Індувідуальне завдання

#include <stdio.h>

#include <string.h>

#include <conio.h>

#include <stdlib.h>

#define ROWS 0

#define MAXNAME 40

#define MAXSERVICE 20

#define MAXCOOPERATION 30

#define MAXRECREATION 30

#define MAXSITE 10

#define MAXPHONENUMBER 20

#define MAXMAIL 20

#define MAXFAXNUMBER 10

#define MAXADDRESS 10

char nameOfDatabase[20];

FILE \*f;

FILE \*fTemp;

struct travel {

 int ID;

 char name[MAXNAME];

 char cooperation[MAXCOOPERATION];

 char recreation[MAXRECREATION];

 char service[MAXSERVICE];

 char phone\_number[MAXPHONENUMBER];

 char mail[MAXMAIL];

 char site[MAXSITE];

 int founded\_in;

 char fax\_number[MAXFAXNUMBER];

 int rate;

 char address[MAXADDRESS];

};

struct travelM2Edit {

 int ID;

 char name[MAXNAME];

 char cooperation[MAXCOOPERATION];

 char recreation[MAXRECREATION];

 char service[MAXSERVICE];

 char phone\_number[MAXPHONENUMBER];

 char mail[MAXMAIL];

 char site[MAXSITE];

 int founded\_in;

 char fax\_number[MAXFAXNUMBER];

 int rate;

 char address[MAXADDRESS];

 int what;

 int whatEdit;

};

void addElementInDatabase(struct travel addTravel);

void deleteElementInDatabase(int);

void edit(int,int,struct travel addTravel);

void revision();

void search(int,struct travel addTravel);

int main(void) {

 int methodBool=0; //Змінна для перевірки вибору методу.

 int back=1; //Змінна для перевірки чи потрібно виводити головне меню.

 int method; //method - змінна для збереження вибору методу.

 int menuItem,menuExit=7,menuExit2=5;// menuItem - змінна для зберігання вибору пункту головного меню, menuExit - змінна для перевірки виходу з програми.

 int menuItemP; //Змінна для зберігання вибору підменю кожного меню.

 int yes\_no; //Змінна для перевірки вибору меню кожного підменю.

 int backGo; //Змінна для перевірки нажато "0" для виходу в гол. меню.

 int yes\_norate=0; //Змінна для перевірки вибору оцінки гри.

 int ID; //ID - елемента, який потрібно видалити або редагувати.

 char nameOfFile[20]; //Змінна для збереження назви нової бази даних.

 FILE \*fileDatabase; //Мітка для файлу ств. бази даних.

 struct travel workTravel;

 system("cls");

 //\*\*\*Вибір методу роботи з файлом бази даних.

 printf("Choose the method of work with the database:\n\n");

 printf("1. Sequential access to the file.\n");

 printf("2. Access to the files with the batch-processing of information.\n\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&method);

 //\*\*Перевірка вибраний пункт меню чи ні.

 do {

 if (method==1 || method==2) {

 methodBool=1;

 }

 else {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&method);

 }

 } while (methodBool != 1);

 //\*\*Кінець - Перевірка вибраний пункт меню чи ні.

 //\*\*\*Кінець - Вибір методу роботи з файлом бази даних.

 //\*\*\*Перший метод.

 if (method==1) {

 //\*\*\*Перевірка виходу з програми.

 do {

 //\*\*\*Перевірка виводу головного меню.

 if (back==1) {

 system("cls");

 printf(" You chosen a method 1.\n");

 printf(" Menu:\n1. Creation a database.\n2. Insert the element in a database (ID).\n3. Delete of element. \n4. Editing is after the key field. \n5. Revision bases given. \n6. Search in the database. \n7. Close a programm.\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItem);

 }

 //\*\*\*Кінець - Перевірка виводу головного меню.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '1'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

 if (menuItem==1) {

 menuItemP=0;

 printf("\n Do you want to create a new database? The old database will be deleted.\n");

 printf("1. Yes\n2. No\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemP);

 //\*\*Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 else {

 yes\_no=0;

 }

 //\*Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 while (yes\_no!=1) {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItemP);

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 }

 //\*Кінець - Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 //\*\*Кінець - Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1) { //Потрібно створювати базу даних???

 printf("\nEnter name of the file (MAX-20): ");

 scanf("%s",&nameOfFile); //Введення назви файлу бази даних.

 strcat(nameOfFile,".txt"); //Додавання розширення .txt до імені.

 //\*\*Перевірка чи створений файл бази даних чи ні та виведення результату.

 if ((fileDatabase=fopen(nameOfFile,"w")) == NULL) {

 printf("\nAn error is at creation of file. %s.",nameOfFile);

 }

 else {

 printf("\nA file is successfully creation. Name of file %s.",nameOfFile);

 }

 //\*\*Кінець - Перевірка чи створений файл бази даних чи ні та виведення результату.

 //\*Перевірка чи закритий файл бази даних чи ні та виведення результату.

 if (fclose(fileDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfFile);

 }

 else {

 printf("\nA file is successfully closed.");

 }

 //\*Кінець - Перевірка чи закритий файл бази даних чи ні та виведення результату.

 }

 //\*\*Повернення в головне меню.

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1; //Вивід головного меню дозволений.

 }

 //\*\*Кінець - Вибір пункту 1.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '2'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==2) {

 menuItemP=0;

 printf("\n Do you want to insert a new element in a database?\n");

 printf("1. Yes\n2. No\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemP);

 //\*\*Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 else {

 yes\_no=0;

 }

 //\*Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 while (yes\_no!=1) {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItemP);

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 }

 //\*Кінець - Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 //\*\*Кінець - Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1) {

 printf("\nEnter the name database which you want to open (MAX-20): ");

 scanf("%s",&nameOfDatabase); //Введення назви файлу бази даних.

 strcat(nameOfDatabase,".txt"); //Додавання розширення .txt до імені.

 //\*\*\*Введення даних.

 printf("\nEnter next information for a record in a database.");

 printf("\n1. Enter ID of travel agency: ");

 scanf("%d",&workTravel.ID);

 printf("2. Enter name of the travel company (MAX-40): ");

 scanf("%s",&workTravel.name);

 printf("3. Enter service (MAX-10): ");

 scanf("%s",&workTravel.service);

 printf("4. Enter cooperation (MAX-30): ");

 scanf("%s",&workTravel.cooperation);

 printf("5. Enter recreation (MAX-30): ");

 scanf("%s",&workTravel.recreation);

 printf("6. Enter the site (MAX-10): ");

 scanf("%s",&workTravel.site);

 printf("7. Enter phone number (MAX-20): ");

 scanf("%s",&workTravel.phone\_number);

 printf("8. Enter mail (MAX-20): ");

 scanf("%s",&workTravel.mail);

 printf("9. Enter fax number (MAX-10): ");

 scanf("%s",&workTravel.fax\_number);

 printf("10. Enter address (MAX-10): ");

 scanf("%s",&workTravel.address);

 printf("11. Enter founded in (MAX-3): ");

 scanf("%d",&workTravel.founded\_in);

 printf("12. Enter rate (1,2,3,4,5) : ");

 scanf("%d",&workTravel.rate);

 if (workTravel.rate==1 || workTravel.rate==2 || workTravel.rate==3 || workTravel.rate==4 || workTravel.rate== 5) {

 yes\_norate=1;

 }

 while (yes\_norate!=1) {

 printf("You have not enter 1, 2, 3, 4 or 5. Please enter once again: ");

 scanf("%d",&workTravel.rate);

 if (workTravel.rate==1 || workTravel.rate==2 || workTravel.rate==3 || workTravel.rate==4 || workTravel.rate== 5) {

 yes\_norate=1;

 }

 }

 //\*\*\*Кінець - Введення даних.

 //\*\*Вивід даних.

 printf("\n----------------------------------");

 printf("\nYour dates:\n");

 printf("\nID: %d\nservice: %s\nName: %s\ncooperation: %s\nrecreation: %s\nThe site: %s\nPhone number: %s\nMail: %s\nFax number: %s\nAddress: %s\nFounded IN: %d\nRate: %d\n",workTravel.ID,workTravel.service,workTravel.name,workTravel.cooperation,workTravel.recreation,workTravel.site,workTravel.phone\_number,workTravel.mail,workTravel.fax\_number,workTravel.address,workTravel.founded\_in,workTravel.rate);

 printf("\n----------------------------------\n");

 //\*\*Кінець - Вивід даних.

 addElementInDatabase(workTravel);

 }

 //\*\*Повернення в головне меню.

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1;

 }

 //\*\*Кінець - Вибір пункту 2.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '3'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==3) {

 menuItemP=0;

 printf("\n Do you want to delete an element in a database?\n");

 printf("1. Yes\n2. No\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemP);

 //\*\*Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 else {

 yes\_no=0;

 }

 //\*Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 while (yes\_no!=1) {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItemP);

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 }

 //\*Кінець - Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 //\*\*Кінець - Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1) {

 printf("\nEnter the name database which you want to open (MAX-20): ");

 scanf("%s",&nameOfDatabase); //Введення назви файлу бази даних.

 strcat(nameOfDatabase,".txt"); //Додавання розширення .txt до імені.

 //\*\*\*Введення даних.

 printf("\nEnter ID of element to delete: ");

 scanf("%d",&ID);

 deleteElementInDatabase(ID);

 }

 //\*\*Повернення в головне меню.

