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. Об’єктно-орієнтоване програмування»

«Файли даних. Бінарні файли»

Варіант 21

Виконав студент ІП-11, Ляля Іван Олександрович

Перевірила Вітковська Ірина Іванівна\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Київ 2022

**Лабораторна робота 2**

**Файли даних. Бінарні файли**

**Мета** – вивчити особливості створення і обробки бінарних файлів.

**Задача:**

Створити файл із списком автомобілів автосалону: назва, дата випуску, дата надходження у продаж. Створити список нових автомобілів (які надійшли у продаж не більш як через 2 місяці після випуску). Вивести інформацію про автомобілі, які були випущені не раніше вказаного року.

**Постановка задачі:**

На С++ ми підключаємо бібліотеки iostream для потокового i/o, fstream для файлового i/o, string роботи з текстом через рядки і їх методи. На обох мовах опишемо функції File\_Create, NewAutosListCreate, Output, AutosReleasedAfter, які прийматимуть параметри-імена файлів.

File\_Create: відкриває на до запис або перезапис файл, зчитує для автомобілів у кількості, бажаній користувачем, модель, дату випуску і дату надходження у продаж, після чого записує дані в файл, закриває цей файл.

NewAutosListCreate: відкриває перший файл у режимі читання, другий – запису. Поки можливо – зчитує дані про автомобіль, і для кожного з них виконує наступне: витягує місяць і рік випуску, місяць і рік надходження у продаж; далі якщо різниця між випуском і надходженням у продаж не більше двох місяців (умова перевіряється), то дані про цей автомобіль записуються у другий файл; закрити файли.

Output: відкриває файл на зчитування і по одному автомобілю виводить їх дані.

AutosReleasedAfter: відкриває файл у режимі читання, запитує користувача від якого року автомобілі відібрати, зчитує по одному автомобілю, і для кожного з них виконує наступне: витягує рік випуску, порівнює з введеним користувачем роком, якщо рік випуску більше рівне введений користувачем рік – виводить дані про цей автомобіль на екран. Закрити файл.

Головна функція є такою: присвоюємо імена: вхідному файлу “file 1.txt”, вихідному - ”file 2.txt”. Викликаємо функцію заповнення першого файлу, функцію виводу його на екран, функцію що з цього файла відбере «нові» автомобілі і запише їх у другий файл, функцію виводу другого файлу і останню функцію – відбору з першого файла автомобілів, що були випущені, починаючи з вказаного року.

**Текст і результат роботи програми на С++:**

**cpp lab 2.cpp:**

#include "Header.h"

int main()

{

string name1 = "file 1.txt", name2 = "file 2.txt";

File\_Create(name1); //створити файл із списком автомобілів автосалону

cout << "\nWhole auto list (file 1):" << endl;

Output(name1); //вивести файл із списком автомобілів автосалону

NewAutosListCreate(name1, name2); //створити список нових автомобілів

cout << "\nNew autos' list (file 2):" << endl;

Output(name2); //вивести список нових автомобілів

AutosReleasedAfter(name1); //вивести інформацію про автомобілі, які були випущені не раніше вказаного року

return 0;

}

**Header.h:**

#pragma once

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

struct Automobile

{

char Model[20], ReleaseDate[8], DateofSale[8]; //модель, дата випуску і дата надходження у продаж

};

void File\_Create(string);

void NewAutosListCreate(string, string);

void Output(string);

void AutosReleasedAfter(string);

**Source.cpp:**

#include "Header.h"

void File\_Create(string name)

{

bool mode;

cout << "Do you want to append file (1) or create a clear file (0)? "; //запитати користувача в режимі перезапису чи дозапису відкрити файл

cin >> mode;

ofstream outfile1;

if (mode) outfile1.open(name, ios::app, ios::binary);

else outfile1.open(name, ios::binary);

Automobile A;

cout << "Input the number of cars you want to add: "; //запитати користувача скільки автомобілів хоче додати до списку

int number; cin >> number;

cin.ignore();

for (int i = 0; i < number; i++) //для кожного автомобіля зчитуємо модель, дату випуску і дату надходження у продаж, записуємо структуру з цим автомобілем у файл

{

cout << "Input car model: "; cin>>A.Model;

cout << "Input release date (MM.YYYY): "; cin>>A.ReleaseDate;

cout << "Input date of sale (MM.YYYY): "; cin>>A.DateofSale;

outfile1.write((char\*)&A, sizeof(Automobile));

}

outfile1.close();

}

void Output(string name)

