Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 11

з навчальної дисципліни

“Базові методології та технології програмування”

На тему:

“ Реалізація програмних модулів оброблення динамічних структур даних та бінарних файлів”

ВИКОНАВ

студент академічної групи КБ-21

\_\_\_\_\_\_\_\_\_ К.О. Задорожний

ПЕРЕВІРИВ

викладач кафедри кібербезпеки

та програмного забезпечення

\_\_\_\_\_\_\_\_\_ П.С. Усік

м. Кропивницький 2022

**Мета роботи :**

Полягає у набутті ґрунтовних вмінь і практичних навичок командної (колективної) реалізації програмного забезпечення, розроблення функцій оброблення динамічних структур даних, використання стандартних засобів С++ для керування динамічною пам’яттю та бінарними файловими потоками.

**Завдання :**

1. У складі команди ІТ-проєкта розробити програмні модулі оброблення динамічної структури даних.
2. Реалізувати програмний засіб на основі розроблених командою ІТ-проєкта модулів

**Склад команди IT-проєкта:**

1. Ян Козлов
2. Константин Задорожний
3. Едуард Ісаченко

**План роботи з виконання IT-проєкта:**

1. Завантажити Git-репозиторій - https://github.com/odorenskyi/student-name;
2. В файл README.md, записати: тему, мету, завдання та варіант.
3. Створити теки Report, prj, Software, TestSuite.
4. В текі \lab11\Report створити Документ Microsoft Word «Звіт до Лабораторної роботи № 11», в ньому написати титульну сторінку, тему, мету, завдання, варіант, склад команди.
5. Розробити детальний аналіз отриманих півзадач та проектування. Для розроблення даних півзадач потрібно використати створений на meeting, заголовковий файл struct\_type\_project\_2.h.
6. Розробити свої дві підзадачі та записати їх в модуль «ModulesПрізвище» підключивши заголовковий файл struct\_type\_project\_2.h, вміст файлу включити до звіту.
7. Завантажити фа**Не знайдено елементи покажчика.**йли на GitHub.
8. Коли всі учасники розроблять свої модулі та проведуть модульні тестування, при успішному тестування, завантажте Git-репозиторії інших учасників.
9. В Code::Blocks IDE створити проект консольного додатка prj\_2\_Прізвище.
10. Відповідно до стандарту ISO/IEC 12207 реалізувати програмний засіб згідно з завданням, використовуючи функції (модулі), розроблені учасниками команди ІТ-проекта.
11. Провести системне тестування консольного додатка prj\_2\_Прізвище та свого модуля. При успішному тестування переходи на наступний крок, якщо результат негативний повернутися до додатка, та виправити помилки.
12. Вихідний код (текст) проекта prj\_N\_Прізвище включити до звіту та завантажи на Git-репозиторій.
13. Проаналізувати хід виконання лабораторних завдань і самостійно одержані результати, на основі чого сформулювати обґрунтовані висновки з виконаної лабораторної роботи, викласти їх обсягом не менше двох сторінок машинного (комп’ютерного) тексту та включити до звіту.
14. Підготувати й зберегти у \Lab11\Report звіт про виконання лабораторної роботи, оформлений згідно з ДСТУ 3008:2015 “Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання”, зі змістом, визначеним цим порядком виконання лабораторної роботи.
15. Представити до захисту звіт з виконаної лабораторної роботи і проект у Git-репозиторії https://github.com/odorenskyi/student\_name.

**Аналіз та постановка задачі:**

**Аналіз виконання Task 11:**

На мітингу, який відбувся в Discord, ми з командою розділили завдання для кожного учасника із команди. Я реалізовував функції для додавання так видалення запису. Потрібно щоб при додаванні запису користувач вводив параметри с клавіатури. І повідомляти користувача про успішну додавання персональної карти. При видаленні запису, треба щоб користувач вводив ідентифікаційний номер і якщо такий номер є, то вивести що запис видалено, інакше запис не знайдено.

**Постановка виконання Task 11:**

Вхідні дані:

Функція для додавання запису:

Дані які повинен ввести користувач

Функція для видалення запису:

Ідентифікаційний номер

Вихідні дані:

Функція для додавання запису:

Повідомлення про те що запис додано

Функція для видалення запису:

Повідомлення про видалення запису або що його не знайдено

**Висновок:**

Під час виконання лабораторної роботи № 11 на тему **«**Реалізація програмних модулів оброблення динамічних структур даних та бінарних файлів**»** з предмету «Базові Методології та Технології Програмування» я відпрацював на практиці роботу с бінарними файлами та динамічними структурами даних. Я набув ґрунтовних вмінь і практичних навичок командної (колективної) реалізації програмного забезпечення, розроблення функцій оброблення динамічних структур даних, використання стандартних засобів С++ для керування динамічною пам’яттю та бінарними файловими потоками.