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1;

 }

 //\*\*Кінець - Вибір пункту 3.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '4'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==4) {

 menuItemP=0;

 printf("\n Do you want to edit an element in a database?\n");

 printf("1. Yes\n2. No\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemP);

 //\*\*Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 else {

 yes\_no=0;

 }

 //\*Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 while (yes\_no!=1) {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItemP);

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 }

 //\*Кінець - Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 //\*\*Кінець - Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1) {

 printf("\nEnter the name database which you want to open (MAX-20): ");

 scanf("%s",&nameOfDatabase); //Введення назви файлу бази даних.

 strcat(nameOfDatabase,".txt"); //Додавання розширення .txt до імені.

 //\*\*\*Введення даних.

 printf("\nEnter ID of element to edit: ");

 scanf("%d",&ID);

 yes\_no=0;

 int menuItemEdit; //Змінна для збереження вибору меню.

 int elemEdit=0; //Який елемент маэ бути відредагований.

 while (yes\_no!=1) {

 system("cls");

 printf("\nWhat you want to edit: ");

 printf("\n1. service\n2. Name\n3. cooperation\n4. recreation\n5. The site\n6. Phone number\n7. Mail\n8. Fax number\n9. Address\n10. Founded in\n11. Rate\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemEdit);

 if (menuItemEdit==1) {

 printf("Enter service (MAX-10): ");

 scanf("%s",&workTravel.service);

 elemEdit=1;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==2) {

 printf("Enter name (MAX-40): ");

 scanf("%s",&workTravel.name);

 elemEdit=2;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==3) {

 printf("Enter cooperation (MAX-30): ");

 scanf("%s",&workTravel.cooperation);

 elemEdit=3;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==4) {

 printf("Enter recreation (MAX-30): ");

 scanf("%s",&workTravel.recreation);

 elemEdit=4;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==5) {

 printf("Enter site (MAX-10): ");

 scanf("%s",&workTravel.site);

 elemEdit=5;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==6) {

 printf("Enter phone number (MAX-20): ");

 scanf("%s",&workTravel.phone\_number);

 elemEdit=6;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==7) {

 printf("Enter mail (MAX-20): ");

 scanf("%s",&workTravel.mail);

 elemEdit=7;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==8) {

 printf("Enter fax number (MAX-10): ");

 scanf("%s",&workTravel.fax\_number);

 elemEdit=8;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==9) {

 printf("Enter address (MAX-10): ");

 scanf("%s",&workTravel.address);

 elemEdit=9;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==10) {

 printf("Enter date : ");

 scanf("%d",&workTravel.founded\_in);

 elemEdit=10;

 edit(ID,elemEdit,workTravel);

 }

 else if (menuItemEdit==11) {

 printf("Enter rate: ");

 scanf("%d",&workTravel.rate);

 elemEdit=11;

 edit(ID,elemEdit,workTravel);

 }

 yes\_no=1;

 }

 }

 //\*\*Повернення в головне меню.

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1;

 }

 //\*\*Кінець - Вибір пункту 4.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '5'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==5) {

 menuItemP=0;

 printf("\n Do you want to revision a database?\n");

 printf("1. Yes\n2. No\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemP);

 //\*\*Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 else {

 yes\_no=0;

 }

 //\*Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 while (yes\_no!=1) {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItemP);

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 }

 //\*Кінець - Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 //\*\*Кінець - Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1) {

 printf("\nEnter the name database which you want to open (MAX-20): ");

 scanf("%s",&nameOfDatabase); //Введення назви файлу бази даних.

 strcat(nameOfDatabase,".txt"); //Додавання розширення .txt до імені.

 //\*\*\*Введення даних.

 revision();

 }

 //\*\*Повернення в головне меню.

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1;

 }

 //\*\*Кінець - Вибір пункту 5.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '6'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==6) {

 menuItemP=0;

 printf("\n Do you want to search in database?\n");

 printf("1. Yes\n2. No\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemP);

 //\*\*Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 else {

 yes\_no=0;

 }

 //\*Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 while (yes\_no!=1) {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItemP);

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 }

 //\*Кінець - Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 //\*\*Кінець - Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1) {

 printf("\nEnter the name database which you want to open (MAX-20): ");

 scanf("%s",&nameOfDatabase); //Введення назви файлу бази даних.

 strcat(nameOfDatabase,".txt"); //Додавання розширення .txt до імені.

 //\*\*\*Введення даних.

 yes\_no=0;

 int menuItemSearch; //Змінна для збереження вибору меню.

 int elemSearch=0; //Який елемент маэ бути відредагований.

 while (yes\_no!=1) {

 system("cls");

 printf("\nWhat you want to edit: ");

 printf("\n1. service\n2. Name\n3. cooperation\n4. recreation\n5. the site\n6. phone\_number\n7. mail\n8. fax\_number\n9. address\n10. founded\_in\n11. rate\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemSearch);

 if (menuItemSearch==1) {

 printf("Enter service (MAX-10): ");

 scanf("%s",&workTravel.service);

 elemSearch=1;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==2) {

 printf("Enter name (MAX-40): ");

 scanf("%s",&workTravel.name);

 elemSearch=2;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==3) {

 printf("Enter cooperation (MAX-30): ");

 scanf("%s",&workTravel.cooperation);

 elemSearch=3;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==4) {

 printf("Enter recreation (MAX-30): ");

 scanf("%s",&workTravel.recreation);

 elemSearch=4;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==5) {

 printf("Enter the site (MAX-10): ");

 scanf("%s",&workTravel.site);

 elemSearch=5;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==6) {

 printf("Enter phone number (MAX-20): ");

 scanf("%s",&workTravel.phone\_number);

 elemSearch=6;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==7) {

 printf("Enter mail (MAX-20): ");

 scanf("%s",&workTravel.mail);

 elemSearch=7;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==8) {

 printf("Enter fax number (MAX-10): ");

 scanf("%s",&workTravel.fax\_number);

 elemSearch=8;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==9) {

 printf("Enter address (MAX-10): ");

 scanf("%s",&workTravel.address);

 elemSearch=9;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==10) {

 printf("Enter year: ");

 scanf("%d",&workTravel.founded\_in);

 elemSearch=10;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==11) {

 printf("Enter rate: ");

 scanf("%d",&workTravel.rate);

 elemSearch=11;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 }

 }

 //\*\*Повернення в головне меню.

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1;

 }

 //\*\*Кінець - Вибір пункту 6.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '7'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==7) {

 printf("Thanks for using a programm. Good bye!!!\n");

 }

 //\*\*Кінець - Вибір пункту 7.

 //\*\*Не вибрано жодного пункту.

 else {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItem);

 back=0;

 }

 //\*\*Кінець - Не вибрано жодного пункту.

 } while (menuItem != menuExit);

 //\*\*\*Кінець - Перевірка виходу з програми.

 }

 //\*\*\*Кінець - Перший метод.

 //??????????????????????????????????????????????????????????????????????????????????????????????

 //Другий метод

 else if (method==2) {

 do {

 //\*\*\*Перевірка виводу головного меню.

 if (back==1) {

 system("cls");

 printf(" You chosen a method 1.\n");

 printf(" Menu:\n1. Creat a new database.\n2. Main funcions. \n3. Revision bases given. \n4. Search in the database. \n5. Close a programm.\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItem);

 }

 //\*\*\*Кінець - Перевірка виводу головного меню.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '1'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

 if (menuItem==1) {

 menuItemP=0;

 printf("\n Do you want to create a new database? The old database will be deleted.\n");

 printf("1. Yes\n2. No\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemP);

 //\*\*Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 else {

 yes\_no=0;

 }

 //\*Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 while (yes\_no!=1) {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItemP);

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 }

 //\*Кінець - Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 //\*\*Кінець - Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1) { //Потрібно створювати базу даних???

 printf("\nEnter name of the file (MAX-20): ");

 scanf("%s",&nameOfFile); //Введення назви файлу бази даних.

 strcat(nameOfFile,".txt"); //Додавання розширення .txt до імені.

 //\*\*Перевірка чи створений файл бази даних чи ні та виведення результату.

 if ((fileDatabase=fopen(nameOfFile,"w")) == NULL) {

 printf("\nAn error is at creation of file. %s.",nameOfFile);

 }

 else {

 printf("\nA file is successfully creation. Name of file %s.",nameOfFile);

 }

 //\*\*Кінець - Перевірка чи створений файл бази даних чи ні та виведення результату.

 //\*Перевірка чи закритий файл бази даних чи ні та виведення результату.

 if (fclose(fileDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfFile);

 }

 else {

 printf("\nA file is successfully closed.");

 }

 //\*Кінець - Перевірка чи закритий файл бази даних чи ні та виведення результату.

 }

 //\*\*Повернення в головне меню.

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1; //Вивід головного меню дозволений.

