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 № 3 з дисципліни

«Основи програмування – 2. Методології програмування»

«Класи та об’єкти»   
Варіант 22

Виконав студент ІП-13, Музичук Віталій Андрійович

(шифр, прізвище, ім'я, по батькові)

Перевірила Вєчерковська Анастасія Сергіївна

( прізвище, ім'я, по батькові)

Київ 2022

**Лабораторна робота 3**

**Класи та об’єкти**

**Мета** – вивчити механізми створення і використання класів та об’єктів.

**Варіант 22**

**Завдання:**

Створити клас «Студент», який характеризується ПІБ студента, номером його групи, датою народження (у форматі ДД-ММ-РРРР). Створити масив об’єктів даного класу. Визначити найстаршого студента вказаної групи (на вказану дату).

*1. Виконання завдання на мові С++:*

**// Lab\_3.cpp**

#include "func.h"

int main()

{

int num;

cout << "How many students do you want to add: ";

cin >> num; cin.ignore(32000, '\n');

Student\* arr = createArray(num);

int numberGroup;

cout << "Enter the group number in which you want to know the oldest student: ";

cin >> numberGroup; cin.ignore(32000, '\n');

getOldestStudent(arr, numberGroup, num);

}

**// func.cpp**

#include "func.h"

Student::Student() {}

Student::Student(string name, int group, string date)

: fullName(name), numGroup(group) {

getDate(date, birthDay, birthMonth, birthYear);

}

void Student::getInfo() {

cout << "Full name of student: " << fullName << "; Number of group: " << numGroup << endl;

printf\_s("Date of birth: %d.%d.%d\n\n", birthDay, birthMonth, birthYear);

}

int Student::getAge() {

return (birthDay + (birthMonth - 1) \* 31 + birthYear \* 365);

}

int Student::getGroup() { return numGroup; }

Student\* createArray(int num) {

Student\* arr = new Student[num];

cout << "Add information about sudent" << endl;

int numGroup; string fullName, date;

for (int i = 1; i <= num; i++) {

cout << "Student - "<< i << endl;

cout << "Enter a full name: ";

getline(cin, fullName);

cout << "Enter a number of group: ";

cin >> numGroup; cin.ignore(32000, '\n');

if (numGroup <= 0) {

cerr << "Invalid number group";

exit(0);

}

cout << "Enter a birthdate in format like this [DD.MM.YYYY]: ";

getline(cin, date); cout << endl;

arr[i - 1] = Student(fullName, numGroup, date);

}

return arr;

}

void getDate(string line, int& day, int& month, int& year) {

int dates[3]; int pos;

for (int i = 0; i < 3; i++)

{

if (i == 2)

dates[i] = stoi(line);

else {

pos = line.find('.');

dates[i] = stoi(line.substr(0, pos));

line.erase(0, pos + 1);

}

}

if (dates[2] <= 1900 || dates[2] > 2022) {

cerr << "You aren't Jesus. Enter your real birht year";

exit(0);

} else year = dates[2];

if (dates[1] <= 0 || dates[1] > 12) {

cerr << "Invalid enter month";

exit(0);

} else month = dates[1];

int monthDate[12] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };

if (dates[0] <= 0 || dates[0] > monthDate[dates[1] - 1]) {

cerr << "Invalid entered day";

exit(0);

} else day = dates[0];

}

void getOldestStudent(Student\* arr, int numberGroup, int amount) {

int num, tempAge;

for (int i = 0; i < amount; i++)

if (numberGroup == arr[i].getGroup()) {

num = i;

break;

}

int ageOldestStudent = arr[num].getAge();

for (int i = num + 1; i < amount; i++) {

if (numberGroup == arr[i].getGroup()) {

tempAge = arr[i].getAge();

if (tempAge <= ageOldestStudent)

{

ageOldestStudent = tempAge;

num = i;

}

}

}

cout << "\nThe oldest student in " << numberGroup << " group is: " << endl;

arr[num].getInfo();

}

**// func.h**

#ifndef FUNC\_H

#define FUNC\_H

#include <iostream>

#include <string>

#include <vector>

using namespace std;

class Student {

private:

string fullName;

int numGroup;

int birthDay;

int birthMonth;

int birthYear;

public:

Student();