{

ifstream file(name, ios::binary);

Automobile A;

while (file.read((char\*)&A, sizeof(Automobile))) //поки можна, зчитати дані про авто і вивести на екран

cout << A.Model << ' ' << A.ReleaseDate << ' ' << A.DateofSale << endl;

file.close();

}

void NewAutosListCreate(string name1, string name2)

{

ifstream file1(name1);

ofstream file2(name2);

Automobile A;

string releaseYear="0000", releaseMonth="00", saleYear="0000", saleMonth="00"; //рік випуску, місяць випуску, рік надходження у продаж, місяць надходження у продаж

while (file1.read((char\*)&A, sizeof(Automobile)))

{

for (int i = 0; i < 2; i++) //місяць випуску чи надходження у продаж при такому форматі запису буде зазначатись у першому-другому елементах рядка

{

releaseMonth[i] = A.ReleaseDate[i];

saleMonth[i] = A.DateofSale[i];

}

for (int i = 3; i < 7; i++) //рік випуску чи надходження у продаж при такому форматі запису буде зазначатись у четвертому-сьомому по порядку елементах рядка

{

releaseYear[i-3] = A.ReleaseDate[i];

saleYear[i-3] = A.DateofSale[i];

}

if ((12\*(stoi(saleYear)-stoi(releaseYear))+stoi(saleMonth)-stoi(releaseMonth))<=2) //щоб рахувати різницю у місяцях між датами випуску і надходження у продаж, треба рік другого мінус рік першого помножити на 12 (кількість місяців у році) додати місяць другого і відняти місяць першого

file2.write((char\*)&A, sizeof(Automobile));

}

file1.close();

file2.close();

}

void AutosReleasedAfter(string name)

{

cout << "\nInput the year to output autos released not earlier than: "; //запитати користувача, від якого року автомобілі показати

int X\_year; cin >> X\_year;

ifstream file(name, ios::binary);

Automobile A;

string releaseYear="0000"; //рік випуску автомобіля

cout << "Autos released in or after " << X\_year << ":"<<endl;

while (file.read((char\*)&A, sizeof(Automobile)))

{

for (int i=3; i<7; i++) //таким же чином, як і в попередній функції, дістаємо із рядка дату випуску автомобіля

releaseYear[i-3] = A.ReleaseDate[i];

if (stoi(releaseYear) >= X\_year) //якщо рік автомобіль із відповідного рядка був випущений у вказаному році або ще новіший, вивести дані цього автомобіля