 }

 //\*\*Кінець - Вибір пункту 1.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '2'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==2) {

 menuItemP=0;

 char words[250];

 int n=0;

 FILE \*fGr;

 printf("\nEnter the name database which you want to open (MAX-20): ");

 scanf("%s",&nameOfDatabase); //Введення назви файлу бази даних.

 strcat(nameOfDatabase,".txt"); //Додавання розширення .txt до імені.

 if ((fileDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 while(fgets(words,250,fileDatabase) != NULL && words[0] != '\n') {

 n++;

 }

 if (fclose(fileDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 if ((f=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 if ((fTemp=fopen("temp.txt","w")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 if ((fGr=fopen("gr.txt","w")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 yes\_no=0;

 int menuItemF; //Змінна для збереження вибору меню.

 int nOp=0; //Змінна для збереження операції, яку потрібно зробити(вставка, редагування, видалення)

 int yes\_norate=0; //Змінна для вибору оцінки гри.

 struct travelM2Edit zenTravel; //Структура потрібна для сортування структур.

 while (yes\_no!=1) {

 system("cls");

 printf("\nWhat you want to do: ");

 printf("\n1. Add an element.\n2. Edit an element.\n3. Delete an element.\n4. Start a modification of database.\n5. Back.\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemF);

 //\*\*\*Введення даних.

 if (menuItemF==1) {

 printf("\nEnter next information for a record in a database.");

 printf("\n1. Enter ID of the travel agency: ");

 scanf("%d",&zenTravel.ID);

 printf("2. Enter service (MAX-10): ");

 scanf("%s",&zenTravel.service);

 printf("3. Enter name (MAX-40): ");

 scanf("%s",&zenTravel.name);

 printf("4. Enter cooperation (MAX-30): ");

 scanf("%s",&zenTravel.cooperation);

 printf("5. Enter recreation (MAX-30): ");

 scanf("%s",&zenTravel.recreation);

 printf("6. Enter the site (MAX-10): ");

 scanf("%s",&zenTravel.site);

 printf("7. Enter phone number (MAX-20): ");

 scanf("%s",&zenTravel.phone\_number);

 printf("8. Enter mail (MAX-20): ");

 scanf("%s",&zenTravel.mail);

 printf("9. Enter fax number (MAX-10): ");

 scanf("%s",&zenTravel.fax\_number);

 printf("10. Enter address (MAX-10): ");

 scanf("%s",&zenTravel.address);

 printf("11. Enter year (MAX-3): ");

 scanf("%d",&zenTravel.founded\_in);

 printf("12. Enter rate (1,2,3,4 or 5): ");

 scanf("%d",&zenTravel.rate);

 if (zenTravel.rate==1 || zenTravel.rate==2 || zenTravel.rate==3 || zenTravel.rate==4 || zenTravel.rate== 5) {

 yes\_norate=1;

 }

 while (yes\_norate!=1) {

 printf("You have not enter 1, 2, 3, 4 or 5. Please enter once again: ");

 scanf("%d",&zenTravel.rate);

 if (zenTravel.rate==1 || zenTravel.rate==2 || zenTravel.rate==3 || zenTravel.rate==4 || zenTravel.rate== 5) {

 yes\_norate=1;

 }

 }

 //\*\*\*Кінець - Введення даних.

 zenTravel.what=1; //Зберігання, що потрібно робити зі структурою.

 //\*\*Вивід даних.

 printf("\n----------------------------------");

 printf("\nYour dates:\n");

 printf("\nID: %d\nservice: %s\nName: %s\ncooperation: %s\nrecreation: %s\nthe site: %s\nphone\_number: %s\nmail: %s\nfax\_number: %s\naddress: %s\nfounded\_in: %d\nrate: %d\nDo: %d\n",zenTravel.ID,zenTravel.service,zenTravel.name,zenTravel.cooperation,zenTravel.recreation,zenTravel.site,zenTravel.phone\_number,zenTravel.mail,zenTravel.fax\_number,zenTravel.address,zenTravel.founded\_in,zenTravel.rate,zenTravel.what);

 printf("\n----------------------------------\n");

 //\*\*Кінець - Вивід даних.

 nOp++;

 fprintf(fGr,"%d %d %s ",zenTravel.what, zenTravel.ID,zenTravel.service);

 fprintf(fGr,"%s %s ",zenTravel.name,zenTravel.cooperation);

 fprintf(fGr,"%s %s ",zenTravel.recreation,zenTravel.site);

 fprintf(fGr,"%s %s ",zenTravel.phone\_number,zenTravel.mail);

 fprintf(fGr,"%s %s ",zenTravel.fax\_number,zenTravel.address);

 fprintf(fGr,"%d %d\n",zenTravel.founded\_in,zenTravel.rate);

 }

 else if (menuItemF==2) {

 printf("\nEnter ID of element to edit: ");

 scanf("%d",&zenTravel.ID);

 zenTravel.what=2;

 fprintf(fGr,"%d %d ",zenTravel.what, zenTravel.ID);

 yes\_no=0;

 int menuItemEdit; //Змінна для збереження вибору меню.

 int yes\_noP=0;

 while (yes\_noP!=1) {

 system("cls");

 printf("%d\n",nOp);

 printf("\nWhat you want to edit: ");

 printf("\n1. Service\n2. Name\n3. Cooperation\n4. Recreation\n5. The site\n6. Phone number\n7. Mail\n8. Fax number\n9. Address\n10. Founded IN\n11. Rate\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemEdit);

 if (menuItemEdit==1) {

 printf("Enter service (MAX-10): ");

 scanf("%s",&zenTravel.service);

 zenTravel.whatEdit=1;

 fprintf(fGr,"%d %s\n",zenTravel.whatEdit, zenTravel.service);

 yes\_noP=1;

 }

 else if (menuItemEdit==2) {

 printf("Enter name (MAX-40): ");

 scanf("%s",&zenTravel.name);

 zenTravel.whatEdit=2;

 fprintf(fGr,"%d %s\n",zenTravel.whatEdit, zenTravel.name);

 yes\_noP=1;

 }

 else if (menuItemEdit==3) {

 printf("Enter cooperation (MAX-30): ");

 scanf("%s",&zenTravel.cooperation);

 zenTravel.whatEdit=3;

 fprintf(fGr,"%d %s\n",zenTravel.whatEdit, zenTravel.cooperation);

 yes\_noP=1;

 }

 else if (menuItemEdit==4) {

 printf("Enter recreation (MAX-30): ");

 scanf("%s",&zenTravel.recreation);

 zenTravel.whatEdit=4;

 fprintf(fGr,"%d %s\n",zenTravel.whatEdit, zenTravel.recreation);

 yes\_noP=1;

 }

 else if (menuItemEdit==5) {

 printf("Enter the site (MAX-10): ");

 scanf("%s",&zenTravel.site);

 zenTravel.whatEdit=5;

 fprintf(fGr,"%d %s\n",zenTravel.whatEdit, zenTravel.site);

 yes\_noP=1;

 }

 else if (menuItemEdit==6) {

 printf("Enter phone number (MAX-20): ");

 scanf("%s",&zenTravel.phone\_number);

 zenTravel.whatEdit=6;

 fprintf(fGr,"%d %s\n",zenTravel.whatEdit, zenTravel.phone\_number);

 yes\_noP=1;

 }

 else if (menuItemEdit==7) {

 printf("Enter mail (MAX-20): ");

 scanf("%s",&zenTravel.mail);

 zenTravel.whatEdit=7;

 fprintf(fGr,"%d %s\n",zenTravel.whatEdit, zenTravel.mail);

 yes\_noP=1;

 }

 else if (menuItemEdit==8) {

 printf("Enter fax number (MAX-10): ");

 scanf("%s",&zenTravel.fax\_number);

 zenTravel.whatEdit=8;

 fprintf(fGr,"%d %s\n",zenTravel.whatEdit, zenTravel.fax\_number);

 yes\_noP=1;

 }

 else if (menuItemEdit==9) {

 printf("Enter address (MAX-10): ");

 scanf("%s",&zenTravel.address);

 zenTravel.whatEdit=9;

 fprintf(fGr,"%d %s\n",zenTravel.whatEdit, zenTravel.address);

 yes\_noP=1;

 }

 else if (menuItemEdit==10) {

 printf("Enter year: ");

 scanf("%d",&zenTravel.founded\_in);

 zenTravel.whatEdit=10;

 fprintf(fGr,"%d %d\n",zenTravel.whatEdit, zenTravel.founded\_in);

 yes\_noP=1;

 }

 else if (menuItemEdit==11) {

 printf("Enter rate: ");

 scanf("%d",&zenTravel.rate);

 zenTravel.whatEdit=11;

 fprintf(fGr,"%d %d\n",zenTravel.whatEdit, zenTravel.rate);

 yes\_noP=1;

 }

 }

 nOp++;

 }

 else if (menuItemF==3) {

 printf("\nEnter ID of element to delete: ");

 scanf("%d",&zenTravel.ID);

 zenTravel.what=3;

 fprintf(fGr,"%d %d\n",zenTravel.what, zenTravel.ID);

 nOp++;

 }

 else if (menuItemF==4) {

 int i,j;

 struct travelM2Edit allTravel[20]; //Масив структур для збереження даних;

 if (fclose(fGr) != 0) {

 printf("\nAn error is at closing of file gr.txt.");

 }

 if ((fGr=fopen("gr.txt","r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 printf("\nnOp = %d\n",nOp);

 for (i=0; i<nOp; i++) {

 char one[40];

 char \*end;

 fscanf(fGr,"%s",&one);

 allTravel[i].what=strtol(one,&end,10);

 fscanf(fGr,"%c");

 if (allTravel[i].what==1) {

 for (j=0;j<12;j++) {

 fscanf(fGr,"%s",&one);

 if (j==0) {

 allTravel[i].ID=strtol(one,&end,10);

 fscanf(fGr,"%c");

 }

 else if (j==1) {

 strcpy(allTravel[i].service,one);

 fscanf(fGr,"%c");

 }

 else if (j==2) {

 strcpy(allTravel[i].name,one);

 fscanf(fGr,"%c");

 }

 else if (j==3) {

 strcpy(allTravel[i].cooperation,one);

 fscanf(fGr,"%c");

 }

 else if (j==4) {

 strcpy(allTravel[i].recreation,one);

 fscanf(fGr,"%c");