Student(string name, int group, string date);

int getAge();

int getGroup();

void getInfo();

};

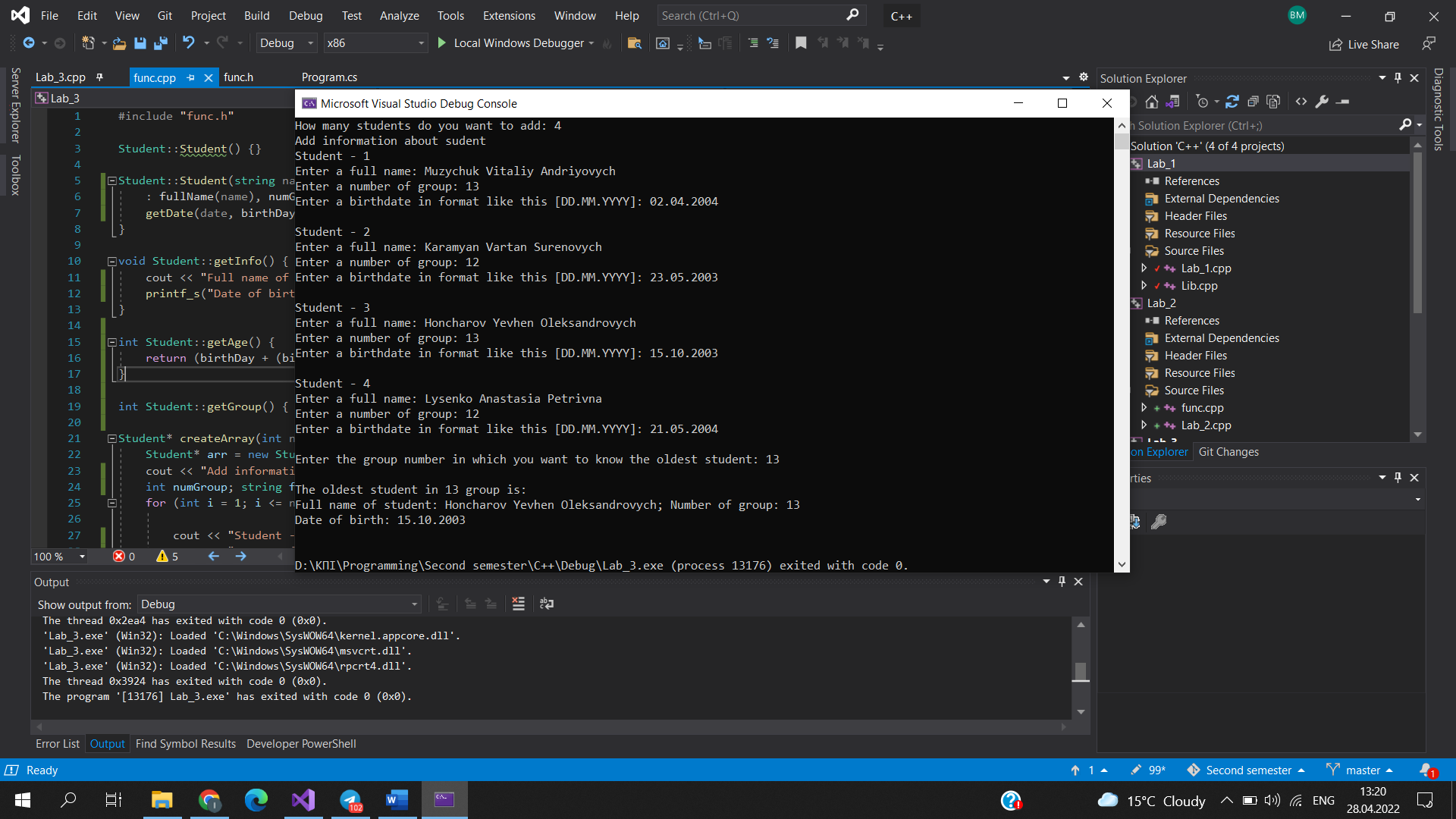
Student\* createArray(int num);