При отриманні лабораторної роботи я дотримувався такого плану:

1. Завантажив свій Git-репозиторій
2. Створити Документ Word, та в нього записати титульну сторінку мету, номер варіанту, завдання та складу команди. Також в
3. В файл README, записати тему, мету, варіант та завдання.
4. Зібралися командою на мітинг, та обговорити:

* Аналіз задачі та вимог до неї.
* Обговорили специфікації ПЗ, концептуальні проектні рішення, сформулювати й узгодити архітектуру програмного засобу, загальні алгоритми функціонування та інтерфейси модулів тощо;
* Визначили вид динамічної структури для реалізації бази даних ПЗ, ми обрали незвичайний тип vector.
* Створили заголовковий файл struct\_type\_project\_2.h, в ньому описали структуру бази даних.
* Розділили задачі між учасниками команди. (Я отримав завдання реалізації додавання персональної картки в базу даних та видалення запису)
* Склали план роботи з виконання ІТ-проєкта, виходячи із розподілених між учасниками команди субзадач та вимог міжнародного стандарту ISO/IEC 12207 щодо виконання процесів реалізації програмних засобів;
* Розроблений план виконання ІТ-проекта затвердити у викладача.

1. Розроблення свого модуля та створення заголовкового файлу
2. Завантаження модуля та заголовкового файлу на github
3. Завантажити з github модулі інших членів команди
4. Розробити консольний додаток реалізований відповідно до стандарту ISO/IEC 12207 для демонстрації роботи с базою даних. Консольний додаток був названий prj\_2\_Zadorozhnyi. Всі модулі були перенесені в папку Modules а заголовкові файли були перенесені в папку h-files
5. Розробити тестові артефакти passed
6. Протестувати застосунок, всі тести було пройдено на позначку
7. Завантажив всі файли на Git-репозиторій.

Закінчивши роботу, я на власному досвіді зрозумів, що командна робота – це весело, та ефективно. Ви розділяєте велику задачу, на маленькі її шматочки, які потім окремо реалізовуєте та потім в кінці роботи їх з’єднуєте в один великий проект.

**Додаток А:**

**Вихідний код заголовкового файлу struct\_type\_project\_2:**

#ifndef STRUCT\_TYPE\_PROJECT\_2\_H\_INCLUDED

#define STRUCT\_TYPE\_PROJECT\_2\_H\_INCLUDED

using namespace std;

struct date{

short day = 0;

short month = 0;

short year = 0;

};

struct entry {

date creationDate;

int regNum = -1;

int id = -1;

char gender = 'm';

char typeOfWork = 'f';

string fullName = "";

date birthDate;

string citizenship = "";

enum {basicSecondary, compSecondary, vocational,

incompHigher, basicHigher, higher} education[4];

string placeOfEducation[4] = {"", "", "", ""};

string lastPlaceOfWork = "";

string lastJob = "";

struct {

short days = 0;

short months = 0;

short years = 0;

} workingExp;

string placeOfLiving = "";

string passportInfo = "";

string additionalInfo = "";

date dateOfFiring;

string reasonOfFiring = "";

};

#endif // STRUCT\_TYPE\_PROJECT\_2\_H\_INCLUDED

**Вихідний код модуля ModulesZadorozhy:**

#include <iostream>

#include <vector>

#include <cstring>

#include <string>

#include <iomanip>

#include <limits>

#include "struct\_type\_project\_2.h"

using namespace std;

void PrintTitle(string title){

cout << title << endl;

cout << ">>> ";

}

void InputString(string title, string &str){

PrintTitle(title);

getline(cin, str);

}

void InputNumber(string title, int &number){

bool iscorrect = true;

string rawNumber = "";

do{

iscorrect = true;

InputString(

title,

rawNumber

);

for(int i = 0; i < rawNumber.size(); i++){

if(rawNumber[i] < '0' || rawNumber[i] > '9'){

cout << "Введено не вірний формат, повторіть спробу ще раз" << endl;

iscorrect = false;

break;