 }

 else if (j==5) {

 strcpy(allTravel[i].site,one);

 fscanf(fGr,"%c");

 }

 else if (j==6) {

 strcpy(allTravel[i].phone\_number,one);

 fscanf(fGr,"%c");

 }

 else if (j==7) {

 strcpy(allTravel[i].mail,one);

 fscanf(fGr,"%c");

 }

 else if (j==8) {

 strcpy(allTravel[i].fax\_number,one);

 fscanf(fGr,"%c");

 }

 else if (j==9) {

 strcpy(allTravel[i].address,one);

 fscanf(fGr,"%c");

 }

 else if (j==10) {

 allTravel[i].founded\_in=strtol(one,&end,10);

 fscanf(fGr,"%c");

 }

 else if (j==11) {

 allTravel[i].rate=strtol(one,&end,10);

 }

 }

 }

 else if (allTravel[i].what==2) {

 fscanf(fGr,"%s",&one);

 allTravel[i].ID=strtol(one,&end,10);

 fscanf(fGr,"%c");

 fscanf(fGr,"%s",&one);

 allTravel[i].whatEdit=strtol(one,&end,10);

 fscanf(fGr,"%c");

 fscanf(fGr,"%s",&one);

 if (allTravel[i].whatEdit==0) {

 }

 else if (allTravel[i].whatEdit==1) {

 strcpy(allTravel[i].service,one);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==2) {

 strcpy(allTravel[i].name,one);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==3) {

 strcpy(allTravel[i].cooperation,one);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==4) {

 strcpy(allTravel[i].recreation,one);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==5) {

 strcpy(allTravel[i].site,one);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==6) {

 strcpy(allTravel[i].phone\_number,one);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==7) {

 strcpy(allTravel[i].mail,one);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==8) {

 strcpy(allTravel[i].fax\_number,one);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==9) {

 strcpy(allTravel[i].address,one);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==10) {

 allTravel[i].founded\_in=strtol(one,&end,10);

 fscanf(fGr,"%c");

 }

 else if (allTravel[i].whatEdit==11) {

 allTravel[i].rate=strtol(one,&end,10);

 }

 }

 else if (allTravel[i].what==3) {

 fscanf(fGr,"%s",&one);

 allTravel[i].ID=strtol(one,&end,10);

 }

 }

 if (fclose(fGr) != 0) {

 printf("\nAn error is at closing of file gr.txt.");

 }

 //\*\*\*Сортування

 for (i=0; i<nOp; i++) {

 for (j=0; j<nOp-1; j++) {

 if (allTravel[j].ID>allTravel[j+1].ID) {

 zenTravel.ID=allTravel[j].ID;

 strcpy(zenTravel.service,allTravel[j].service);

 strcpy(zenTravel.name,allTravel[j].name);

 strcpy(zenTravel.cooperation,allTravel[j].cooperation);

 strcpy(zenTravel.recreation,allTravel[j].recreation);

 strcpy(zenTravel.site,allTravel[j].site);

 strcpy(zenTravel.phone\_number,allTravel[j].phone\_number);

 strcpy(zenTravel.mail,allTravel[j].mail);

 strcpy(zenTravel.fax\_number,allTravel[j].fax\_number);

 strcpy(zenTravel.address,allTravel[j].address);

 zenTravel.founded\_in=allTravel[j].founded\_in;

 zenTravel.rate=allTravel[j].rate;

 zenTravel.what=allTravel[j].what;

 zenTravel.whatEdit=allTravel[j].whatEdit;

 allTravel[j].ID=allTravel[j+1].ID;

 strcpy(allTravel[j].service,allTravel[j+1].service);

 strcpy(allTravel[j].name,allTravel[j+1].name);

 strcpy(allTravel[j].cooperation,allTravel[j+1].cooperation);

 strcpy(allTravel[j].recreation,allTravel[j+1].recreation);

 strcpy(allTravel[j].site,allTravel[j+1].site);

 strcpy(allTravel[j].phone\_number,allTravel[j+1].phone\_number);

 strcpy(allTravel[j].mail,allTravel[j+1].mail);

 strcpy(allTravel[j].fax\_number,allTravel[j+1].fax\_number);

 strcpy(allTravel[j].address,allTravel[j+1].address);

 allTravel[j].founded\_in=allTravel[j+1].founded\_in;

 allTravel[j].rate=allTravel[j+1].rate;

 allTravel[j].what=allTravel[j+1].what;

 allTravel[j].whatEdit=allTravel[j+1].whatEdit;

 allTravel[j+1].ID=zenTravel.ID;

 strcpy(allTravel[j+1].service,zenTravel.service);

 strcpy(allTravel[j+1].name,zenTravel.name);

 strcpy(allTravel[j+1].cooperation,zenTravel.cooperation);

 strcpy(allTravel[j+1].recreation,zenTravel.recreation);

 strcpy(allTravel[j+1].site,zenTravel.site);

 strcpy(allTravel[j+1].phone\_number,zenTravel.phone\_number);

 strcpy(allTravel[j+1].mail,zenTravel.mail);

 strcpy(allTravel[j+1].fax\_number,zenTravel.fax\_number);

 strcpy(allTravel[j+1].address,zenTravel.address);

 allTravel[j+1].founded\_in=zenTravel.founded\_in;

 allTravel[j+1].rate=zenTravel.rate;

 allTravel[j+1].what=zenTravel.what;

 allTravel[j+1].whatEdit=zenTravel.whatEdit;

 }

 }

 }

 //\*\*Кінець - Сортування структур.

 //\*\*\*Вивід даних, якими потрібно модифікувати базу даних.

 int m;

 if ((fGr=fopen("gr.txt","w")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 if (nOp>0) {

 for (m=0; m<nOp; m++) {

 if (allTravel[m].what==1) { //Додавання запису.

 printf("\n%d. Add a record:\n",m);

 //\*\*Вивід даних.

 printf("\n----------------------------------");

 printf("\nYour dates:\n");

 printf("\nID: %d\nService: %s\nName: %s\nCooperation: %s\nRecreation: %s\nThe site: %s\nPhone pumber: %s\nMail: %s\nFax number: %s\nAddress: %s\nFounded in: %d\nRate: %d\n",allTravel[m].ID,allTravel[m].service,allTravel[m].name,allTravel[m].cooperation,allTravel[m].recreation,allTravel[m].site,allTravel[m].phone\_number,allTravel[m].mail,allTravel[m].fax\_number,allTravel[m].address,allTravel[m].founded\_in,allTravel[m].rate);

 printf("\n----------------------------------\n");

 fprintf(fGr,"%d %d %s ",allTravel[m].ID,allTravel[m].what,allTravel[m].service);

 fprintf(fGr,"%s %s ",allTravel[m].name,allTravel[m].cooperation);

 fprintf(fGr,"%s %s ",allTravel[m].recreation,allTravel[m].site);

 fprintf(fGr,"%s %s ",allTravel[m].phone\_number,allTravel[m].mail);

 fprintf(fGr,"%s %s ",allTravel[m].fax\_number,allTravel[m].address);

 fprintf(fGr,"%d %d\n",allTravel[m].founded\_in,allTravel[m].rate);

 //\*\*Кінець - Вивід даних.

 }

 else if (allTravel[m].what==2) {//Редагування запису.

 printf("\n%d. Edit a record:\n",m);

 printf("\nID: %d\n",allTravel[m].ID);

 fprintf(fGr,"%d %d ",allTravel[m].ID,allTravel[m].what);

 if (allTravel[m].whatEdit==1) {

 printf("\nservice: %s\n",allTravel[m].service);

 fprintf(fGr,"%d %s\n",allTravel[m].whatEdit,allTravel[m].service);

 }

 else if (allTravel[m].whatEdit==2) {

 printf("\nName: %s\n",allTravel[m].name);

 fprintf(fGr,"%d %s\n",allTravel[m].whatEdit,allTravel[m].name);

 }

 else if (allTravel[m].whatEdit==3) {

 printf("\ncooperation: %s\n",allTravel[m].cooperation);

 fprintf(fGr,"%d %s\n",allTravel[m].whatEdit,allTravel[m].cooperation);

 }

 else if (allTravel[m].whatEdit==4) {

 printf("\nrecreation: %s\n",allTravel[m].recreation);

 fprintf(fGr,"%d %s\n",allTravel[m].whatEdit,allTravel[m].recreation);

 }

 else if (allTravel[m].whatEdit==5) {

 printf("\nsite: %s\n",allTravel[m].site);

 fprintf(fGr,"%d %s\n",allTravel[m].whatEdit,allTravel[m].site);

 }

 else if (allTravel[m].whatEdit==6) {

 printf("\nphone\_number: %s\n",allTravel[m].phone\_number);

 fprintf(fGr,"%d %s\n",allTravel[m].whatEdit,allTravel[m].phone\_number);

 }

 else if (allTravel[m].whatEdit==7) {

 printf("\nmail: %s\n",allTravel[m].mail);

 fprintf(fGr,"%d %s\n",allTravel[m].whatEdit,allTravel[m].mail);

 }

 else if (allTravel[m].whatEdit==8) {

 printf("\nfax\_number: %s\n",allTravel[m].fax\_number);

 fprintf(fGr,"%d %s\n",allTravel[m].whatEdit,allTravel[m].fax\_number);

 }

 else if (allTravel[m].whatEdit==9) {

 printf("\naddress: %s\n",allTravel[m].address);

 fprintf(fGr,"%d %s\n",allTravel[m].whatEdit,allTravel[m].address);

