Міністерство освіти і науки України

Вінницький національний технічний університет

Факультет комп’ютерних систем та автоматики

Кафедра КСУ

Лабораторна робота №3

з дисципліни: «Компютерні технології та програмування»

Тема: «Робота з файлами»

Виконав

студент групи 2АКІТ-20б

Н. Бондар

Перевірив

к.т.н., доц.кафедри КСУ

Севастьянов В. М.

м.Вінниця, ВНТУ 2021

**Лабораторна робота №3**

**Тема: «Робота з файлами»**

**Варіант 1.**

**Завдання :**

1. Створити програму на мові С згідно варіанту використавши середовище програмування Dev-C++ 4.0: для збереження даних про планшетні сканери описати структуру наступного вигляду:

struct scan\_іnfo{

char model [25]; //найменування моделі

іnt prіce; // ціна

double x\_sіze;

області сканування // горизонтальний розмір

double y\_sіze;

області сканування // вертикальний розмір

іnt optr; // оптичний дозвіл

іnt grey; // число градацій сірого

};

структура файлу: спочатку у файлі розміщається значення типу іnt, що визначає кількість зроблених у файлі записів; далі в кожній новій стрічці розміщуються записи про сканери.

2. Відкомпілювати та відлагодити програму.

3. Розробити набір тестів і перевірити роботу програми на них.

4. Відповісти на контрольні запитання.

5. Зробити висновки.

6. Звіт по лабораторній роботі має складатися з титульної сторінки, лістингів програм, висновків по роботі.

Варіант 1.

1. Написати функцію, що записує у текстовий файл дані про сканер із приведеної структури.

2. Написати функцію, що витягає з цього файлу дані про сканер у структуру типу scan\_іnfo. Обов'язковий параметр - номер необхідного запису. Функція повинна повертати нульове значення, якщо читання пройшло успішно, і -1 у протилежному випадку.

3. Привести приклад програми, що створює файл із даними про сканери, (дані вводяться з клавіатури) - 6-8 записів, і виводить на дисплей дані про запитуваний запис.

4. Усі необхідні дані для функцій повинні передаватися їм як параметри. Використання глобальних змінних у функціях не допускається.

**Хід роботи :**

**1. Ретельно вивчаю поставлене передімною завдання.**

**2. Створюю програму на мові С згідно варіанту використавши середовище програмування Dev-C++ 4.0: для збереження даних про планшетні сканери описую структуру наступного вигляду:**

struct scan\_іnfo{

char model [25]; //найменування моделі

іnt prіce; // ціна

double x\_sіze;

області сканування // горизонтальний розмір

double y\_sіze;

області сканування // вертикальний розмір

іnt optr; // оптичний дозвіл

іnt grey; // число градацій сірого

};

**структура файлу: спочатку у файлі розміщається значення типу іnt, що визначає кількість зроблених у файлі записів; далі в кожній новій стрічці розміщуються записи про сканери.**

**3. Відкомпільовую та відлагоджую програму.**

**4. Розробляю набір тестів і перевіряю роботу програми на них.**

**5. Відповідаю на контрольні запитання.**

**6. Роблю висновки.**

**Варіант 1.**

**1. Пишу функцію, що записує у текстовий файл дані про сканер із приведеної структури.**

void write\_struct(scan\_info scaners[]) {

FILE \* file;

char file\_name[20];

std::cout << "Enter file name : ";

std::cin >> file\_name;

file = fopen(file\_name, "wb");

/\*

char model [25]; //найменування моделі

int price; // ціна

double x\_size; // області сканування // горизонтальний розмір

double y\_size; // області сканування // вертикальний розмір

int optr; // оптичний дозвіл

int grey; // число градацій сірого

\*/

for (int i = 0; i < scaners\_number; i++) {

fprintf(file, "%s,", scaners[i].model);

fprintf(file, "%i,", scaners[i].price);

fprintf(file, "%lf,", scaners[i].x\_size);

fprintf(file, "%lf,", scaners[i].y\_size);

fprintf(file, "%i,", scaners[i].optr);

fprintf(file, "%i;", scaners[i].grey);

}

fclose(file);

}

**2. Пишу функцію, що витягає з цього файлу дані про сканер у структуру типу scan\_іnfo. Обов'язковий параметр - номер необхідного запису. Функція повертає нульове значення, якщо читання пройшло успішно, і -1 у протилежному випадку.**

int read\_struct(scan\_info scaners[]) {

FILE \* file;

char file\_name[20];

std::cout << "Enter file name : ";

std::cin >> file\_name;

file = fopen(file\_name, "rb");

if (file == NULL) {

std::cout << "Can not open file !!!\n";

return -1;

}

else {

std::cout << "Data was read seccesfully !!!\n";

}

for (int i = 0; i < scaners\_number; i++) {

for (int j = 0; j < 25; j++) {

fscanf(file, "%c", &scaners[i].model[j]);

if (scaners[i].model[j] == ',') {

scaners[i].model[j] = '\0';

break;

}

}

fscanf(file, "%i", &scaners[i].price);

fscanf(file, "%c");

fscanf(file, "%lf", &scaners[i].x\_size);

fscanf(file, "%c");

fscanf(file, "%lf", &scaners[i].y\_size);

fscanf(file, "%c");

fscanf(file, "%i", &scaners[i].optr);

fscanf(file, "%c");

fscanf(file, "%i", &scaners[i].grey);

fscanf(file, "%c");

}

fclose(file);

return 0;

}

**3. Приводжу приклад програми, що створює файл із даними про сканери, (дані вводяться з клавіатури) - 6-8 записів, і виводить на дисплей дані про запитуваний запис.**

**4. Усі необхідні дані для функцій передаю їм як параметри. Не використовую глобальних змінних.**

**Лістинг коду :**

#include<iostream>

/\*

Варіант 1.