}

}

if(!iscorrect){

continue;

}

number = stoi(rawNumber);

iscorrect = true;

}while(!iscorrect);

}

void InputThreeNumber(short &numOne, short &numTwo, short &numThree, char sign, string title, string errorText){

bool iscorrect = true;

string rawDate = "";

string num = "";

int date[3];

do{

iscorrect = true;

InputString(

title,

rawDate

);

for(int i = 0, dotCount = 0; i < rawDate.size(); i++){

if(rawDate[i] == sign){

dotCount++;

}

else if(rawDate[i] < '0' || rawDate[i] > '9'){

cout << errorText << endl;

iscorrect = false;

break;

}

if(dotCount != 2 && i + 1 == rawDate.size()){

cout << errorText << endl;

iscorrect = false;

break;

}

}

if(!iscorrect){

continue;

}

for(int i = 0, j = 0; i < rawDate.size(); i++){

if(rawDate[i] != sign){

num += rawDate[i];

}

else{

date[j] = stoi(num);

num = "";

j++;

}

if(i+1 == rawDate.size()){

date[j] = stoi(num);

}

}

numOne = date[0];

numTwo = date[1];

numThree = date[2];

iscorrect = true;

}while(!iscorrect);

}

void InputDate(string title, date &creationDate){

bool iscorrect = true;

do{

InputThreeNumber(

creationDate.day,

creationDate.month,

creationDate.year,

'.',

title + "{день.місяць.рік}",

"Дата заповнена неправильно"

);

if((creationDate.day > 31 || creationDate.day <= 0) ||

(creationDate.month > 12 || creationDate.month <= 0) ||

(creationDate.year > 2072 || creationDate.year <= 0)){

cout << "Дата заповнена неправильно" << endl;

iscorrect = false;

continue;

}

iscorrect = true;

}while(!iscorrect);

}

void InputGenderOrTypeWork(string title, char &input, char checkMark, char defaultMark){

string str = "";

InputString(

title,

str

);

if(str[0] == checkMark - 32){

input = str[0] + 32;

}

else{

input = str[0];

}

if(input != checkMark){

input = defaultMark;

}

}

void InputEducation(entry &personalcard){

int numberOfEducation = 0;

int choiseAction = 0;

do{

InputNumber(

"Введіть кількість освіт(максимальна кількість освіт для введення 4)",

numberOfEducation

);

if(numberOfEducation > 4){

cout << "Введіть будь ласка коректне число" << endl;

}

}while(numberOfEducation > 4);

string title = R"(Освіта:

1 - базова середня

2 - повна середня,

3 - професійна,

4 - неповна вища,

5 - базова вища,

6 - вища): )";

for(int i = 0; i < numberOfEducation; i++){

InputNumber(

title,

choiseAction

);

switch(choiseAction){

case 1: personalcard.education[i] = entry::basicSecondary; break;

case 2: personalcard.education[i] = entry::basicHigher; break;

case 3: personalcard.education[i] = entry::vocational; break;

case 4: personalcard.education[i] = entry::incompHigher; break;

case 5: personalcard.education[i] = entry::basicHigher; break;

case 6: personalcard.education[i] = entry::higher; break;

default: cout << "Немає такого варіанту відповіді" << endl; i--; continue;

}

InputString(

"В якому закладі здобували освіту?",

personalcard.placeOfEducation[i]

);

}

}

void InputWorkExp(entry &personalcard){

bool iscorrect = false;

do{

InputThreeNumber(

personalcard.workingExp.days,

personalcard.workingExp.months,

personalcard.workingExp.years,

',',

"Введіть ваш досвід роботи в форматі{день,місяць,рік}",

"Досвід роботи заповнений неправильно"

);

if(personalcard.workingExp.days < 0 ||

personalcard.workingExp.months < 0 ||

personalcard.workingExp.years < 0){

cout << "Введіть ваш досвід роботи в форматі{день місяць рік}" << endl;

continue;

}

iscorrect = true;

}while(!iscorrect);

}

bool YouWorked(){

string action = "";

do{

InputString(

"Ви працювали?(Y/y - так N/n - ні)",

action

);

if(action[0] == 'Y' || action[0] == 'y'){

return true;

}

else if(action[0] == 'N' || action[0] == 'n'){

return false;

}

}while(true);

}

entry InputPersonalCard(){

entry personalcard;

cin.ignore();

InputDate(

"Дата заповнення: ",

personalcard.creationDate

);