 }

 else if (allTravel[m].whatEdit==10) {

 printf("\nfounded\_in: %d\n",allTravel[m].founded\_in);

 fprintf(fGr,"%d %d\n",allTravel[m].whatEdit,allTravel[m].founded\_in);

 }

 else if (allTravel[m].whatEdit==11) {

 printf("\nrate: %d\n",allTravel[m].rate);

 fprintf(fGr,"%d %d\n",allTravel[m].whatEdit,allTravel[m].rate);

 }

 }

 else if (allTravel[m].what==3) { //Видалення запису.

 printf("\n%d. Delete a record:\n",m);

 printf("\nID: %d\n",allTravel[m].ID);

 fprintf(fGr,"%d %d\n",allTravel[m].ID,allTravel[m].what);

 }

 }

 }

 if (fclose(fGr) != 0) {

 printf("\nAn error is at closing of file gr.txt.");

 }

 //\*\*\*Кінець - Вивід даних, якими потрібно модифікувати базу даних.

 int i\_vn=0; //Змінна для зберігання скільки записів додано, редаговано, видалено.

 for (m=0; m<n; m++) {

 char one[40];

 char \*end;

 //Зчитування з файлу запису.

 for (j=0;j<12;j++) {

 fscanf(f,"%s",&one);

 if (j==0) {

 zenTravel.ID=strtol(one,&end,10);

 fscanf(f,"%c");

 }

 else if (j==1) {

 strcpy(zenTravel.service,one);

 fscanf(f,"%c");

 }

 else if (j==2) {

 strcpy(zenTravel.name,one);

 fscanf(f,"%c");

 }

 else if (j==3) {

 strcpy(zenTravel.cooperation,one);

 fscanf(f,"%c");

 }

 else if (j==4) {

 strcpy(zenTravel.recreation,one);

 fscanf(f,"%c");

 }

 else if (j==5) {

 strcpy(zenTravel.site,one);

 fscanf(f,"%c");

 }

 else if (j==6) {

 strcpy(zenTravel.phone\_number,one);

 fscanf(f,"%c");

 }

 else if (j==7) {

 strcpy(zenTravel.mail,one);

 fscanf(f,"%c");

 }

 else if (j==8) {

 strcpy(zenTravel.fax\_number,one);

 fscanf(f,"%c");

 }

 else if (j==9) {

 strcpy(zenTravel.address,one);

 fscanf(f,"%c");

 }

 else if (j==10) {

 zenTravel.founded\_in=strtol(one,&end,10);

 fscanf(f,"%c");

 }

 else if (j==11) {

 zenTravel.rate=strtol(one,&end,10);

 }

 }

 //Кінець - Зчитування з файлу запису.

 //\*\*\*Модифікація бази даних.

 if (allTravel[i\_vn].ID<=zenTravel.ID) { //Якщо ID менший або рівний, інакше записати в temp.

 if (i\_vn<nOp) { //Всі записи пройдені, інакше записати в temp.

 printf("\ni\_vn = %d",i\_vn);

 //Якщо 3 ID рівні.

 if (allTravel[i\_vn].ID==allTravel[i\_vn+1].ID && allTravel[i\_vn].ID==allTravel[i\_vn+2].ID) {

 if (allTravel[i\_vn+2].what==3) {

 printf("\nAdd, edit, delete. Nichoho ne potribno robutu");

 }

 i\_vn=i\_vn+3; //Пройдені 3 записи.

 }

 //Якщо 2 ID рівні.

 else if (allTravel[i\_vn].ID==allTravel[i\_vn+1].ID) {

 //Додати редагувати??

 if (allTravel[i\_vn].what==1 && allTravel[i\_vn+1].what==2) {

 printf("\nAdd, edit.");

 if (allTravel[i\_vn+1].whatEdit==1) {

 strcpy(allTravel[i\_vn].service,allTravel[i\_vn+1].service);

 }

 else if (allTravel[i\_vn+1].whatEdit==2) {

 strcpy(allTravel[i\_vn].name,allTravel[i\_vn+1].name);

 }

 else if (allTravel[i\_vn+1].whatEdit==3) {

 strcpy(allTravel[i\_vn].cooperation,allTravel[i\_vn+1].cooperation);

 }

 else if (allTravel[i\_vn+1].whatEdit==4) {

 strcpy(allTravel[i\_vn].recreation,allTravel[i\_vn+1].recreation);

 }

 else if (allTravel[i\_vn+1].whatEdit==5) {

 strcpy(allTravel[i\_vn].site,allTravel[i\_vn+1].site);

 }

 else if (allTravel[i\_vn+1].whatEdit==6) {

 strcpy(allTravel[i\_vn].phone\_number,allTravel[i\_vn+1].phone\_number);

 }

 else if (allTravel[i\_vn+1].whatEdit==7) {

 strcpy(allTravel[i\_vn].mail,allTravel[i\_vn+1].mail);

 }

 else if (allTravel[i\_vn+1].whatEdit==8) {

 strcpy(allTravel[i\_vn].fax\_number,allTravel[i\_vn+1].fax\_number);

 }

 else if (allTravel[i\_vn+1].whatEdit==9) {

 strcpy(allTravel[i\_vn].address,allTravel[i\_vn+1].address);

 }

 else if (allTravel[i\_vn+1].whatEdit==10) {

 allTravel[i\_vn].founded\_in=allTravel[i\_vn+1].founded\_in;

 }

 else if (allTravel[i\_vn+1].whatEdit==11) {

 allTravel[i\_vn].rate=allTravel[i\_vn+1].rate;

 }

 fprintf(fTemp,"%d %s ",allTravel[i\_vn].ID,allTravel[i\_vn].service);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].name,allTravel[i\_vn].cooperation);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].recreation,allTravel[i\_vn].site);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].phone\_number,allTravel[i\_vn].mail);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].fax\_number,allTravel[i\_vn].address);

 fprintf(fTemp,"%d %d\n",allTravel[i\_vn].founded\_in,allTravel[i\_vn].rate);

 fprintf(fTemp,"%d %s ",zenTravel.ID,zenTravel.service);

 fprintf(fTemp,"%s %s ",zenTravel.name,zenTravel.cooperation);

 fprintf(fTemp,"%s %s ",zenTravel.recreation,zenTravel.site);

 fprintf(fTemp,"%s %s ",zenTravel.phone\_number,zenTravel.mail);

 fprintf(fTemp,"%s %s ",zenTravel.fax\_number,zenTravel.address);

 fprintf(fTemp,"%d %d\n",zenTravel.founded\_in,zenTravel.rate);

 }

 //Додати і видалити??

 else if (allTravel[i\_vn].what==1 && allTravel[i\_vn+1].what==3) {

 printf("\nAdd, delete: nichoho ne potribno robutu");

 }

 //Редагувати і видалити??

 else if (allTravel[i\_vn].what==2 && allTravel[i\_vn+1].what==3) {

 printf("\nEdit, delete: nichoho ne potribno robutu");

 }

 i\_vn=i\_vn+2; //2 записи пройдені.

 }

 else {

 if (allTravel[i\_vn].ID<zenTravel.ID) { //ID менший???

 if (allTravel[i\_vn].what==1) { //Додати запис??

 printf("\nAdd element.");

 fprintf(fTemp,"%d %s ",allTravel[i\_vn].ID,allTravel[i\_vn].service);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].name,allTravel[i\_vn].cooperation);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].recreation,allTravel[i\_vn].site);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].phone\_number,allTravel[i\_vn].mail);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].fax\_number,allTravel[i\_vn].address);

 fprintf(fTemp,"%d %d\n",allTravel[i\_vn].founded\_in,allTravel[i\_vn].rate);

 fprintf(fTemp,"%d %s ",zenTravel.ID,zenTravel.service);

 fprintf(fTemp,"%s %s ",zenTravel.name,zenTravel.cooperation);

 fprintf(fTemp,"%s %s ",zenTravel.recreation,zenTravel.site);

 fprintf(fTemp,"%s %s ",zenTravel.phone\_number,zenTravel.mail);

 fprintf(fTemp,"%s %s ",zenTravel.fax\_number,zenTravel.address);

 fprintf(fTemp,"%d %d\n",zenTravel.founded\_in,zenTravel.rate);

 }

 else if (allTravel[i\_vn].what==2 || allTravel[i\_vn].what==3) {

 printf("\nSuch element is not in a database.");

 }

 }

 else if (allTravel[i\_vn].ID==zenTravel.ID) {

 if (allTravel[i\_vn].what==2) { //Редагувати запис??

 printf("\nEdit element.");

 if (allTravel[i\_vn].whatEdit==1) {

 strcpy(zenTravel.service,allTravel[i\_vn].service);

 }

 else if (allTravel[i\_vn].whatEdit==2) {

 strcpy(zenTravel.name,allTravel[i\_vn].name);

 }

 else if (allTravel[i\_vn].whatEdit==3) {

 strcpy(zenTravel.cooperation,allTravel[i\_vn].cooperation);

 }

 else if (allTravel[i\_vn].whatEdit==4) {

 strcpy(zenTravel.recreation,allTravel[i\_vn].recreation);

 }

 else if (allTravel[i\_vn].whatEdit==5) {

 strcpy(zenTravel.site,allTravel[i\_vn].site);

 }

 else if (allTravel[i\_vn].whatEdit==6) {

 strcpy(zenTravel.phone\_number,allTravel[i\_vn].phone\_number);

 }

 else if (allTravel[i\_vn].whatEdit==7) {

 strcpy(zenTravel.mail,allTravel[i\_vn].mail);

 }

 else if (allTravel[i\_vn].whatEdit==8) {

 strcpy(zenTravel.fax\_number,allTravel[i\_vn].fax\_number);

 }

 else if (allTravel[i\_vn].whatEdit==9) {

 strcpy(zenTravel.address,allTravel[i\_vn].address);

 }

 else if (allTravel[i\_vn].whatEdit==10) {

 zenTravel.founded\_in=allTravel[i\_vn].founded\_in;

 }

 else if (allTravel[i\_vn].whatEdit==11) {

 zenTravel.rate=allTravel[i\_vn].rate;

 }

 fprintf(fTemp,"%d %s ",zenTravel.ID,zenTravel.service);

 fprintf(fTemp,"%s %s ",zenTravel.name,zenTravel.cooperation);

 fprintf(fTemp,"%s %s ",zenTravel.recreation,zenTravel.site);

 fprintf(fTemp,"%s %s ",zenTravel.phone\_number,zenTravel.mail);

 fprintf(fTemp,"%s %s ",zenTravel.fax\_number,zenTravel.address);

 fprintf(fTemp,"%d %d\n",zenTravel.founded\_in,zenTravel.rate);

 }

 else if (allTravel[i\_vn].what==3) { //Видалити запис??

 printf("\nDelete element.");

 }

 }

 i\_vn++; //1 запис пройдено.

