

Центральноукраїнський національний технічний університет  
Кафедра програмування комп'ютерних систем і мереж  
Дисципліна: Базові методології та технології програмування

## Звіт

з лабораторних робіт №21

Теми: Технології програмування роботи з файлами

Мета: Отримати знання та навички, необхідні для освоєння технології роботи з даними на основі використання файлів за допомогою засобів мови C++.

Виконав: ст. гр. КМ-20

Калиновський В.М.

Перевірила: доцент кафедри ПКСМ

Рибакова Л.В.

Варіант 3

Дата виконання: 22 травня 2021

Дата оформлення: 22 травня 2021

**Завдання 1.** Взяти за основу лабораторну роботу №18, змінити код програмування так, що:

1. Вхідні дані вводилися не з клавіатури, а з файла
2. Результати виконання виводились і на екран, і у файл

**Завдання 2.** Взяти за основу лабораторну роботу №19 (завдання 1), змінити код програмування так, що:

1. Вхідні дані (поля структури) вводилися з клавіатури і після введення записувалися у файл (окрема функція)
2. Програма мала можливість дописувати дані у файл (окрема функція) і виводити їх на екран
3. Дані з файла виводились на екран (окрема функція)

**Робота.** Кожне із завдань являє собою два класи. Перший клас було перенесено із класа Eight лабораторної роботи Semester, яке являє собою вирішення 18 лабораторної роботи. У конструкторі із завдання 1 було зроблено наступні зміни:

1. Введення строки замінено на зчитування з одного файла
2. Дописано у кінці конструктора запис нової строки до другого файла

**Було:**

```
cout << "Введите строку: ";  
cin.getline(userSTR, 140);
```

**Стало:**

```
while (!feof(file1))  
    if (fgets(userSTR, 140, file1) != NULL )  
        fputs(userSTR, stdout);
```

**Було:**

```
cout << "\n\nНовое предложение без слов с удвоенными буквами:\n\" "  
    << userSTRnew << "\"";
```

**Стало:**

```
cout << "\n\nНовое предложение без слов с удвоенными буквами:\n\" "  
    << userSTRnew << "\"";  
FILE *file2 = fopen("Result.txt", "w");  
fputs(userSTRnew, file2);  
fclose(file2);
```

**Що очікується від функції?** Аналогічно із завданням лабораторної роботи №18 за винятком того, що програма зчитує дані з одного файла та записує до іншого.

Другий клас було створено на основі завдання 19 лабораторної. Спочатку воно не було класом, але щоб ця лабораторна була гарно написаною, а перше завдання зроблено класом, то й друге завдання перероблено під клас. Для конструктора із завдання 2 написана загальна структура функції так:

Крок 1: Спитати у користувача, чи буде він заносити інформацію (продовжити 2.1.1), зчитати (продовжити 2.2.1), чи завершити (продовжити 3);

Крок 2.1.1: Зчитати файл (навіть якщо він пустий);

Крок 2.1.2: Знайти останню структуру

Крок 2.1.3: Дописати інформацію;

Крок 2.1.4: Зберегти інформацію;

Крок 2.1.5: Повернутися до кроку 1;

Крок 2.2.1: Зчитати інформацію;

Крок 2.2.2: Спитати у користувача, яку інформацію вивести;

Крок 2.2.3: Вивести задану інформацію;

Крок 2.2.4: Повернутися до кроку 1;

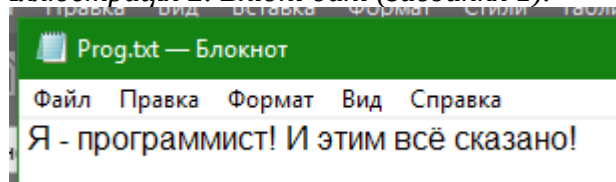
Крок 3: Завершити програму.

*Що очікується від функції?* Аналогічно із завданням 19, за винятком того, що вона може зчитувати інформацію з файлу і зберігати інформацію у файлі.

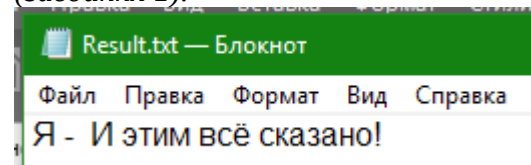
*Код функції буде наведено у повному коді у класі Two.*

*Що очікується від програми?* Програма поєднує попередні завдання і надає нових можливостей зчитуванні з файлу та запису у файл.

