

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний
інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної
техніки Кафедра інформатики та програмної
інженерії

Звіт

з лабораторної роботи № 1 з дисципліни
«Основи програмування-2. Методології програмування»
«Файли даних. Бінарні файли»

Варіант 06

Виконав студент ІП-13 Вдовиченко Станіслав Юрійович
(шифр, прізвище, ім'я, по батькові)

Перевірив Всечерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Київ 2022

Лабораторна робота №1.2

Файли даних. Бінарні файли

Варіант 6.

Мета роботи: вивчити особливості створення і обробки бінарних файлів.

Завдання: створити файл з інформацією про товари меблевого магазину: назва товару, його вид (наприклад: стілець кухонний, стілець барний, стілець офісний, стіл кухонний, стіл комп'ютерний, тощо), колір, ціна та наявна кількість. В новому бінарному файлі сформувати список усіх наявних у магазині стільців вказаного виду, їх кількість та вартість. Видалити з нового файлу інформацію про стільці вартістю від 300 до 500 грн.

Постановка задачі: Заповнюємо структуру даними від користувача, записуємо дані в перший файл. Після цього читаємо дані з першого файла та формуємо структуру, залишаючи лише ті стільці, які відповідають введеному виду. Записуємо отриманий список у другий файл. Зчитуємо другий файл і видаляємо з нього стільці, вартість яких від 300 до 500, перезаписуємо структуру в той самий файл.

Код програми

C++

Header.h

```
#include <iostream>
#include <string>
#include <fstream>
#include "vector"
#include "cstring"
using namespace std;
struct catalog{
    char name[70];
    char type[40];
    char color[30];
    int price;
    int amount;
};

void input_file(const char*);
void output_file(const char*);
void create_second_file(const char*, const char*);
void edit_file(const char*);
```

Header.cpp

```
#include "header.h"

void input_file(const char* name){
    ofstream file;
    catalog product{};
    int mode = 0;
    int number;
    do {
        cout << "Select the mode: \n 1 - create new file. \n 2 - append file.\n";
        cin >> mode;
    }while(mode != 1 && mode != 2);
    if (mode == 1)
        file.open(name, ios::binary );
    else if (mode == 2)
        file.open(name, ios::binary | ios::app);
    if(!file.is_open()) {
        cout << "Cant open the file!";
        exit(0);
    }
    do {
        cout << "Enter number of products: ";
        cin >> number;
    }while(number <= 0);
    for (int i = 0; i < number; ++i) {
        cout << "Enter name of product: "; cin >> product.name;
        cout << "Enter type of product: "; cin >> product.type;
        cout << "Enter color of product: ";cin >> product.color;
        cout << "Enter price of product: ";cin >> product.price;
        cout << "Enter amount of product: ";cin >> product.amount;
        cout << endl;
        file.write((char*)&product, sizeof(catalog));
    }
    file.close();
}

void output_file(const char* name){
    catalog product{};
```

```

    ifstream file(name, ios::binary);
    while(file.read((char*)&product, sizeof(catalog))){
        cout << "Name: " << product.name << endl;
        cout << "Type: " << product.type << endl;
        cout << "Color: " << product.color << endl;
        cout << "Price: " << product.price << endl;
        cout << "Amount: " << product.amount << endl;
        cout << endl;
    }
    file.close();
}

void create_second_file(const char*first_name, const char* second_name){
    ifstream first_file(first_name, ios::binary);
    ofstream second_file(second_name, ios::binary);
    catalog product{};
    char new_name[40];
    char kind[40];
    cout << "Enter the name of products you'd like to add to new file: ";
    cin >> new_name;
    cout << "Enter the type of " << new_name << " you'd like to add to new file: ";
    cin >> kind;

    while(first_file.read((char*)&product, sizeof(catalog))){
        if(strcmp(product.name, new_name) == 0 && strcmp(product.type, kind) == 0 && product.amount > 0){
            second_file.write((char*)&product, sizeof(catalog));
        }
    }
    first_file.close();
    second_file.close();
}

void edit_file(const char* name) {
    catalog product{};
    const char* temp_file_name = "third.dat";
    ifstream file(name, ios::binary);
    ofstream temp_file(temp_file_name, ios::binary);
    int min = 300;
    int max = 500;
    while (file.read((char *) &product, sizeof(catalog))) {
        if (product.price < min || product.price > max) {
            temp_file.write((char*)&product, sizeof(catalog));
        }
    }
    file.close();
    temp_file.close();
    remove(name);
    rename(temp_file_name, name);
}

```

Main.cpp

```
#include "header.h"

int main() {
    const char* first = "first.dat";
    const char* second = "second.dat";
    input_file(first);
    cout << "First file: " << endl;
    output_file(first);
    create_second_file(first, second);
    cout << "Second file: " << endl;
    output_file(second);
    edit_file(second);
    cout << "\nEdited file: " << endl;
    output_file(second);
    return 0;
}
```

Робота програми:

Select the mode:

1 - create new file.

2 - append file.