{

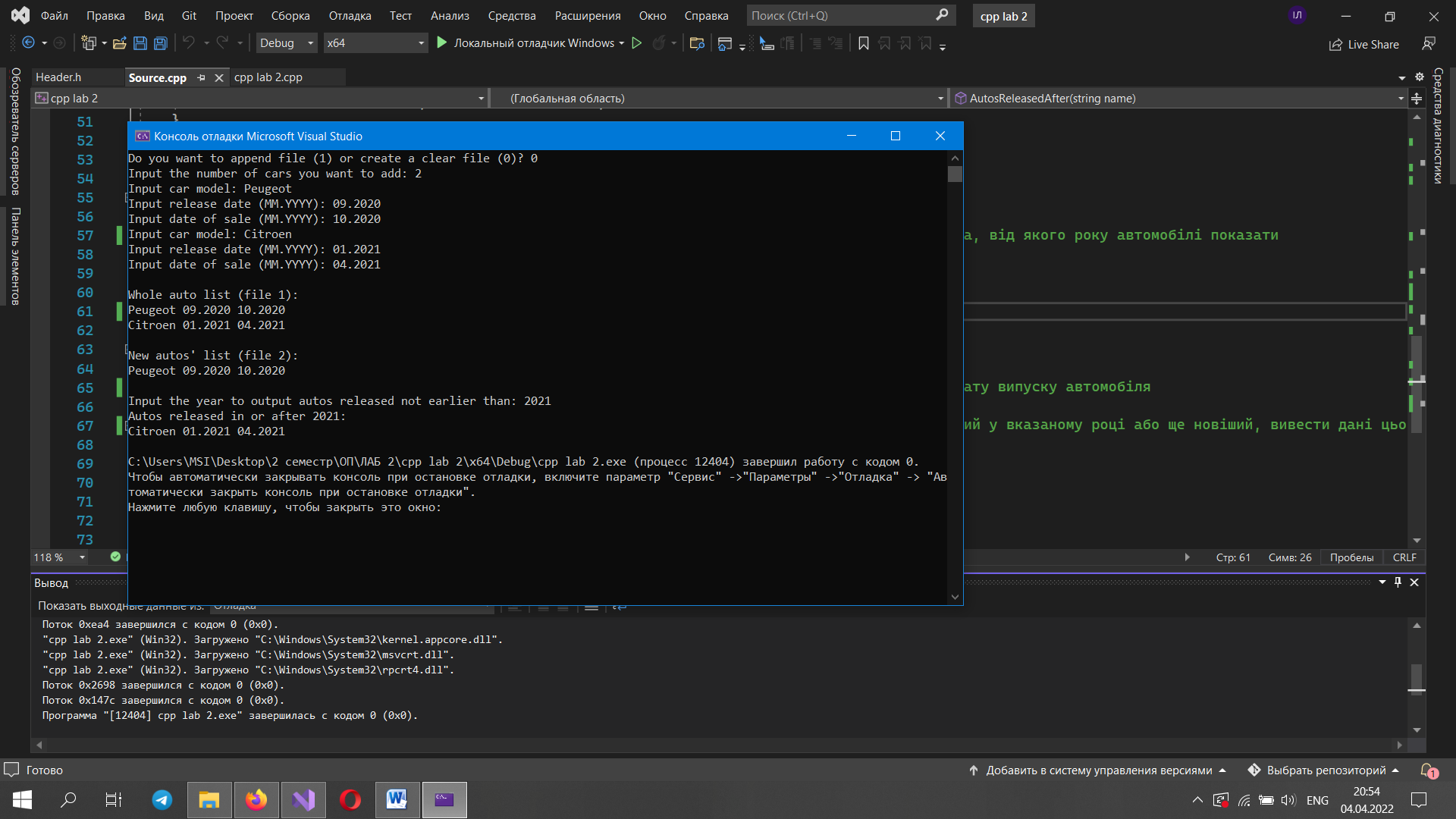
cout << A.Model << ' ' << A.ReleaseDate << ' ' << A.DateofSale << endl;

