin(), m.end(), back_inserter(diff_N_M));
27     cout << diff_N_M.size() << endl;
28     for (size_t i = 0; i < diff_N_M.size(); i++) {
29         if (i > 0) cout << " ";
30         cout << diff_N_M[i];
31     }
32     cout << endl;
33
34     vector<int> diff_M_N;
35     set_difference(m.begin(), m.end(), n.begin(), n.end(), back_inserter(diff_M_N));
36     cout << diff_M_N.size() << endl;
37     for (size_t i = 0; i < diff_M_N.size(); i++) {
38         if (i > 0) cout << " ";
39         cout << diff_M_N[i];
40     }
41     cout << endl;
42 }
```

```

43     vector<int> intersec;
44     set_intersection(n.begin(), n.end(), m.begin(), m.end(), back_inserter(intersec));
45     cout << intersec.size() << endl;
46     for (size_t i = 0; i < intersec.size(); i++) {
47         if (i > 0) cout << " ";
48         cout << intersec[i];
49     }
50     cout << endl;
51
52     vector<int> uni;
53     set_union(n.begin(), n.end(), m.begin(), m.end(), back_inserter(uni));
54     cout << uni.size() << endl;
55     for (size_t i = 0; i < uni.size(); i++) {
56         if (i > 0) cout << " ";
57         cout << uni[i];
58     }
59     cout << endl;
60
61     vector<int> diff_simm;
62     set_symmetric_difference(n.begin(), n.end(), m.begin(), m.end(), back_inserter(diff_simm));
63     cout << diff_simm.size() << endl;
64     for (size_t i = 0; i < diff_simm.size(); i++) {
65         if (i > 0) cout << " ";
66         cout << diff_simm[i];
67     }
68     cout << endl;
69
70     return 0;
71 }

```

Без STL:

```

75     vector<int> SymmDiff(vector<int> n, vector<int> m) {
76         vector<int> res1 = Diff(n, m);
77         vector<int> res2 = Diff(m, n);
78         return Union(res1, res2);
79     }
80
81     int main() {
82         int N, M;
83
84         cin >> N;
85         vector<int> n(N);
86         for (int i = 0; i < N; i++) {
87             cin >> n[i];
88         }
89
90         cin >> M;
91         vector<int> m(M);
92         for (int i = 0; i < M; i++) {
93             cin >> m[i];
94         }
95
96         sort(n.begin(), n.end());
97         sort(m.begin(), m.end());
98
99         vector<int> diff1 = Diff(n, m);
100        Print(diff1);
101        vector<int> diff2 = Diff(m, n);
102        Print(diff2);
103        vector<int> intersection = Intersect(n, m);
104        Print(intersection);
105        vector<int> union_set = Union(n, m);
106        Print(union_set);
107        vector<int> sym_diff = SymmDiff(n, m);
108        Print(sym_diff);
109
110        return 0;
111    }

```

Завдання №5: Algotester Lab 6 Variant 1

```

43
44         cout << sorted.size() << endl;
45         for (size_t i = 0; i < sorted.size(); i++) {
46             cout << sorted[i] << " ";
47         }
48         cout << endl;
49
50         return 0;
51     }

```

Завдання №6: Practice Work Task

Завдання 1:

```
36  int main(){
37      char name[200];
38      char content[500];
39
40      cout << "Enter the file name: ";
41      cin.getline(name, 1000);
42
43      cout << "Enter what you want to write into the file: ";
44      cin.getline(content, 1000);
45
46      FileOpResult res = write_to_file(name, content);
47
48      if (res == Success){
49          cout << "Operation is successfull!";
50      } else cout << "Operation Failed...";
51
52      return 0;
53  }
```

З
а
в
д
а
н
н
я

2
:

```

1  #include <iostream>
2  #include <fstream>
3  #include <string>
4  using namespace std;
5
6  enum FileOpResult { Success, Failure };
7
8  FileOpResult copy_file(char *file_from, char *file_to){
9      ifstream old_file;
10     ofstream new_file;
11     old_file.open(file_from, ios::in);
12     new_file.open(file_to, ios::out);
13
14     if (old_file.is_open()){
15         if (new_file.is_open()){
16
17             new_file << old_file.rdbuf();
18             if (new_file.fail() || old_file.fail()) {
19                 cout << "Failed to copy over file elements";
20                 return Failure;
21             }
22
23             old_file.close();
24             new_file.close();
25             if (new_file.fail() || old_file.fail()) {
26                 cout << "Failed to close files";
27                 return Failure;
28             }
29
30             return Success;
31         } else {
32             cout << "Failed to open new file";
33             return Failure;
34         }
35     } else {
36         cout << "Failed to open old file";
37         return Failure;
38     }
39 }
40
41 int main(){
42     char file_from[100];
43     char file_to[100];
44
45     cout << "Enter the file you want to copy: ";
46     cin.getline(file_from, 100);
47
48     cout << "Enter the name of the new file: ";
49     cin.getline(file_to, 100);
50
51     FileOpResult result = copy_file(file_from, file_to);
52
53     if (result == Success) {
54         cout << "Successfully copied file\n";
55     }
56     else {
57         cout << "Failed to copy file\n";
58     }
59     return 0;
60 }

```

Результат виконання завдань, тестування та фактично затрачених час:

Завдання №1: VNS Lab 6 Variant 19

Фактично затрачених час: 45 хв