InputNumber(

"Табельний номер:",

personalcard.regNum

);

InputNumber(

"Індифікаційний номер:",

personalcard.id);

InputGenderOrTypeWork(

"Стать(m - чоловік, w - жінка):",

personalcard.gender,

'm',

'w'

);

InputGenderOrTypeWork(

"Вид роботи(f - основна, p - за сумісництвом):",

personalcard.typeOfWork,

'f',

'p'

);

InputString(

"ПІБ працівника:",

personalcard.fullName

);

InputDate(

"День народження: ",

personalcard.birthDate

);

InputString(

"Громадянство:",

personalcard.citizenship

);

InputEducation(personalcard);

bool isjob = YouWorked();

if(isjob){

InputString(

"Останнє місце роботи:",

personalcard.lastPlaceOfWork

);

InputString(

"Посада:",

personalcard.lastJob

);

InputWorkExp(personalcard);

}

InputString(

"Місце проживання:",

personalcard.placeOfLiving

);

InputString(

"Паспортні дані:",

personalcard.passportInfo

);

InputString(

"Додаткова інформація:",

personalcard.additionalInfo

);

if(isjob){

InputDate(

"Дата звільнення: ",

personalcard.dateOfFiring

);

InputString(

"Причина звільнення:",

personalcard.reasonOfFiring

);

}

cout << setw(20) << "" << "+———————————————————————————————————————+" << endl;

cout << setw(20) << "" << "| Дані успішно занесено |" << endl;

cout << setw(20) << "" << "+———————————————————————————————————————+" << endl << endl;

return personalcard;

}

void AddPersonalCard(vector<entry> &database){

database.push\_back(InputPersonalCard());

}

void RemovePersonalCard(vector<entry> &database, int id){

auto iter = database.cbegin();

for(int i = 0; i < database.size(); i++){

if(database[i].id == id){

database.erase(iter + i);

cout << setw(20) << "" << "+———————————————————————————————————————+" << endl;

cout << setw(20) << "" << "| Запис видалено |" << endl;

cout << setw(20) << "" << "+———————————————————————————————————————+" << endl << endl;

return;

}

}

cout << setw(20) << "" << "+———————————————————————————————————————+" << endl;

cout << setw(20) << "" << "| Запис не знайдено |" << endl;

cout << setw(20) << "" << "+———————————————————————————————————————+" << endl << endl;

}

**Вихідний код заголовкового файлу ModulesZadorozhny:**

#ifndef MODULESZADOROZHNY\_H\_INCLUDED

#define MODULESZADOROZHNY\_H\_INCLUDED

#include <vector>

#include "struct\_type\_project\_2.h"

using namespace std;

void AddPersonalCard(vector<entry> &database);

void RemovePersonalCard(vector<entry> &database, int id);

#endif // MODULESZADOROZHNY\_H\_INCLUDED

**Вихідний код prj\_2\_Zadorozhnyi**

#include <iostream>

#include <iomanip>

#include <fstream>

#include <string>

#include <cstring>

#include "ModulesKozlov.h"

#include "ModulesZadorozhny.h"

#include "ModulesIsachenkov.h"

int main()

{

system("chcp 1251 && cls");

vector<entry> database;

loadDatabase(database, "database.dat");

do {

int option = 0;

cout << setfill('=') << setw(48) << "|ВІДДІЛ КАДРІВ|" << setw(32) << "" << endl;

cout << setfill(' ') << "| " << setw(77) << left << " Це програмне забезпечення "

"створене для роботи з особовими картками" << "|" << endl;

cout << "| " << setw(77) << left << "працівників ф. П-2. Нижче наведено список "

"опцій, доступних користувачу." << "|" << endl;

cout << "| " << setw(77) << "" << "|" << endl;

cout << "| " << setw(77) << "1. Вивести всю базу даних на екран або у текстовий файл." << "|" << endl;

cout << "| " << setw(77) << "2. Додати новий запис до бази даних." << "|" << endl;

cout << "| " << setw(77) << "3. Пошук запису за прізвищем." << "|" << endl;

cout << "| " << setw(77) << "4. Вилучити запис з бази даних." << "|" << endl;

cout << "| " << setw(77) << "5. Завершити роботу програми." << "|" << endl;

cout << setfill('=') << setw(80) << "" << endl << setfill(' ');

cout << "Введіть номер операції, яку ви бажаєте виконати: " << endl << ">>> ";

if (!(cin >> option)) {

cin.clear();

cin.ignore();

