

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
КІ-22-2

_____ І.М. Карась

ПЕРЕВІРИВ
викладач кафедри
кібербезпеки
та програмного
забезпечення

_____ О.Г. Собінов

Мета роботи :

Полягає у набутті ґрунтовних вмінь і практичних навичок командної (колективної) реалізації програмного забезпечення, розроблення функцій оброблення динамічних структур даних, використання стандартних засобів C++ 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 ІТ-проєкта:

1. Карась Ілля
2. Краснітська Єлизавета
3. Буркун Станіслав

План роботи з виконання ІТ-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 фаНе 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 IT-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.

Завдання:

Базові технології та технології програмування 9 Лабораторна робота № 11

<https://github.com/odorenskyi/Burkun-Stanislaw-KI222>

<https://github.com/odorenskyi/Karas-Illia-KI222>

<https://github.com/odorenskyi/Krasnitska-Yelyzaveta-KB222>

ВАРІАНТ 5

— ЗАВДАННЯ НА РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ —

Створити базу даних “Відділ кадрів: особова картка працівника, ф. П-2” (дата заповнення, табельний номер, ідентифікаційний номер, стать, вид роботи /основна, за сумісництвом/ ПІБ працівника, дата народження, громадянство, освіта, останнє місце роботи і посада, стаж роботи /днів, місяців, років/, місце проживання, паспортні дані, додаткові відомості, дата і причина звільнення).

За вибором користувача (оператора) програма забезпечує:

- виведення всієї бази на екран або у текстовий файл (на вибір користувача);
- додавання нового запису до бази даних;
- пошук запису за введеним прізвищем працівника;
- видалення заданого оператором запису з бази даних;
- завершення роботи програми з автоматичним записом БД у файл.

База даних автоматично завантажується з файлу під час запуску програми.

Найменування підприємства (установи, організації) Код ЄДРПОУ _____					Типова форма П П-2 ЗАТВЕРДЖЕНО наказом Держкомстату та Міністерства оборони України від 25 грудня 2009 р. N 495/656	
Дата заповнення	Табельний номер	Індивідуальний ідентифікаційний номер	Стать (чоловіча, жіноча)	Вид роботи (основна, за сумісництвом)	місце для фотокартки	
ОСОБОВА КАРТКА ПРАЦІВНИКА						
I. ЗАГАЛЬНІ ВІДОМОСТІ						
1. Прізвище _____		Ім'я _____		По батькові _____		
2. Дата народження "___" _____ 19__ р.		3. Громадянство _____				
4. Освіта (базова загальна середня, повна загальна середня, професійно-технічна, неповна вища, базова вища, повна вища)						
Назва освітнього закладу		Диплом (свідчення), серія, номер			Рік закінчення	
Спеціальність (професія) за дипломом (свідченням)		Кваліфікація за дипломом (свідченням)			Форма навчання (денна, вечірня, заочна)	

Аналіз та постановка задачі:

Аналіз виконання Task 11:

На мітингу, який відбувся в Discord, ми з командою розділили завдання для кожного учасника із команди.

Прізвище та ім'я	Функції для реалізації
Краснітьська Ліза	1. Виведення бази даних на екран або у файл. 2. Пошук картки за прізвищем.
Буркун Станіслав	1. Загрузка бази даних. 2. Завершення роботи програми з автоматичним записом бази даних у файл.
Карась Ілля	1. Додавання нової картки в базу даних. 2. Вилучення картки з бази даних.

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 C++ 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-проекта, виходячи із розподілених між учасниками команди субзадач та вимог

міжнародного стандарту ISO/IEC 12207 щодо виконання процесів реалізації програмних засобів;

- Розроблений план виконання IT-проекта затвердити у викладача.
5. Розроблення свого модуля та створення заголовкового файлу
 6. Завантаження модуля та заголовкового файлу на github
 7. Завантажити з github модулі інших членів команди
 8. Розробити консольний додаток реалізований відповідно до стандарту ISO/IEC 12207 для демонстрації роботи з базою даних. Консольний додаток був названий prj_2_Karas. Всі модулі були перенесені в папку Modules а заголовкові файли були перенесені в папку h-files
 9. Розробити тестові артефакти passed
 10. Протестувати застосунок, всі тести було пройдено на позначку passed
 11. Завантажив всі файли на Git-репозиторій.

Під час мітингу, ми вирішили використовувати клас `vector`, так як ми вирішили, що так для нас буде зручніше, і ось чому. Шаблон `vector` розміщений у заголовковому файлі `<vector>`. Як і всі стандартні компоненти, він розміщений у просторі імен `std`. Даний інтерфейс емулює роботу стандартного масиву мови C (наприклад швидкий довільний доступ до елементів), а також деякі додаткові можливості, такі, як автоматична зміни розміру вектора при вставці або видаленні елемента. Та інші ще інші можливості, які надає цей клас.