1. Написати функцію, що записує у текстовий файл дані про сканер із приведеної

структури.

2. Написати функцію, що витягає з цього файлу дані про сканер у структуру типу

scan\_іnfo. Обов'язковий параметр - номер необхідного запису. Функція повинна повертати нульове

значення, якщо читання пройшло успішно, і -1 у протилежному випадку.

3. Привести приклад програми, що створює файл із даними про сканери, (дані вводяться з

клавіатури) - 6-8 записів, і виводить на дисплей дані про запитуваний запис.

4. Усі необхідні дані для функцій повинні передаватися їм як параметри. Використання

глобальних змінних у функціях не допускається.

\*/

#define scaners\_number 3

struct scan\_info{

char model[25]; //найменування моделі

int price; // ціна

double x\_size; // області сканування // горизонтальний розмір

double y\_size; // області сканування // вертикальний розмір

int optr; // оптичний дозвіл

int grey; // число градацій сірого

};

void fill\_struct(scan\_info scaners[]);

void show\_struct(scan\_info scaners[]);

void write\_struct(scan\_info scaners[]);

int read\_struct(scan\_info scaners[]);

int main() {

scan\_info scaners[scaners\_number];

int user\_choice;

while (true) {

std::cout << "Menu :\n 0.Exit\n 1.Fill struct\n 2.Show struct\n 3.Write struct\n 4.Read struct\n>>> ";

std::cin >> user\_choice;

if (user\_choice == 0) {

std::cout << "Good buy\n";

break;

}

else if (user\_choice == 1) {

fill\_struct(scaners);

}

else if (user\_choice == 2) {

show\_struct(scaners);

}

else if (user\_choice == 3) {

write\_struct(scaners);

}

else if (user\_choice == 4) {

read\_struct(scaners);

}

else {

std::cout << "Wrong number !!!\n";

}

}

return 0;

}

void fill\_struct(scan\_info scaners[]) {

for (int i = 0; i < scaners\_number; i++) {

std::cout << "Enter scaner number " << i + 1 << " parameters :\n";

std::cout << "Model : ";

std::cin >> scaners[i].model;

std::cout << "Price : ";

std::cin >> scaners[i].price;

std::cout << "X\_size : ";

std::cin >> scaners[i].x\_size;

std::cout << "Y\_size : ";

std::cin >> scaners[i].y\_size;

std::cout << "Optr : ";

std::cin >> scaners[i].optr;

std::cout << "Grey : ";

std::cin >> scaners[i].grey;

std::cout << '\n';

}

}

void show\_struct(scan\_info scaners[]) {

for (int i = 0; i < scaners\_number; i++) {

std::cout << "Scaner number " << i + 1<< " parameters :";

std::cout << "\n Model : " << scaners[i].model;

std::cout << "\n Price : " << scaners[i].price;

std::cout << "\n X\_size : " << scaners[i].x\_size;

std::cout << "\n Y\_size : " << scaners[i].y\_size;

std::cout << "\n Optr : " << scaners[i].optr;

std::cout << "\n Grey : " << scaners[i].grey;

std::cout << "\n\n";

}

}

void write\_struct(scan\_info scaners[]) {

FILE \* file;

char file\_name[20];

std::cout << "Enter file name : ";

std::cin >> file\_name;

file = fopen(file\_name, "wb");

/\*

char model [25]; //найменування моделі

int price; // ціна

double x\_size; // області сканування // горизонтальний розмір

double y\_size; // області сканування // вертикальний розмір

int optr; // оптичний дозвіл

int grey; // число градацій сірого

\*/

for (int i = 0; i < scaners\_number; i++) {

fprintf(file, "%s,", scaners[i].model);

fprintf(file, "%i,", scaners[i].price);

fprintf(file, "%lf,", scaners[i].x\_size);

fprintf(file, "%lf,", scaners[i].y\_size);

fprintf(file, "%i,", scaners[i].optr);

fprintf(file, "%i;", scaners[i].grey);

}

fclose(file);

}

int read\_struct(scan\_info scaners[]) {

FILE \* file;

char file\_name[20];

std::cout << "Enter file name : ";

std::cin >> file\_name;

file = fopen(file\_name, "rb");

if (file == NULL) {

std::cout << "Can not open file !!!\n";

return -1;

}

else {

std::cout << "Data was read seccesfully !!!\n";

}

for (int i = 0; i < scaners\_number; i++) {

for (int j = 0; j < 25; j++) {

fscanf(file, "%c", &scaners[i].model[j]);

if (scaners[i].model[j] == ',') {

scaners[i].model[j] = '\0';

break;

}

}

fscanf(file, "%i", &scaners[i].price);

fscanf(file, "%c");

fscanf(file, "%lf", &scaners[i].x\_size);

fscanf(file, "%c");

fscanf(file, "%lf", &scaners[i].y\_size);

fscanf(file, "%c");

fscanf(file, "%i", &scaners[i].optr);

fscanf(file, "%c");

fscanf(file, "%i", &scaners[i].grey);

fscanf(file, "%c");

}

fclose(file);

return 0;

}

**Висновок :** виконуючи дану лабораторно роботу, я навчився використовувати роботу з файлами та працювати з різними типами інформації зокрема зі структурами.