 }

 }

 else {

 fprintf(fTemp,"%d %s ",zenTravel.ID,zenTravel.service);

 fprintf(fTemp,"%s %s ",zenTravel.name,zenTravel.cooperation);

 fprintf(fTemp,"%s %s ",zenTravel.recreation,zenTravel.site);

 fprintf(fTemp,"%s %s ",zenTravel.phone\_number,zenTravel.mail);

 fprintf(fTemp,"%s %s ",zenTravel.fax\_number,zenTravel.address);

 fprintf(fTemp,"%d %d\n",zenTravel.founded\_in,zenTravel.rate);

 }

 }

 else {

 fprintf(fTemp,"%d %s ",zenTravel.ID,zenTravel.service);

 fprintf(fTemp,"%s %s ",zenTravel.name,zenTravel.cooperation);

 fprintf(fTemp,"%s %s ",zenTravel.recreation,zenTravel.site);

 fprintf(fTemp,"%s %s ",zenTravel.phone\_number,zenTravel.mail);

 fprintf(fTemp,"%s %s ",zenTravel.fax\_number,zenTravel.address);

 fprintf(fTemp,"%d %d\n",zenTravel.founded\_in,zenTravel.rate);

 }

 }

 //\*\*Якщо файл пустий, то додати в кінець файла.

 if (zenTravel.ID<allTravel[i\_vn].ID) {

 printf("\nAdd element.");

 fprintf(fTemp,"%d %s ",allTravel[i\_vn].ID,allTravel[i\_vn].service);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].name,allTravel[i\_vn].cooperation);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].recreation,allTravel[i\_vn].site);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].phone\_number,allTravel[i\_vn].mail);

 fprintf(fTemp,"%s %s ",allTravel[i\_vn].fax\_number,allTravel[i\_vn].address);

 fprintf(fTemp,"%d %d\n",allTravel[i\_vn].founded\_in,allTravel[i\_vn].rate);

 }

 //\*\*Кінець - Якщо файл пустий, то додати в кінець файла.

 //\*\*\*Кінець - Модифікація бази даних.

 //\*Перевірка чи закритий файл бази даних чи ні та виведення результату.

 if (fclose(f) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 if (fclose(fTemp) != 0) {

 printf("\nAn error is at closing of file temp.txt.");

 }

 //\*Кінець - Перевірка чи закритий файл бази даних чи ні та виведення результату.

 remove(nameOfDatabase);

 rename("temp.txt",nameOfDatabase);

 nOp=0;

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1;

 }

 else if (menuItemF==5) {

 yes\_no=1;

 }

 }

 //\*\*Повернення в головне меню.

 }

 //\*\*Кінець - Вибір пункту 2.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '3'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==3) {

 menuItemP=0;

 printf("\n Do you want to revision a database?\n");

 printf("1. Yes\n2. No\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemP);

 //\*\*Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 else {

 yes\_no=0;

 }

 //\*Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 while (yes\_no!=1) {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItemP);

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 }

 //\*Кінець - Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 //\*\*Кінець - Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1) {

 printf("\nEnter the name database which you want to open (MAX-20): ");

 scanf("%s",&nameOfDatabase); //Введення назви файлу бази даних.

 strcat(nameOfDatabase,".txt"); //Додавання розширення .txt до імені.

 //\*\*\*Введення даних.

 revision();

 }

 //\*\*Повернення в головне меню.

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1;

 }

 //\*\*Кінець - Вибір пункту 3.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '4'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==4) {

 menuItemP=0;

 printf("\n Do you want to search in database?\n");

 printf("1. Yes\n2. No\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemP);

 //\*\*Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 else {

 yes\_no=0;

 }

 //\*Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 while (yes\_no!=1) {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItemP);

 if (menuItemP==1 || menuItemP==2) {

 yes\_no=1;

 }

 }

 //\*Кінець - Перевірка чи потрібно виконувати цикл, якщо не вибраний пункт підменю.

 //\*\*Кінець - Перевірка вибраний пункт підменю головного меню чи ні.

 if (menuItemP==1) {

 printf("\nEnter the name database which you want to open (MAX-20): ");

 scanf("%s",&nameOfDatabase); //Введення назви файлу бази даних.

 strcat(nameOfDatabase,".txt"); //Додавання розширення .txt до імені.

 //\*\*\*Введення даних.

 yes\_no=0;

 int menuItemSearch; //Змінна для збереження вибору меню.

 int elemSearch=0; //Який елемент маэ бути відредагований.

 while (yes\_no!=1) {

 system("cls");

 printf("\nWhat you want to edit: ");

 printf("\n1. Service\n2. Name\n3. Cooperation\n4. Recreation\n5. The site\n6. Phone number\n7. Mail\n8. Fax number\n9. Address\n10. Founded IN\n11. Rate\n");

 printf("Choose and enter one of menu items: ");

 scanf("%d",&menuItemSearch);

 if (menuItemSearch==1) {

 printf("Enter service (MAX-10): ");

 scanf("%s",&workTravel.service);

 elemSearch=1;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==2) {

 printf("Enter name (MAX-40): ");

 scanf("%s",&workTravel.name);

 elemSearch=2;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==3) {

 printf("Enter cooperation (MAX-30): ");

 scanf("%s",&workTravel.cooperation);

 elemSearch=3;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==4) {

 printf("Enter recreation (MAX-30): ");

 scanf("%s",&workTravel.recreation);

 elemSearch=4;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==5) {

 printf("Enter the site (MAX-10): ");

 scanf("%s",&workTravel.site);

 elemSearch=5;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==6) {

 printf("Enter phone number (MAX-20): ");

 scanf("%s",&workTravel.phone\_number);

 elemSearch=6;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==7) {

 printf("Enter mail (MAX-20): ");

 scanf("%s",&workTravel.mail);

 elemSearch=7;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==8) {

 printf("Enter fax number (MAX-10): ");

 scanf("%s",&workTravel.fax\_number);

 elemSearch=8;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==9) {

 printf("Enter address (MAX-10): ");

 scanf("%s",&workTravel.address);

 elemSearch=9;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==10) {

 printf("Enter year: ");

 scanf("%d",&workTravel.founded\_in);

 elemSearch=10;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 else if (menuItemSearch==11) {

 printf("Enter rate: ");

 scanf("%d",&workTravel.rate);

 elemSearch=11;

 search(elemSearch,workTravel);

 yes\_no=1;

 }

 }

 }

 //\*\*Повернення в головне меню.

 printf("\nEnter '0' for returning in a main menu: ");

 scanf("%d",&backGo);

 while (backGo!=0) {

 printf("You have not enter '0'. Please enter once again: ");

 scanf("%d",&backGo);

 }

 //\*\*Кінець - Повернення в головне меню.

 back=1;

 }

 //\*\*Кінець - Вибір пункту 4.

 //\*\*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ВИБІР ПУНКТУ '5'.@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@.

 else if (menuItem==5) {

 printf("Thanks for using a programm. Good bye!!!\n");

 }

 //\*\*Кінець - Вибір пункту 5.

 //\*\*Не вибрано жодного пункту.

 else {

 printf("You have not chosen any menu item. Please enter once again: ");

 scanf("%d",&menuItem);

 back=0;

 }

 //\*\*Кінець - Не вибрано жодного пункту.

 } while (menuItem != menuExit2);

 //\*\*\*Кінець - Перевірка виходу з програми.

 }

 //Кінець - Другий метод

 return 0;

}

//@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

//@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ ФУНКЦІЇ @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

//@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

//--------------------------------------------------

//\*\*\*Функція додавання нового елемента в базу даних.

void addElementInDatabase(struct travel addTravel) {

 FILE \*fileOldDatabase; //Мітка для файлу старої (створеної) бази даних.

 FILE \*fileTemp; //Мітка для файлу тимчасової бази даних.

 char words[250]; //Масив символів для зберігання стрічки, яку зчитуєм з файлу.

 int i=0,n=0,k,j; //i - Змінна для перевірки чи добавлений новий запис чи ні, n - к-ть записів.

 char \*end; //Вказівник для закінчення визначення цифр зі стрічки.

 struct travel oldTravel; //Структура для зберігання запису з бази даних.

