

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 № 2 з дисципліни
«Основи програмування 2. Модульне програмування»

«Файли даних»

Варіант 29

Виконав студент

ПІ-15 Рибалка Ілля Сергійович
(шифр, прізвище, ім'я, по батькові)

Перевірів

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

Лабораторна робота 1

Бінарні файли

Мета - вивчити особливості створення і обробки бінарних файлів.

Індивідуальне завдання

Вимоги до програми

- 1) Створення файлу програмним шляхом.
- 2) Організація введення інформації аналогічно текстовим редакторам (ознака кінця рядка – натиснення Enter, ознака кінця введення файла – будь-яка комбінація клавіш або клавіша, що генерують розширений код).
- 3) Можливість доповнення файла даних.
- 4) Виведення усіх вхідних, проміжних і вихідних даних.

Варіант 29

29. Створити файл із списком покупців, які придбали товари зі знижкою на день акції: прізвище, стать, дата народження, кількість одиниць товару. Передбачається, що вартість однієї одиниці товару - 100 грн., знижка на товар дорівнює віку особи. Пенсіонерам (з 60-ти років) надається додаткова знижка 5%. Визначити виторг магазину за день. Створити новий файл з інформацією про покупців, які придбали товару більш ніж на 250 грн.

Розв'язання

Код

C++

```
#include "func.h"

int main()
{
    string path1 = "input.bin", path2 = "output.bin";
    inputfile(path1);
    croutfile(path1, path2);
    printfile(path1);
    printfile(path2);
    return 0;
}
```

func.h

```
#pragma once
#include <iostream>
#include <fstream>
#include <vector>
#include <ctime>
using namespace std;

struct customer
{
    string surn;
    char sex;
    string age;
```

Основи програмування 2. Модульне програмування

```
int amt;
};

void inputfile(string); // Введення файлу через консоль
bool validation(customer, bool); // Перевірка введених даних
bool strtostuct(string, customer *); // Переведення зі строки в структуру
int years(string); // Визначення кількості років
void croutfile(string, string); // Створення вихідного файлу
int disc(string, int); // Визначення ціни зі знижкою
vector<customer> lstlines(string); // Створення вектора з файлу
void printfile(string); // Вивід файлу в консоль
```

func.cpp

```
#include "func.h"

void inputfile(string path)
{
    bool val;
    string str, ans;
    customer *cust = new customer;
    ofstream file;
    ifstream f(path, ofstream::binary);
    if (f.is_open())
    {
        if(!f.eof())
        {
            cout << "Очистити файл? Y/N ";
            cin >> ans;
            while(!((ans == "Y" || ans == "y") || (ans == "N" || ans == "n")))
            {
                cout << "Введіть Y або N ";
                cin >> ans;
            }
            if(ans == "Y" || ans == "y") file.open(path, ofstream::binary);
            else if(ans == "N" || ans == "n") file.open(path, ofstream::app | ofstream::binary);
            cin.ignore();
        }
    }
    else file.open(path, ofstream::binary);
    f.close();
    cout << "Введіть данні покупців за шаблоном Прізвище;Стать (М/В);Дата народження (DD.MM.YYYY);Кількість товару" << endl;
    getline(cin, str);
    while(str.find(1) == string::npos) //Ознака кінця введення файлу Ctrl+A
    {
        val = strtostuct(str, &*cust);
        if(validation(*cust, val)) file.write((char*)cust, sizeof(customer));
        getline(cin, str);
    }
    file.close();
}

bool validation(customer cust, bool val)
{
    if(!val)
    {
        cout << "Некоректно введені дані" << endl;
        return false;
    }
    if(cust.sex != 'M' && cust.sex != 'm' && cust.sex != 'W' && cust.sex != 'w')
    {
        cout << "Некоректно введена стать" << endl;
        return false;
    }
    int age = years(cust.age);
    if(age < 16 || age >= 122)
    {
        if(age >= 0)
        {
            if (age < 16) cout << "Введена дата замала" << endl;
            else cout << "Введена дата завелика" << endl;
        }
        else cout << "Некоректно введена дата" << endl;
        return false;
    }
    if(cust.amt <= 0)
    {
        cout << "Некоректно введена кількість товару" << endl;
        return false;
    }
    return true;
}

bool strtostuct(string str, customer *cust)
{
    vector<string> words;
    int n1 = 0, n2 = str.find(';'), n = 0;
```