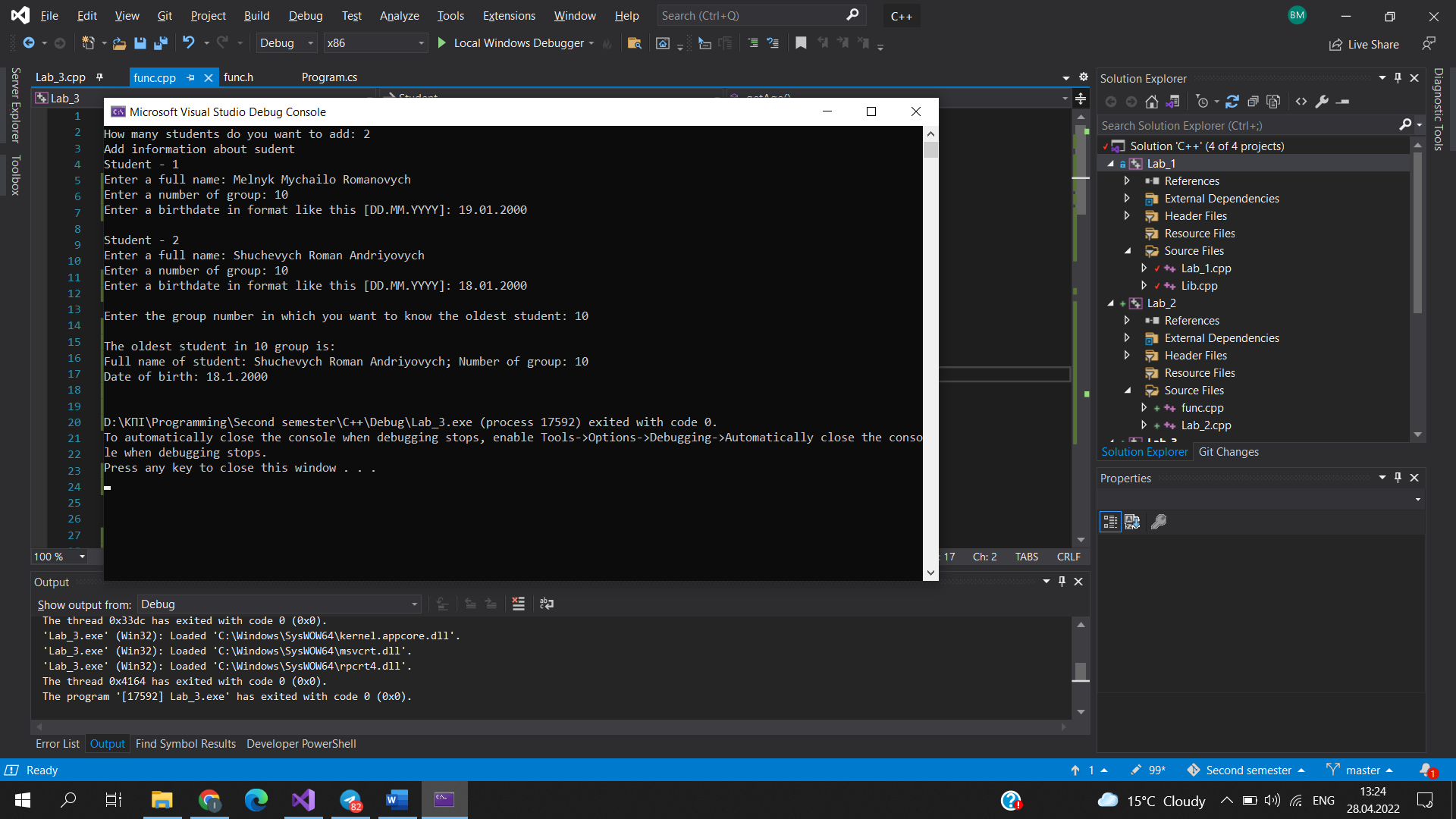
void getDate(string line, int& day, int& month, int& year);

void getOldestStudent(Student\* arr, int numberGroup, int amount);

#endif

**Тестування програми:**





**Висновок**: Під час виконання лабораторної роботи я вивчив особливості створення і використання класів та об’єктів на прикладі мови С++. Результатом виконання лабораторної роботи є програма, основним завданням якої є створення масиву об’єктів студентів та визначення найстаршого студента в групі. Після тестування програм можна зробити висновок, що вони справляються із поставленою задачею.