 //\*\*\*Перевірка чи файл відкритий без помилок і вивід резульату.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 while(fgets(words,250,fileOldDatabase) != NULL && words[0] != '\n') {

 n++;

 }

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 //\*\*\*Кінець - Перевірка чи файл відкритий без помилок і вивід резульату.

 //\*\*\*Ввід запису в базу даних.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 if ((fileTemp=fopen("temp.txt","w")) == NULL) {

 printf("\nAn error is at opening of database temp.txt.");

 }

 i=1; //Запис не доданий.

 for (k=0;k<n;k++) {

 char one[40]; //Масив для збереження зчитуваного елемента.

 for (j=0;j<12;j++) { //Цикл по к-ті елементів.

 fscanf(fileOldDatabase,"%s",&one); //Зчитування з файлу перший елемент.

 //Збереження елементів в структуру.

 if (j==0) {

 oldTravel.ID=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==1) {

 strcpy(oldTravel.service,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==2) {

 strcpy(oldTravel.name,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==3) {

 strcpy(oldTravel.cooperation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==4) {

 strcpy(oldTravel.recreation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==5) {

 strcpy(oldTravel.site,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==6) {

 strcpy(oldTravel.phone\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==7) {

 strcpy(oldTravel.mail,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==8) {

 strcpy(oldTravel.fax\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==9) {

 strcpy(oldTravel.address,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==10) {

 oldTravel.founded\_in=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==11) {

 oldTravel.rate=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 //Кінець - Збереження елементів в структуру.

 }

 if (addTravel.ID<oldTravel.ID && i==1) { //Якщо ID запису, який потрібно додати менший від того що в базі даних.

 //Дадання нового запису.

 fprintf(fileTemp,"%d %s ",addTravel.ID,addTravel.service);

 fprintf(fileTemp,"%s %s ",addTravel.name,addTravel.cooperation);

 fprintf(fileTemp,"%s %s ",addTravel.recreation,addTravel.site);

 fprintf(fileTemp,"%s %s ",addTravel.phone\_number,addTravel.mail);

 fprintf(fileTemp,"%s %s ",addTravel.fax\_number,addTravel.address);

 fprintf(fileTemp,"%d %d\n",addTravel.founded\_in,addTravel.rate);

 //Кінець - Дадання нового запису.

 fprintf(fileTemp,"%d %s ",oldTravel.ID,oldTravel.service);

 fprintf(fileTemp,"%s %s ",oldTravel.name,oldTravel.cooperation);

 fprintf(fileTemp,"%s %s ",oldTravel.recreation,oldTravel.site);

 fprintf(fileTemp,"%s %s ",oldTravel.phone\_number,oldTravel.mail);

 fprintf(fileTemp,"%s %s ",oldTravel.fax\_number,oldTravel.address);

 fprintf(fileTemp,"%d %d\n",oldTravel.founded\_in,oldTravel.rate);

 i++; //Запич доданий

 }

 else if (addTravel.ID==oldTravel.ID) {

 printf("Such record in a database is already. It will not be added.");

 fprintf(fileTemp,"%d %s ",oldTravel.ID,oldTravel.service);

 fprintf(fileTemp,"%s %s ",oldTravel.name,oldTravel.cooperation);

 fprintf(fileTemp,"%s %s ",oldTravel.recreation,oldTravel.site);

 fprintf(fileTemp,"%s %s ",oldTravel.phone\_number,oldTravel.mail);

 fprintf(fileTemp,"%s %s ",oldTravel.fax\_number,oldTravel.address);

 fprintf(fileTemp,"%d %d\n",oldTravel.founded\_in,oldTravel.rate);

 i++;

 }

 else {

 fprintf(fileTemp,"%d %s ",oldTravel.ID,oldTravel.service);

 fprintf(fileTemp,"%s %s ",oldTravel.name,oldTravel.cooperation);

 fprintf(fileTemp,"%s %s ",oldTravel.recreation,oldTravel.site);

 fprintf(fileTemp,"%s %s ",oldTravel.phone\_number,oldTravel.mail);

 fprintf(fileTemp,"%s %s ",oldTravel.fax\_number,oldTravel.address);

 fprintf(fileTemp,"%d %d\n",oldTravel.founded\_in,oldTravel.rate);

 }

 }

 if (i==1) { //Якщо запис не доданий, його в кінець бази даних.

 fprintf(fileTemp,"%d %s ",addTravel.ID,addTravel.service);

 fprintf(fileTemp,"%s %s ",addTravel.name,addTravel.cooperation);

 fprintf(fileTemp,"%s %s ",addTravel.recreation,addTravel.site);

 fprintf(fileTemp,"%s %s ",addTravel.phone\_number,addTravel.mail);

 fprintf(fileTemp,"%s %s ",addTravel.fax\_number,addTravel.address);

 fprintf(fileTemp,"%d %d",addTravel.founded\_in,addTravel.rate);

 }

 //\*Перевірка чи закритий файл бази даних чи ні та виведення результату.

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 if (fclose(fileTemp) != 0) {

 printf("\nAn error is at closing of file temp.txt.");

 }

 //\*Кінець - Перевірка чи закритий файл бази даних чи ні та виведення результату.

 remove(nameOfDatabase);

 rename("temp.txt",nameOfDatabase);

 //\*\*\*Кінець - Якщо вибраний метод 1 то запис запису у файл

}

//\*\*\*Кінець - Функція додавання нового елемента в базу даних.

//--------------------------------------------------

//\*\*\*Функція видалення елемента з бази даних.

void deleteElementInDatabase(int ID) {

 FILE \*fileOldDatabase; //Мітка для файлу старої (створеної) бази даних.

 FILE \*fileTemp; //Мітка для файлу тимчасової бази даних.

 char words[250]; //Масив символів для зберігання стрічки, яку зчитуєм з файлу.

 int i=0,n=0,k,j; //i - Змінна для перевірки видалений файл чи ні, n - к-ть записів.

 char \*end; //Вказівник для закінчення визначення цифр зі стрічки.

 struct travel oldTravel; //Структура для зберігання запису з бази даних.

 //\*\*\*Перевірка чи файл відкритий без помилок і вивід резульату.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 while(fgets(words,250,fileOldDatabase) != NULL && words[0] != '\n') {

 n++;

 }

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 //\*\*\*Кінець - Перевірка чи файл відкритий без помилок і вивід резульату.

 //\*\*\*Ввід запису в базу даних.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 if ((fileTemp=fopen("temp.txt","w")) == NULL) {

 printf("\nAn error is at opening of database temp.txt.");

 }

 for (k=0;k<n;k++) {

 char one[40];

 for (j=0;j<12;j++) {

 fscanf(fileOldDatabase,"%s",&one);

 if (j==0) {

 oldTravel.ID=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==1) {

 strcpy(oldTravel.service,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==2) {

 strcpy(oldTravel.name,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==3) {

 strcpy(oldTravel.cooperation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==4) {

 strcpy(oldTravel.recreation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==5) {

 strcpy(oldTravel.site,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==6) {

 strcpy(oldTravel.phone\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==7) {

 strcpy(oldTravel.mail,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==8) {

 strcpy(oldTravel.fax\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==9) {

 strcpy(oldTravel.address,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==10) {

 oldTravel.founded\_in=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==11) {

 oldTravel.rate=strtol(one,&end,10);

 }

 }

 if (ID==oldTravel.ID) { //Якщо ID рівні.

 i=1;

 printf("Element was deleted.");

 }

 else { //Інакше зберегти старий запис.

 fprintf(fileTemp,"%d %s ",oldTravel.ID,oldTravel.service);

 fprintf(fileTemp,"%s %s ",oldTravel.name,oldTravel.cooperation);

 fprintf(fileTemp,"%s %s ",oldTravel.recreation,oldTravel.site);

 fprintf(fileTemp,"%s %s ",oldTravel.phone\_number,oldTravel.mail);

 fprintf(fileTemp,"%s %s ",oldTravel.fax\_number,oldTravel.address);

 fprintf(fileTemp,"%d %d\n",oldTravel.founded\_in,oldTravel.rate);

 }

 }

 if (i==0) {

 printf("There is not such ID in the database.");

 }

 //\*Перевірка чи закритий файл бази даних чи ні та виведення результату.

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 if (fclose(fileTemp) != 0) {

 printf("\nAn error is at closing of file temp.txt.");

 }

 //\*Кінець - Перевірка чи закритий файл бази даних чи ні та виведення результату.

 remove(nameOfDatabase);

 rename("temp.txt",nameOfDatabase);

 //\*\*\*Кінець - Якщо вибраний метод 1 то запис запису у файл

}

//\*\*\*Кінець - Функція видалення елемента з бази даних.

//\*\*\*Функція редагування елемента в базі даних.

void edit(int ID, int elem, struct travel editTravel) {

 FILE \*fileOldDatabase; //Мітка для файлу старої (створеної) бази даних.

 FILE \*fileTemp; //Мітка для файлу тимчасової бази даних.

 char words[250]; //Масив символів для зберігання стрічки, яку зчитуєм з файлу.

 int i=0,n=0,k,j; //i - Змінна для перевірки видалений файл чи ні, n - к-ть записів.

 char \*end; //Вказівник для закінчення визначення цифр зі стрічки.

 struct travel oldTravel; //Структура для зберігання запису з бази даних.