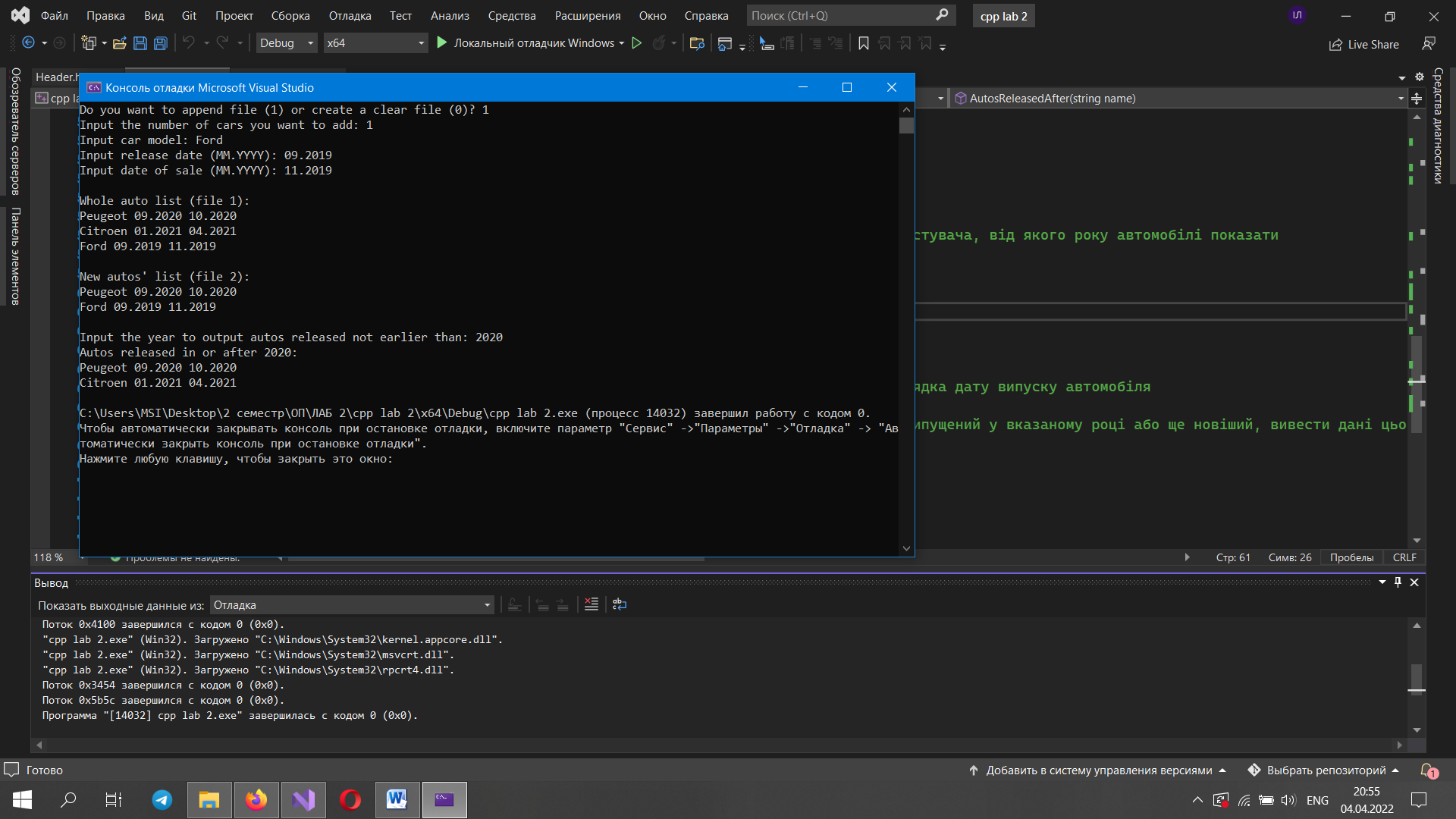
}

}

file.close();

}





**Текст і результат роботи програми на Python:**

**python\_lab\_2.py:**

import module

name1="file 1.txt"

name2="file 2.txt"

module.File\_Create(name1) # створити файл із списком автомобілів автосалону

print("Whole auto list (file 1):")

module.Output(name1) # вивести файл із списком автомобілів автосалону

module.NewAutosListCreate(name1, name2) # створити список нових автомобілів

print("New autos' list (file 2):")

module.Output(name2) # вивести список нових автомобілів

module.AutosReleasedAfter(name1) # вивести інформацію про автомобілі, які були випущені не раніше вказаного року

**module.py:**

import string

def File\_Create(name):

mode=int(input("Do you want to append file (1) or create a clear file (0)? ")) # запитати користувача, чи відкривати файл у режимі перезапису чи дозапису

if mode==1: m='ab'

else: m='wb'

with open(name,m) as outfile1:

number=int(input("Input the number of cars you want to add: ")) # запитати користувача скільки автомобілів він хоче додати у файл

for i in range(number):

Model=str(input("Input car model: ")) # для кожного автомобіля просимо ввести модель, дату випуску і дату надходження у продаж (дати у форматі місяць.рік, як зазвичай зазначається для автомобілів)

ReleaseDate=str(input("Input release date (MM.YYYY): "))

DateofSale=str(input("Input date of sale (MM.YYYY): "))

Automobile=Model+', '+ReleaseDate+', '+DateofSale+'\n' # записуємо ці три критерії в один рядок через кому і пробіл

if len(Automobile)>0:

outfile1.write(Automobile.encode()) # записуємо відповідний рядок у список всіх автомобілів салону (одночасно кодуючи його в bytes)

def Output(name):

with open(name, 'rb') as file:

for Automobile in file:

print(Automobile.decode()) # зчитуємо по рядку із файла, розшифровуємо із bytes і виводимо на екран

def NewAutosListCreate(name1, name2):

with open(name1,'rb') as file1:

with open(name2,'wb') as file2:

for Auto in file1:

Auto=Auto.decode() # по рядку зчитуємо і розкодовуємо

Automobile=Auto.split(', ') # розбиваємо по комі і пробілу рядок на list, де першим(нульовий індекс) елементом буде модель машини, а наступними відповідно будуть дата випуску і дата надходження у продаж

releaseYear=int(Automobile[1][3:7]) # рік випуску чи надходження у продаж при такому форматі запису буде зазначатись у четвертому-сьомому по порядку елементах рядка

releaseMonth=int(Automobile[1][:2]) # місяць випуску чи надходження у продаж при такому форматі запису буде зазначатись у першому-другому елементах рядка

saleYear=int(Automobile[2][3:7])

saleMonth=int(Automobile[2][:2])

if (12\*(saleYear-releaseYear)+saleMonth-releaseMonth)<=2: # щоб рахувати різницю у місяцях між датами випуску і надходження у продаж, треба рік другого мінус рік першого помножити на 12 (кількість місяців у році) додати місяць другого і відняти місяць першого

file2.write(Auto.encode());

def AutosReleasedAfter(name):