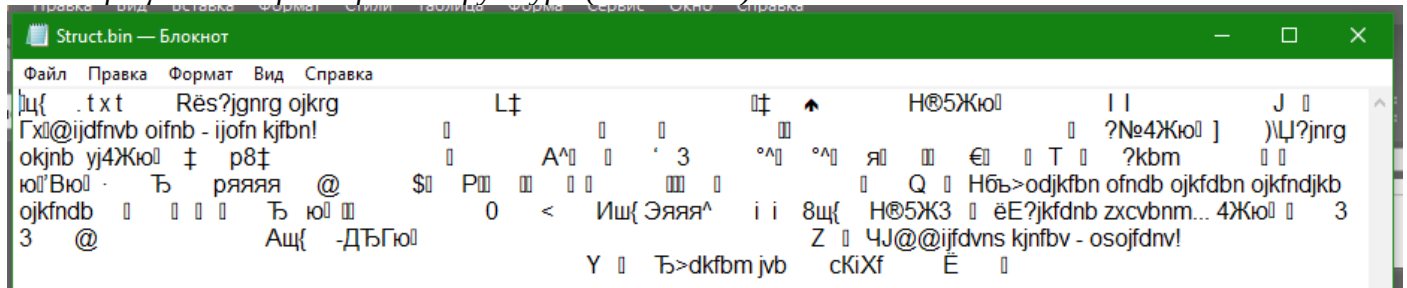
Ілюстрація 2. Вхідні дані (завдання 1).



Ілюстрація 1. Вихідні дані (завдання 1).



Ілюстрація 3. Бінарний файл структури (завдання 2).



Ілюстрація 4. Файлова структура програми.

Имени	Дата изменения	Тип	Размера
main.o	22.05.2021 23:30	Файл "О"	195 КБ
Prog.txt	09.05.2021 19:15	Текстовый докум...	1 КБ
Result.txt	22.05.2021 23:39	Текстовый докум...	1 КБ
Sem2Zad8.exe	22.05.2021 23:30	Приложение	3 233 КБ
Struct.bin	22.05.2021 23:40	UltraISO Файл	2 КБ

*Зауваження:* Тратити час на збереження та читання інформації структури у читаємому стані у файлі я не хотів, тому вирішив зберігати інформацію у бінарному файлі.

### Ілюстрація 5. Результат виконання програми (завдання 1-2)

```
Q:\Projects\build-Sem2Zad8-Desktop_x86_windows_msys_pe_64bit-Debug\debug\Sem2Zad8.exe
Я - программист! И этим всё сказано!
Количество слов в вашем тексте: 6
Количество написанных символов: 36

Новое предложение без слов с удвоенными буквами:
" Я - И этим всё сказано! "

Начать заполнение таблицы (2 - вывести то, что есть; 1 - заполнить; 0 - ничего не делать)? 1
№8
Введите длительность показа: 0.25
Введите название программы: dkfbm
Введите рейтинг программы: 48
Продолжить (1 - да, 0 - нет)? 0

Начать заполнение таблицы (2 - вывести то, что есть; 1 - заполнить; 0 - ничего не делать)? 1
№9
Введите длительность показа: 2.13
Введите название программы: eijng kjdfnb ijfdbn iuojfdbn
Введите рейтинг программы: 81
Продолжить (1 - да, 0 - нет)? 0

Начать заполнение таблицы (2 - вывести то, что есть; 1 - заполнить; 0 - ничего не делать)? 2
Какую передачу вывести на экран?
5

№5          0.49          odjkfbn ofndb ojkfdbn ojkfndjkb ojkfdbn          51

Начать заполнение таблицы (2 - вывести то, что есть; 1 - заполнить; 0 - ничего не делать)? 2
Какую передачу вывести на экран?
8

№8          0.25          dkfbm          48

Начать заполнение таблицы (2 - вывести то, что есть; 1 - заполнить; 0 - ничего не делать)?
2
Какую передачу вывести на экран?
1

№1          1.49          jgnrg ojkrg          74

Начать заполнение таблицы (2 - вывести то, что есть; 1 - заполнить; 0 - ничего не делать)? 0
—
```

**Результат:** як бачимо, програма зчитала інформацію з першого файлу, обробила її, записала результат до другого файлу та коректно працює зі структурою і бінарним файлом. Інформація зберігається та зчитується. *Вважаємо завдання виконаним!* ✓