```
Enter a sentence:  
flood joke 0full stack_overflow gill^fool  
Result: flood joke stack_overflow  
PS C:\Users\admin1\Documents\CPP> █
```

Завдання №2: VNS Lab 8 Variant 19

Фактично затрачених час: 2 год

```
Title: Film1  
Director: hah  
Country of origin: Ukraine  
Profits: 100  
Title: Interstellar  
Director: idk  
Country of origin: USA  
Profits: 900000000  
Title: Film3  
Director: keys  
Country of origin: Uganda  
Profits: 200  
  
With added film:  
  
Title: Film1  
Director: hah  
Country of origin: Ukraine  
Profits: 100  
Title: Interstellar  
Director: idk  
Country of origin: USA  
Profits: 900000000  
Title: New Film  
Director: Biden  
Country of origin: USA  
Profits: 950000000  
Title: Film3  
Director: keys  
Country of origin: Uganda  
Profits: 200  
PS C:\Users\admin1\Documents\CPP> █
```

Завдання №3: VNS Lab 9 Variant 19

Фактично затрачений час: 1 год

```
Enter the contents for source.txt (type 'END' to finish):
fish
indiana
op
ghoul
jolly
fish
banner
house fish catch
vertigo
END
PS C:\Users\admin1\Documents\CPP> 
```

```
≡ final.txt
1  fish
2  indiana
3  op
4  ghoul
5  jolly
6  banner
7  vertigo
8  Amount of consonants in 1st line: 3
9
```

Завдання №4: Algotester Lab 4 Variant 1

Фактично затрачений час: 1.5 год

6 hours ago	Lab 4v1 - Lab 4v1	C++ 23	Accepted	0.003	1.281	1935950
-------------	-------------------	--------	----------	-------	-------	---------

Завдання №5: Algotester Lab 6 Variant 1

Фактично затрачений час: 45 хв

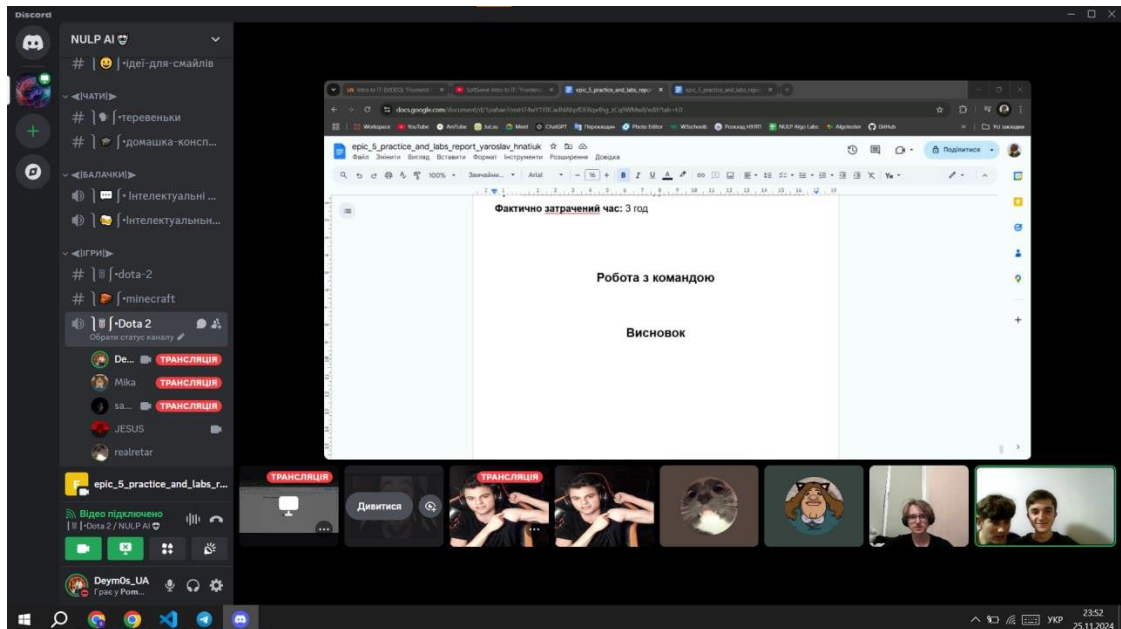
6 minutes ago	Lab 6v1 - Lab 6v1	C++ 23	Accepted	0.043	6.484	1936059
---------------	-------------------	--------	----------	-------	-------	---------

Завдання №6: Practice Work Task

Фактично затрачений час: 1 год

```
epics > epic_5 > ≡ new.txt
1  Hello Polytech
```

Робота з командою



Висновки:

Я навчився створювати та зчитувати файли у C++, створювати бінарні файли, працювати з символьними та рядковими змінними, дізнався про стандартну бібліотеку та створив свої бібліотеки.

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/595