1

Enter number of products:5

Enter name of product:chair

Enter type of product:bar

Enter color of product:blue

Enter price of product:

400

Enter amount of product:5

Enter name of product:chair

Enter type of product:bar

Enter color of product:red

Enter price of product:600

Enter amount of product:3

Enter name of product:chair

Enter type of product:bathroom

Enter color of product:white

Enter price of product:436

Enter amount of product:0

Enter name of product:table

Enter type of product:bar

Enter color of product:red

Enter price of product:400

Enter amount of product:4

Enter name of product:bed

Enter type of product:bunk

Enter color of product:white

Enter price of product:1200

Enter amount of product:4

First file:

Name: chair

Type: bar

Color: blue

Price: 400

Amount: 5

Name: chair

Type: bar

Color: red

Price: 600

Amount: 3

Name: chair

Type: bathroom

Color: white

Price: 436

Amount: 0

Name: table

Type: bar

Color: red

Price: 400

Amount: 4

Name: bed

Type: bunk

Color: white

Price: 1200

Amount: 4

Enter the name of products you'd like to add to new file: *chair*

Enter the type of chair you'd like to add to new file: *bar*

Second

file:

Name: chair

Type: bar

Color: blue

Price: 400

Amount: 5

Name: chair

Type: bar

Color: red

Price: 600

Amount: 3

Edited file:

Name: chair

Type: bar

Color: red

Price: 600

Amount: 3

Process finished with exit code 0

Python
Module.py

```
import pickle
import os

def inputFile(name):
    product = {}
    while True:
        mode = int(input("Enter 1 if u want to create a new file. \nEnter 2 if you want to append the file.\n"))
        if mode == 1:
            file = open(name, "wb")
            break
        elif mode == 2:
            file = open(name, "ab")
            break
    number = int(input("Enter number of products: "))
    while number <= 0:
        number = int(input("Enter number of products: "))
    for i in range(number):
        product["name"] = input("Name: ")
        product["type"] = input("Type: ")
        product["color"] = input("Color: ")
        product["price"] = int(input("Price: "))
        product["amount"] = int(input("Amount: "))
        print("\n")
        pickle.dump(product, file)

    file.close()

def showFile(name):
    file = open(name, "rb")
    filesize = file.seek(0, 2)
    file.seek(0)

    while file.tell() < filesize:
        product = pickle.load(file)
        print("\nName: ", product["name"])
        print("Type: ", product["type"])
        print("Color: ", product["color"])
        print("Price: ", product["price"])
        print("Amount: ", product["amount"])

    file.close()

def createSecondFile(first_name, second_name):
    requested_name = input("Enter the name of products you'd like to add to new file: ")
    requested_type = input("Enter the type of that product you'd like to add to new file: ")
    first_file = open(first_name, "rb")
    second_file = open(second_name, "wb")
    filesize = first_file.seek(0, 2)
    first_file.seek(0)
    while first_file.tell() < filesize:
        product = pickle.load(first_file)
        if product["name"] == requested_name and product["type"] == requested_type and product["amount"] > 0:
```

```

        pickle.dump(product, second_file)
    first_file.close()
    second_file.close()

def editFile(name):
    temp_name = "third.dat"
    file = open(name, "rb")
    temp_file = open(temp_name, "wb")
    filesize = file.seek(0, 2)
    file.seek(0)
    while file.tell() < filesize:
        product = pickle.load(file)
        if product["price"] < 300 or product["price"] > 500:
            pickle.dump(product, temp_file)
    file.close()
    temp_file.close()
    os.remove(name)
    os.rename(temp_name, name)

```

main.py

```

from module import *

firstName = "first.dat"
secondName = "second.dat"
inputFile(firstName)
showFile(firstName)
createSecondFile(firstName, secondName)
print("Second file: ")
showFile(secondName)
editFile(secondName)
print("\nEdited file: ")
showFile(secondName)

```

Робота програми:

Enter 1 if u want to create a new file.
Enter 2 if you want to append the file.

1

Enter number of products: 4

Name: *chair*

Type: *bar*

Color: *red*

Price: *400*

Amount: *5*

Name: *chair*

Type: *bar*

Color: *white*

Price: *600*

Amount: *5*

Name: *chair*

Type: *bar*

Color: *blue*

Price: *500*

Amount: *0*

Name: *table*

Type: *bar*

Color: *red*

Price: *400*

Amount: *5*

Name: chair
Type: bar
Color: red
Price: 400
Amount: 5

Name: chair
Type: bar
Color: white
Price: 600
Amount: 5

Name: chair
Type: bar
Color: blue
Price: 500
Amount: 0

Name: table
Type: bar
Color: red
Price: 400
Amount: 5

Enter the name of products you'd like to add to new file: *chair*
Enter the type of that product you'd like to add to new file: *bar*

```
Second file:
```

```
Name:  chair
```

```
Type:  bar
```

```
Color:  red
```

```
Price:  400
```

```
Amount:  5
```

```
Name:  chair
```

```
Type:  bar
```

```
Color:  white
```

```
Price:  600
```

```
Amount:  5
```

```
Edited file:
```

```
Name:  chair
```

```
Type:  bar
```

```
Color:  white
```

```
Price:  600
```

```
Amount:  5
```

```
Process finished with exit code 0
```

Висновок: під час виконання даної лабораторної роботи я навчився працювати з бінарними файлами на мовах C++ та Python.