Далі наведено повний код програми:

```
#include <windows.h>
#include <string>
#include <iostream>
using namespace std;

class One{
public:
    One()
    {
        //cout << "Введіть строку: ";
        //cin.getline(userSTR, 140);
        FILE *file1 = fopen("Prog.txt", "r");
        while (!feof(file1))                                //Пока не кінець файлу
```

```

        if (fgets(userSTR , 140 , file1) ≠ NULL ) //Считать символы из файла
            fputs(userSTR, stdout);                //Вывод на экран
fclose(file1);
countWord();
WordSuggestion();
checkDubbling();
}
private:
string word;
char userSTR[140]{{}, userSTRnew[140]{{}, *userSTRword{}};
int countSymbol = 0, countWORD = 1;
bool check{};
void countWord()
{
    for (unsigned long int i = 0; i < sizeof(userSTR) / sizeof(char); i++)
        if (userSTR[i] == ' ' && userSTR[i - 1] ≠ '-')
            countWORD++;
    cout << "\nКоличество слов в вашем тексте: " << countWORD;
}
void WordSuggestion()
{
    for (char i : userSTR)
        if (i ≠ '\0')
            countSymbol++;
    cout << "\nКоличество написанных символов: " << countSymbol;
}
void checkDubbling()
{
    userSTRword = strtok(userSTR, " ");
    while (userSTRword ≠ nullptr)
    {
        word = userSTRword;
        for (unsigned long int j = 0; j < word.size(); j++)
        {
            if (userSTRword[j] == userSTRword[j + 1]
                && userSTRword[j + 1] ≠ '.')
            {
                check = false;
                break;
            }
            else
                check = true;
        }
        if (check)
            strcat(userSTRnew, userSTRword);
        strcat(userSTRnew, " ");
        userSTRword = strtok(nullptr, " ");
    }
    strcat(userSTRnew, "\0");
    cout << "\n\nНовое предложение без слов с удвоенными буквами:\n\" "
        << userSTRnew << "\"";
    FILE *file2 = fopen("Result.txt", "w");
    fputs(userSTRnew, file2);
    fclose(file2);
}

```

```

    }
};

struct TV
{
    float timeVision;
    char nameVision[150];
    int rate, num;
};

class Two{
public:
    Two()
    {
        TV tv[longTV];
        for(;;){
            cout << "\n\nНачать заполнение таблицы (2 - вывести то, что есть; "
                    "1 - заполнить; 0 - ничего не делать)? ";
            cin >> ctn;
            switch(ctn)
            {
                case 2:
                    load(tv);
                    output(tv);
                    break;
                case 1:
                    load(tv);
                    input(tv);
                    save(tv);
                    break;
            }
            if (ctn == 0) break;
        }
    }
private:
    const int longTV = 10;
    bool continueTV = true;
    int ctnTV, ctn, numvis, ptr;
    int NumStruct(struct TV *tv, int count)
    {
        if(tv[count - 1].num == count) return NumStruct(tv, count + 1);
        else return count - 1;
    }
    void input(struct TV *tv)
    {
        ptr = NumStruct(tv, 1);
        for (int i = ptr; i < longTV; i++)
        {
            tv[i].num = i + 1;
            cout << "\n№" << tv[i].num;
            cout << "\nВведите длительность показа: ";
            cin >> tv[i].timeVision;
            cout << "Введите название программы: ";
            fflush(stdin);
        }
    }
};

```

```

        cin.getline(tv[i].nameVision, 150);
        cout << "Введіть рейтинг програми: ";
        cin >> tv[i].rate;
        cout << "Продолжити (1 - да, 0 - нет)? ";
        cin >> ctnTV;
        if (ctnTV == 0) continueTV = false;
        else if (ctnTV == 1) continueTV = true;
        else cout << "Ошибка ввода";
        if (!continueTV) break;
    }
}

void save(struct TV *tv)
{
    FILE *file = fopen("Struct.bin", "wb");
    fflush(file);
    fwrite(&tv, sizeof(tv), 153, file);
    fclose(file);
}

void load(struct TV *tv)
{
    FILE *file = fopen("Struct.bin", "rb");
    fread(&tv, sizeof(tv), 153, file);
    fclose(file);
}

//NOTE: 153 элемента структуры = 2 целочисленные + 1 с плавающей точкой + 1
символ * 150
void output(struct TV *tv)
{
    fflush(stdin);
    cout << "Какую передачу вывести на экран?\n";
    cin >> numvis;
    cout << "\n№" << tv[numvis - 1].num << "\t\t"
        << tv[numvis - 1].timeVision << "\t\t"
        << tv[numvis - 1].nameVision << "\t\t"
        << tv[numvis - 1].rate << "\n";
}

};

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    One one;
    Two two;
    cin.get();
    cin.get();
    return 0;
}

```

**Висновок:** програма зчитує інформацію з файлу і записує її у файл. Ми отримали знання та навички, необхідні для роботи з даними на основі використання файлів.