X\_year=int(input("Input the year to output autos released not earlier than: ")) # запитати користувача, від якого року автомобілі показати

with open(name,'rb') as file:

print("Autos released in or after " + str(X\_year) + ':')

for auto in file:

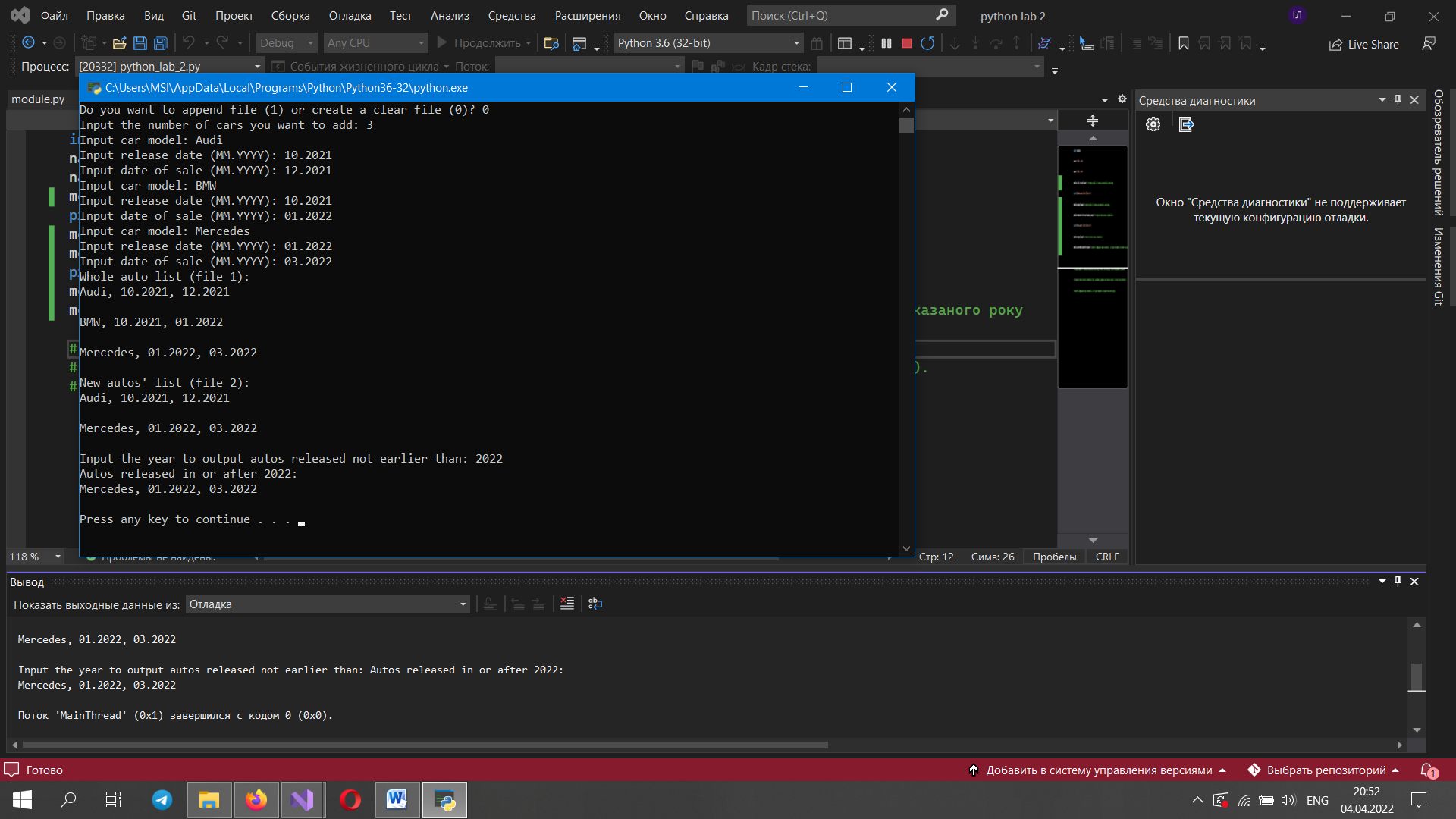
auto=auto.decode()