Основи програмування 2. Модульне програмування

```
while(n2!=-1 && n<4)
{
    words.push_back(str.substr(n1,n2-n1));
    n++;
    n1 = n2+1;
    n2 = str.find(';', n1+1);
}
words.push_back(str.substr(n1,str.size()-n1));
if(n==3)
{
    cust->surname = words[0];
    cust->sex = words[1][0];
    cust->age = words[2];
    cust->amt = stoi(words[3]);
    return true;
}
return false;
}

int years(string date)
{
    int val[13]={0,31,28,31,30,31,30,31,31,30,31,30,31}; // Кількість днів у кожному місяці року
    int age, day, mon, year;
    int t1=date.find("."), t2=date.find(".", t1+1);
    struct tm inptime;
    day = stoi(date.substr(0,t1));
    mon = stoi(date.substr(t1+1,t2-t1));
    year = stoi(date.substr(t2+1,date.size()-t2));
    if(day < 1 || day > val[mon] || mon < 1 || mon > 12 || year <= 1900) return -1;
    inptime.tm_mday = day;
    inptime.tm_mon = mon-1;
    inptime.tm_year = year-1900;
    inptime.tm_hour = 0;
    inptime.tm_min = 0;
    inptime.tm_sec = 0;
    time_t dt1, dt2 = mktime(&inptime);
    time(&dt1);
    age = difftime(dt1, dt2)/(60*60*24*365.2425);
    return age;
}

void croutfile(string path1, string path2)
{
    ifstream inpf(path1, ifstream::binary);
    ofstream outf(path2, ofstream::binary);
    customer *cust = new customer;
    string ans;
    int price;
    bool swpr = false;
    cout << "Виводити вартість товарів? Y/N ";
    cin >> ans;
    while(!(ans == "Y" || ans == "y" || (ans == "N" || ans == "n")))
    {
        cout << "Введіть Y або N ";
        cin >> ans;
    }
    if(ans == "Y" || ans == "y") swpr = true;
    while(inpf.read((char*)cust, sizeof(customer)))
    {
        price = disc(cust->age, cust->amt);
        if(price>250) outf.write((char*)cust, sizeof(customer));
        if(swpr) cout << cust->surname << " - " << price << endl;
    }
    outf.close();
}

int disc(string date, int amt)
{
    int n=amt*100, age=years(date);
    if(age<60) n-=n*age*0.01;
    else if(age>=95) n-=n*0.99;
    else n-=n*((age*0.01)+0.05);
    return n;
}

vector<customer> lstlines(string path)
{
    ifstream file(path, ifstream::binary);
    customer *cust = new customer;
    vector<customer> lst;
    while (file.read((char*)cust, sizeof(customer))) lst.push_back(*cust);
    file.close();
    return lst;
}

void printfile(string path)
{
    cout << "Зміст файлу " << path << endl;
    vector<customer> lst = lstlines(path);
}
```

Основи програмування 2. Модульне програмування

```
if (lst.size()!=0)
{
    for (int i = 0; i < lst.size(); i++) cout << lst[i].surn << ";" << lst[i].sex << ";" << lst[i].age << ";" << lst[i].amt << endl;
}
else cout << "Файл пустий" << endl;
}
```

Python

```
from func import *

path1 = 'Lab2/Python/input.bin'
path2 = 'Lab2/Python/output.bin'

inputfile(path1)
croutfile(path1, path2)
printfile(path1)
printfile(path2)
```