}

cout << resetiosflags(ios::left);

string surname = "";

int id = 0;

switch (option) {

case 1:

showDatabase(database);

break;

case 2:

AddPersonalCard(database);

break;

case 3:

cout << "Введіть прізвище працівника, чию картку ви шукаєте: " << endl << ">>> ";

cin >> surname;

searchPersonalCard(database, surname);

break;

case 4:

cout << "Введіть ідентифікаційний номер картки, яку ви хочете вилучити: " << endl << ">>> ";

cin >> id;

RemovePersonalCard(database, id);

break;

case 5:

saveDatabase(database, "database.dat");

return 0;

default:

continue;

}

} while(true);

}

**Додаток Б:**

**Системне тестування prj\_2\_Zadorozhnyi файл TestSuite №11:**

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TestSuite №11 |
| Назва проекта / ПЗ  Name of Project / Software | prj\_2\_Zadorozhnyi.exe |
| Рівень тестування  Level of Testing | системний / System Testing |
| Автор тест-сьюта  Test Suite Author | Задорожний Костянтин |
| Виконавець  Implementer | Задорожний Костянтин |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест-кейса / Test Case ID | Дії (кроки) /  Action  (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування (пройшов/не вдалося/ заблокований) /  Test Result (passed/failed/ blocked) |
| ТС\_1 | 1. Відкрити *prj\_2\_Zadorozhnyi.exe* 2. Ввести “ 25, 45, 56” 3. Ввести “ 12.03.2022” 4. Ввести “ 45” 5. Ввести “ 38” 6. Ввести “ m” 7. Ввести “ f ” 8. Ввести “ Задорожний Костянтин Олександрович” 9. Ввести “ 22.10.2004” 10. Ввести “ Українець” 11. Ввести “2” 12. Ввести “ 2 ” 13. Ввести “ГНТН” 14. Ввести “ 4” 15. Ввести “ЦНТУ” 16. Ввести “ n ” 17. Ввести “ Кропивницький, Україна” 18. Ввести “ Паспортні дані” 19. Ввести “ Додаткова інформація” | Введіть номер операції, яку ви бажаєте виконати:  >>> 2  Дата заповнення: {день.місяць.рік}  >>> 25, 45, 56  Дата заповнена неправильно  Дата заповнення: {день.місяць.рік}  >>> 12.03.2022  Табельний номер:  >>> 45  Індифікаційний номер:  >>> 38  Стать(m - чоловік, w - жінка):  >>> m  Вид роботи(f - основна, p - за сумісництвом):  >>> f  ПІБ працівника:  >>> Задорожний Костянтин Олександрович  День народження: {день.місяць.рік}  >>> 22.10.2004  Громадянство:  >>> Українець  Введіть кількість освіт(максимальна кількість освіт для введення 4)  >>> 2  Освіта:  1 - базова середня  2 - повна середня,  3 - професійна,  4 - неповна вища,  5 - базова вища,  6 - вища):  >>> 2  В якому закладі здобували освіту?  >>> ГНТН  Освіта:  1 - базова середня  2 - повна середня,  3 - професійна,  4 - неповна вища,  5 - базова вища,  6 - вища):  >>> 4  В якому закладі здобували освіту?  >>> ЦНТУ  Ви працювали?(Y/y - так N/n - ні)  >>> n  Місце проживання:  >>> Кропивницький, Україна  Паспортні дані:  >>> Паспортні дані  Додаткова інформація:  >>> Додаткова інформація | Passed |
| TC\_2 | 1. Відкрити *prj\_2\_Zadorozhnyi.exe* 2. Ввести “ 1” 3. Ввести “ c ” |  | Passed |
| ТС\_3 | 1. Відкрити *prj\_2\_Zadorozhnyi.exe* 2. Ввести “ 1” 3. Ввести “ t ” 4. Ввести «PersonalCard.txt» 5. Ввести “ 5 ” 6. В папці з програмою відкрити файл «PersonalCard.txt». |  | Passed |
| ТС\_4 | 1. Відкрити *prj\_2\_Zadorozhnyi.exe* 2. Ввести “4” 3. Ввести “445” 4. Ввести “4” 5. Ввести “38” |  | Passed |
| ТС\_5 | 1. Відкрити *prj\_2\_Zadorozhnyi.exe* 2. Ввести “ 5 ” |  | Passed |