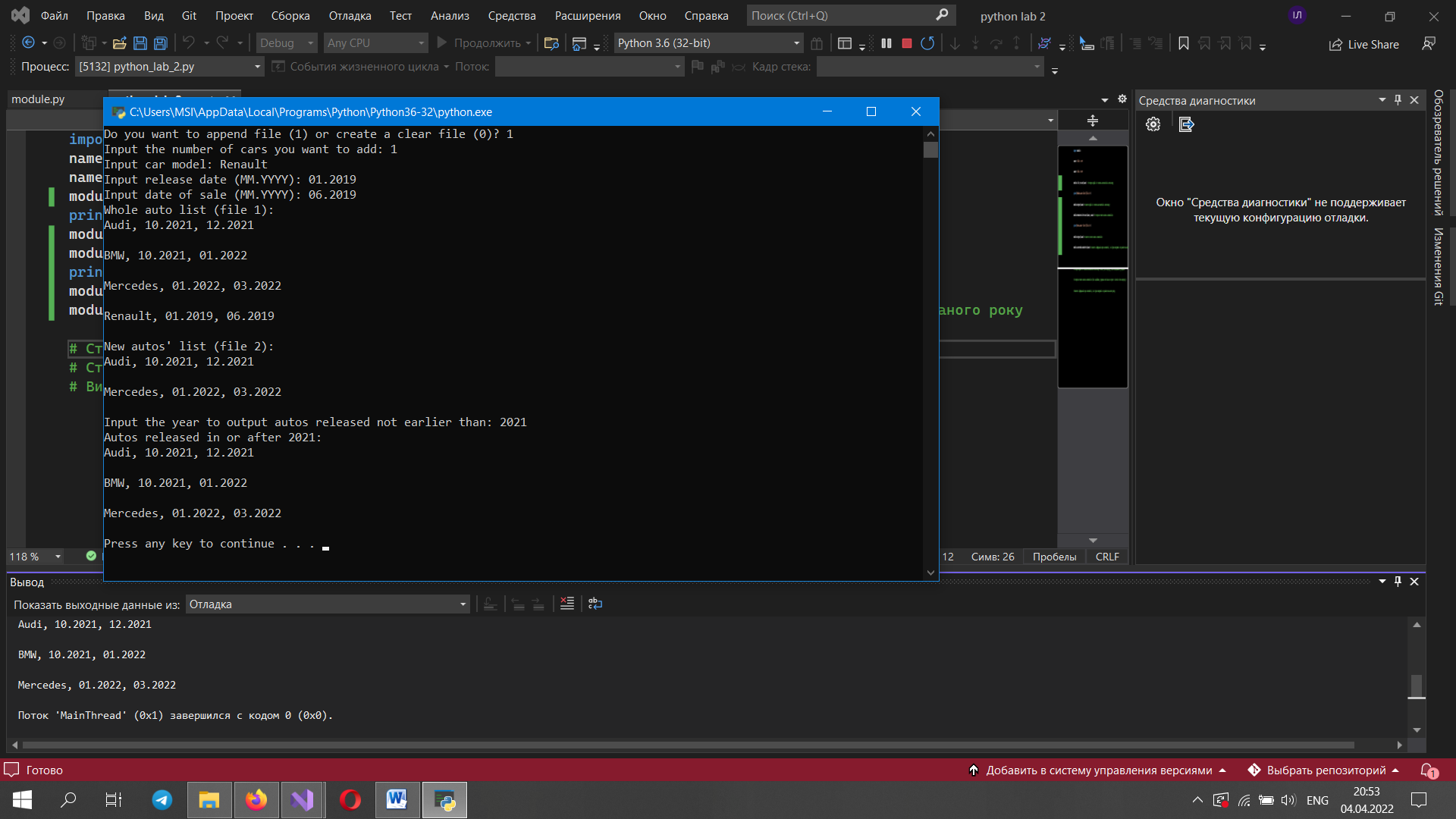
Automobile=auto.split(', ')

releaseYear=int(Automobile[1][3:7]) # таким же чином, як і в попередній функції, дістаємо із рядка дату випуску автомобіля

if (releaseYear >= X\_year):

print(auto) # якщо рік автомобіль із відповідного рядка був випущений у вказаному році або ще новіший, вивести цей рядок





**Висновок:**

Під час виконання лабораторної роботи мені довелося освоїти навички роботи з бінарними файлами даних. На мові С++ це реалізується методами побайтового записування і зчитування, на Python – методами encode та decode. Власне робота з даними відбувалась уже вивченими раніше методами роботи з структурами (на С++) та списками (Python). Особливістю задачі стала необхідність неодноразово використовувати перетворення типів.