Під час розробки функції з додаванням персональної картки в базу даних було прийняте рішення розробити додаткову обробку введеної інформації користувачем, для того щоб програма не закінчувала працювати після некоректного введення інформації. Я розробив функції для обробки введення чисел, дати, досвіду роботи та закладу освіти. Також було створено запит на працювала раніше людина чи ні, якщо ні, то всі питання про роботу пропускаються, та заповнюються тільки тире.

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 ModulesKaras:

```
#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 incorrect = true;
    string rawNumber = "";

    do{
        incorrect = 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;
                incorrect = false;
                break;
            }
        }

        if(!incorrect){
            continue;
        }

        number = stoi(rawNumber);
        incorrect = true;
    }while(!incorrect);
}

void InputThreeNumber(short &numOne, short &numTwo, short &numThree, char
sign, string title, string errorText){
    bool incorrect = true;
    string rawDate = "";
    string num = "";
    int date[3];

    do{
        incorrect = 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;
                incorrect = false;
                break;
            }
        }
        if(dotCount != 2 && i + 1 == rawDate.size()){
            cout << errorText << endl;
            incorrect = false;
            break;
        }
    }
}

```

```

        }
    }

    if(!incorrect){
        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];

    incorrect = true;
}while(!incorrect);
}

void InputDate(string title, date &creationDate){
    bool incorrect = 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;
            incorrect = false;
            continue;
        }

        incorrect = true;
    }while(!incorrect);
}

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 incorrect = 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}"
    }

    continue;

}

incorrect = true;

}while(!incorrect);
}

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(
        "Ідентифікаційний номер:",
        personalcard.id);

    InputGenderOrTypeWork(
        "Стать (м - чоловік, ж - жінка):",
        personalcard.gender,

```

```

        'm',
        'w'
    );

    InputGenderOrTypeWork(
        "Вид роботи(f - основна, р - 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 ModulesKaras:

```

#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_Karas

```

#include <iostream>
#include <iomanip>
#include <fstream>
#include <string>
#include <cstring>

#include "ModulesKaras.h"
#include "ModulesKrasnitska.h"
#include "ModulesBurkun.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_Karas файл TestSuite №11:

Назва тестового набору Test Suite Description	TestSuite №11
Назва проекту / ПЗ Name of Project / Software	prj_2_Karas.exe
Рівень тестування Level of Testing	системний / System Testing
Автор тест-сюита Test Suite Author	Карась Ілля
Виконавець Implementer	Карась Ілля

Ід-р тес т- кей са / Test Cas e ID	Дії (кроки) / Action (Test Steps)	Очікуваний результат / Expected Result	Результ ат тестува ння (пройшов/не вд алося/ заблокований) / Test Result (passed/fa iled/ blocked)
ТС _1	1. Відкри ти prj_2_Ka rasi.exe 2. Ввести “ 25, 45, 56” 3. Ввести “ 12.03.2 022” 4. Ввести “ 45”	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	Passed

5. Ввести “ 38”	Ідифікаційний номер: >>> 38	
6. Ввести “ m”	Стать(m - чоловік, w - жінка):	
7. Ввести “ f”	>>> m	
8. Ввести “ Задоро жний Костян тин Олекса ндрови ч”	Вид роботи(f - основна, p - за сумісництвом): >>> f ПІБ працівника: >>> Карась Ілля Миколайович День народження: {день.місяць.рік} >>> 05.01.2005	
9. Ввести “ 22.10.2 004”	Громадянство: >>> Українець	
10.Ввести “ Україн ець”	Введіть кількість освіт(максимальна кількість освіт для введення 4) >>> 2	
11.Ввести “2”	Освіта: 1 - базова середня	
12.Ввести “ 2 ”	2 - повна середня,	
13.Ввести “ГНТН ”	3 - професійна,	
14.Ввести “ 4”	4 - неповна вища,	
15.Ввести “ЦНТ у”	5 - базова вища, 6 - вища): >>> 2	
16.Ввести “ n ”	В якому закладі здобували освіту? >>> ГНТН	
17.Ввести “ Кропи вницьк ий, Україн а”	Освіта: 1 - базова середня 2 - повна середня, 3 - професійна, 4 - неповна вища,	

	<p>18.Ввести “Паспортні дані”</p> <p>19.Ввести “Додаткова інформація”</p>	<p>5 - базова вища, 6 - вища):</p> <p>>>> 4</p> <p>В якому закладі здобували освіту?</p> <p>>>> ЦНТУ</p> <p>Ви працювали?(Y/y - так N/n - ні)</p> <p>>>> n</p> <p>Місце проживання:</p> <p>>>> Кропивницький, Україна</p> <p>Паспортні дані:</p> <p>>>> Паспортні дані</p> <p>Додаткова інформація:</p> <p>>>> Додаткова інформація</p>	
ТС _2	<p>1. Відкрити <i>prj_2_Karas.i.exe</i></p> <p>2. Ввести “ 1”</p> <p>3. Ввести “ с ”</p>	<p>Введіть номер операції, яку ви бажаєте виконати:</p> <p>>>> 1</p> <p>+=====</p> <p>=====+</p> <p>Ви хочете вивести базу даних на консоль (с) або у файл (t) - с</p> <p>+=====</p> <p>=====+</p> <p>+=====</p> <p>=====+</p> <p>Ви хочете вивести базу даних на консоль (с) або у файл (t) - с</p> <p>+=====</p> <p>=====+</p> <p>+_____</p> <p>_____+</p> <p> ВІДДІЛ КАДРІВ: ОСОБОВА КАРТКА ПРАЦІВНИКА, Ф. П-2 </p>	Passed

		<div><div></div><div></div></div>	
		<div><div>1</div><div>Дата створення:</div><div> 11.05.2023</div></div>	
		<div><div></div><div>Реєстраційний номер:</div><div> 45</div></div>	
		<div><div></div><div>Ідентифікаційний номер:</div><div> 38</div></div>	
		<div><div></div><div>Стать:</div><div> чоловіча</div></div>	
		<div><div></div><div>Вид роботи:</div><div> основна</div></div>	
		<div><div></div><div>ПІБ працівника:</div><div> Карась Ілля Миколайович </div></div>	
		<div><div></div><div>Дата народження:</div><div> 05.01.2005 </div></div>	
		<div><div></div><div>Громадянство:</div><div> Українець</div></div>	
		<div><div></div><div>Освіта 1:</div><div> базова вища</div></div>	
		<div><div></div><div></div><div> ГНТН</div></div>	
		<div><div></div><div>Освіта 2:</div><div> неповна вища</div></div>	
		<div><div></div><div></div><div> ЦНТУ</div></div>	
		<div><div></div><div>Останнє місце роботи:</div><div> -</div></div>	
		<div><div></div><div>посада:</div><div> -</div></div>	
		<div><div></div><div>Стаж роботи:</div><div> -</div></div>	
		<div><div></div><div>Місце проживання:</div><div> Кропивницький, Україна</div></div>	
		<div><div></div><div>Паспортні дані:</div><div> Паспортні дані</div></div>	

		<div> <div>Додаткові відомості:</div> <div>Додаткова інформація</div> <div>Дата звільнення:</div> <div>-</div> <div>причина:</div> <div>-</div> </div> <div> <div></div> <div>+</div> </div>	
ТС _3	<div>1. Відкрити prj_2_Karas.exe</div> <div>2. Ввести "1"</div> <div>3. Ввести "t"</div> <div>4. Ввести «PersonalCard.txt»</div> <div>5. Ввести "5"</div> <div>6. В папці з програмою відкрити файл «PersonalCard.txt».</div>	<div>Введіть номер операції, яку ви бажаєте виконати:</div> <div>>>> 1</div> <div>+=====</div> <div>=====+</div> <div>Ви хочете вивести базу даних на консоль (с) або у файл (t) - t</div> <div>+=====</div> <div>=====+</div> <div>Введіть назву файлу: PersonalCard.txt</div> <div>+=====</div> <div>=====+</div>	Passed
ТС _4	<div>1. Відкрити prj_2_Karas.exe</div> <div>2. Ввести "4"</div>	<div>Введіть номер операції, яку ви бажаєте виконати:</div> <div>>>> 4</div> <div>Введіть ідентифікаційний номер картки, яку ви хочете вилучити:</div> <div>>>> 445</div>	Passed

	<div>3. Введіть і "445"</div> <div>4. Введіть і "4"</div> <div>5. Введіть і "38"</div>	<div><div>+-----+</div><div> Запис не знайдено </div></div> <div><div>+-----+</div><div>Введіть номер операції, яку ви бажаєте виконати:</div><div>>>> 4</div><div>Введіть ідентифікаційний номер картки, яку ви хочете вилучити:</div><div>>>> 38</div><div><div>+-----+</div><div> Запис видалено </div><div>+-----+</div></div></div> <div></div>	
<div>ТС_5</div>	<div>1. Відкрити prj_2_Karas.i.exe</div> <div>2. Введіть і " 5 "</div>		<div>Passed</div>