func.py

```
from datetime import datetime
from os import stat
from os.path import exists
import pickle

# Введення файлу через консоль
def inputfile(path):
    if exists(path):
        if stat(path).st_size != 0:
            ans = input("Очистити файл? Y/N ")
            while ans.lower() != "y" and ans.lower() != "n":
                ans = input("Введіть Y або N ")
            if ans.lower() == "y":
                lst = []
            else:
                lst = lstlines(path)
        else:
            lst = []
    else:
        lst = []
    file = open(path, "wb")
    string = input("Введіть дані покупців за шаблоном Прізвище;Стать (М/В) ;Дата народження (DD.MM.YYYY) ;Кількість товару\n")
    while chr(1) not in string: #Ознака кінця введення файлу Ctrl+A
        if validation(string):
            lst.append(strtostruct(string))
            string = input()
        pickle.dump(lst, file)
    file.close()

# Перевірка введених даних
def validation(string):
    if ";" not in string or string.count(";") != 3:
        print("Некоректно введені дані")
        return False
    lst = string.split(";")
    age = years(lst[2])
    if lst[1].lower() != "w" and lst[1].lower() != "m":
        print("Некоректно введена стать")
        return False
    if age < 16 or age >= 122:
        if age >= 0:
            if age < 16:
                print("Введена дата замала")
            else:
                print("Введена дата завелика")
        else:
            print("Некоректно введена дата")
        return False
    if int(lst[3]) <= 0:
        print("Некоректно введена кількість товару")
        return False
    return True

# Переведення зі строки в структуру
def strtostruct(string):
    cust = {}
    lst = string.split(";")
    cust["surn"] = lst[0]
    cust["sex"] = lst[1]
    cust["age"] = lst[2]
    cust["amt"] = int(lst[3])
    return cust
```

Основи програмування 2. Модульне програмування

```
# Визначення кількості років
def years(date):
    val = [0,31,28,31,30,31,30,31,31,30,31,30,31]
    tdate = date.split(".")
    if int(tdate[1])<13 and int(tdate[1])>0:
        if int(tdate[0])>0 and int(tdate[0])<=val[int(tdate[1])]:
            curdate = datetime.now()
            inptime = datetime.strptime(date, "%d.%m.%Y")
            res = (curdate-inptime).days
            return int(res/365.2425)
        return -1
    else:
        return -1

# Створення вихідного файлу
def croufile(path1, path2):
    res = []
    inpf = lstlines(path1)
    outf = open(path2, "wb")
    ans = input("Виводити вартість товарів? Y/N ")
    while ans.lower() != "y" and ans.lower() != "n":
        ans = input("Введіть Y або N ")
    if ans.lower() == "y":
        swpr = True
        for dct in inpf:
            price = disc(dct["age"], dct["amt"])
            if(price>250):
                res.append(dct)
            if(swpr):
                print(dct["surn"], " - ", price)
        pickle.dump(res,outf)
    outf.close()

# Визначення ціни зі знижкою
def disc(date, amt):
    n=amt*100
    age=years(date)
    if age<60:
        n-=n*age*0.01
    elif age>=95:
        n-=n*0.99
    else:
        n-=n*((age*0.01)+0.05)
    return int(n)

# Створення списку з файлу
def lstlines(path):
    file = open(path, "rb")
    lst = pickle.load(file)
    file.close()
    return lst

# Вивід файлу в консоль
def printfile(path):
    print("Зміст файлу", path.split("/")[-1:][0])
    lst = lstlines(path)
    if len(lst)!=0:
        for i in lst:
            prt = []
            for key in i:
                prt.append(str(i[key]))
            print(";".join(prt))
    else:
        print("Файл пустий")
```

Тестування

C++

```
Введіть данні покупців за шаблоном Прізвище;Стать(М/Ж);Дата народження(DD.MM.YYYY);Кількість товару
qwe;l;12.12.2000;1
Некоректно введена стать
qwe;m;12.13.2000;1
Некоректно введена дата
qwe;m;32.12.2000;1
Некоректно введена дата
qwe;m;12.12.2012;1
Введена дата замала
qwerty
Некоректно введені дані
qwe;M;12.12.2000;1000
rty;W;12.12.1950;10
^A
Виводити вартість товарів? Y/N Y
qwe - 79000
rty - 240
Зміст файлу input.bin
qwe;M;12.12.2000;1000
rty;W;12.12.1950;10
Зміст файлу output.bin
qwe;M;12.12.2000;1000
```

Python

```
Введіть данні покупців за шаблоном Прізвище;Стать(М/В);Дата народження(DD.MM.YYYY);Кількість товару
qwe;p;12.03.2000;1
Некоректно введена стать
qwerty
Некоректно введені дані
qwe;w;12.30;2000;1
Некоректно введені дані
qwe;m;12.12.1930;11
^A
Виводити вартість товарів? Y/N Y
qwe - 44
Зміст файлу input.bin
qwe;m;12.12.1930;11
Зміст файлу output.bin
Файл пустий
```

Висновок

Я дослідив особливості створення і обробки бінарних файлів. В ході роботи на лабораторною роботою було створено програму для внесення даних про покупців в бінарний файл і на основі даного файлу створення вихідного файлу згідно завдання.