 //\*\*\*Перевірка чи файл відкритий без помилок і вивід резульату.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 while(fgets(words,250,fileOldDatabase) != NULL && words[0] != '\n') {

 n++;

 }

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 //\*\*\*Кінець - Перевірка чи файл відкритий без помилок і вивід резульату.

 //\*\*\*Редагування бази даних.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 if ((fileTemp=fopen("temp.txt","w")) == NULL) {

 printf("\nAn error is at opening of database temp.txt.");

 }

 for (k=0;k<n;k++) {

 char one[40];

 for (j=0;j<12;j++) {

 fscanf(fileOldDatabase,"%s",&one);

 if (j==0) {

 oldTravel.ID=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==1) {

 strcpy(oldTravel.service,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==2) {

 strcpy(oldTravel.name,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==3) {

 strcpy(oldTravel.cooperation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==4) {

 strcpy(oldTravel.recreation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==5) {

 strcpy(oldTravel.site,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==6) {

 strcpy(oldTravel.phone\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==7) {

 strcpy(oldTravel.mail,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==8) {

 strcpy(oldTravel.fax\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==9) {

 strcpy(oldTravel.address,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==10) {

 oldTravel.founded\_in=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==11) {

 oldTravel.rate=strtol(one,&end,10);

 }

 }

 if (ID==oldTravel.ID) { //Якщо ID рівні.

 i=1;

 if (elem==1) {

 strcpy(oldTravel.service,editTravel.service);

 }

 else if (elem==2) {

 strcpy(oldTravel.name,editTravel.name);

 }

 else if (elem==3) {

 strcpy(oldTravel.cooperation,editTravel.cooperation);

 }

 else if (elem==4) {

 strcpy(oldTravel.recreation,editTravel.recreation);

 }

 else if (elem==5) {

 strcpy(oldTravel.site,editTravel.site);

 }

 else if (elem==6) {

 strcpy(oldTravel.phone\_number,editTravel.phone\_number);

 }

 else if (elem==7) {

 strcpy(oldTravel.mail,editTravel.mail);

 }

 else if (elem==8) {

 strcpy(oldTravel.fax\_number,editTravel.fax\_number);

 }

 else if (elem==9) {

 strcpy(oldTravel.address,editTravel.address);

 }

 else if (elem==10) {

 oldTravel.founded\_in=editTravel.founded\_in;

 }

 else if (elem==11) {

 oldTravel.rate=editTravel.rate;

 }

 fprintf(fileTemp,"%d %s ",oldTravel.ID,oldTravel.service);

 fprintf(fileTemp,"%s %s ",oldTravel.name,oldTravel.cooperation);

 fprintf(fileTemp,"%s %s ",oldTravel.recreation,oldTravel.site);

 fprintf(fileTemp,"%s %s ",oldTravel.phone\_number,oldTravel.mail);

 fprintf(fileTemp,"%s %s ",oldTravel.fax\_number,oldTravel.address);

 fprintf(fileTemp,"%d %d\n",oldTravel.founded\_in,oldTravel.rate);

 printf("\nElement was edited.");

 }

 else { //Інакше зберегти старий запис.

 fprintf(fileTemp,"%d %s ",oldTravel.ID,oldTravel.service);

 fprintf(fileTemp,"%s %s ",oldTravel.name,oldTravel.cooperation);

 fprintf(fileTemp,"%s %s ",oldTravel.recreation,oldTravel.site);

 fprintf(fileTemp,"%s %s ",oldTravel.phone\_number,oldTravel.mail);

 fprintf(fileTemp,"%s %s ",oldTravel.fax\_number,oldTravel.address);

 fprintf(fileTemp,"%d %d\n",oldTravel.founded\_in,oldTravel.rate);

 }

 }

 if (i==0) {

 printf("There is not such ID in the database.");

 }

 //\*Перевірка чи закритий файл бази даних чи ні та виведення результату.

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 if (fclose(fileTemp) != 0) {

 printf("\nAn error is at closing of file temp.txt.");

 }

 //\*Кінець - Перевірка чи закритий файл бази даних чи ні та виведення результату.

 remove(nameOfDatabase);

 rename("temp.txt",nameOfDatabase);

 //\*\*\*Кінець - Якщо вибраний метод 1 то запис запису у файл

}

//\*\*\*Кінець - Функція редагування елемента в базі даних.

//--------------------------------------------------

//\*\*\*Функція перегляду бази даних.

void revision() {

 FILE \*fileOldDatabase; //Мітка для файлу старої (створеної) бази даних.

 char words[250]; //Масив символів для зберігання стрічки, яку зчитуєм з файлу.

 int n=0,k,j; //n - к-ть записів.

 char \*end; //Вказівник для закінчення визначення цифр зі стрічки.

 struct travel oldTravel; //Структура для зберігання запису з бази даних.

 //\*\*\*Перевірка чи файл відкритий без помилок і вивід резульату.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 while(fgets(words,250,fileOldDatabase) != NULL && words[0] != '\n') {

 n++;

 }

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 //\*\*\*Кінець - Перевірка чи файл відкритий без помилок і вивід резульату.

 //\*\*\*Ввід запису в базу даних.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 for (k=0;k<n;k++) {

 char one[40];

 for (j=0;j<12;j++) {

 fscanf(fileOldDatabase,"%s",&one);

 if (j==0) {

 oldTravel.ID=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==1) {

 strcpy(oldTravel.service,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==2) {

 strcpy(oldTravel.name,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==3) {

 strcpy(oldTravel.cooperation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==4) {

 strcpy(oldTravel.recreation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==5) {

 strcpy(oldTravel.site,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==6) {

 strcpy(oldTravel.phone\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==7) {

 strcpy(oldTravel.mail,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==8) {

 strcpy(oldTravel.fax\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==9) {

 strcpy(oldTravel.address,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==10) {

 oldTravel.founded\_in=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==11) {

 oldTravel.rate=strtol(one,&end,10);

 }

 }

 printf("\n");

 printf("\n|----------------------------------");

 printf("\n| Record - %d",(k));

 printf("\n| Your dates: ");

 printf("\n| ID: %d\n| Service: %s\n| Name: %s\n| Cooperation: %s\n| Recreation: %s\n| The site: %s\n| Phone number: %s\n| Mail: %s\n| Fax number: %s\n| Address: %s\n| Founded in: %d\n| Rate: %d\n",oldTravel.ID,oldTravel.service,oldTravel.name,oldTravel.cooperation,oldTravel.recreation,oldTravel.site,oldTravel.phone\_number,oldTravel.mail,oldTravel.fax\_number,oldTravel.address,oldTravel.founded\_in,oldTravel.rate);

 printf("\n| ----------------------------------\n");

 }

 printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

 printf("\_\_\_\_\_ Records: %d \_\_\_\_\_\_\_",n);

 printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

}

//\*\*\*Кінець - Функція перегляду бази даних.

void search(int elem, struct travel searchTravel) {

 FILE \*fileOldDatabase; //Мітка для файлу старої (створеної) бази даних.

 char words[250]; //Масив символів для зберігання стрічки, яку зчитуєм з файлу.

 int i=0,n=0,k,j; //i - Змінна для перевірки видалений файл чи ні, n - к-ть записів.

 char \*end; //Вказівник для закінчення визначення цифр зі стрічки.

 struct travel oldTravel; //Структура для зберігання запису з бази даних.

 //\*\*\*Перевірка чи файл відкритий без помилок і вивід резульату.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 while(fgets(words,250,fileOldDatabase) != NULL && words[0] != '\n') {

 n++;

 }

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 //\*\*\*Кінець - Перевірка чи файл відкритий без помилок і вивід резульату.

 //\*\*\*Пошук в базі даних.

 if ((fileOldDatabase=fopen(nameOfDatabase,"r+")) == NULL) {

 printf("\nAn error is at opening of database. %s.",nameOfDatabase);

 }

 for (k=0;k<n;k++) {

 char one[40];

 //Зчитування з файлу запису.

 for (j=0;j<12;j++) {

 fscanf(fileOldDatabase,"%s",&one);

 if (j==0) {

 oldTravel.ID=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==1) {

 strcpy(oldTravel.service,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==2) {

 strcpy(oldTravel.name,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==3) {

 strcpy(oldTravel.cooperation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==4) {

 strcpy(oldTravel.recreation,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==5) {

 strcpy(oldTravel.site,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==6) {

 strcpy(oldTravel.phone\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==7) {

 strcpy(oldTravel.mail,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==8) {

 strcpy(oldTravel.fax\_number,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==9) {

 strcpy(oldTravel.address,one);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==10) {

 oldTravel.founded\_in=strtol(one,&end,10);

 fscanf(fileOldDatabase,"%c");

 }

 else if (j==11) {

 oldTravel.rate=strtol(one,&end,10);

 }

 }

 //Кінець - Зчитування з файлу запису.

 i=0;

 int nStr;

 //Порівння шуканого елемента в записі в залежності від поля пошуку.

 if (elem==1) {

 nStr=strlen(searchTravel.service);

 if (strncmp(oldTravel.service,searchTravel.service,nStr) == 0) {

 i=1;

 }

 }

 else if (elem==2) {

 nStr=strlen(searchTravel.name);

 if (strncmp(oldTravel.name,searchTravel.name,nStr) == 0) {

 i=1;

 }

 }

 else if (elem==3) {

 nStr=strlen(searchTravel.cooperation);

 if (strncmp(oldTravel.cooperation,searchTravel.cooperation,nStr) == 0) {

 i=1;

 }

 }

 else if (elem==4) {

 nStr=strlen(searchTravel.recreation);

 if (strncmp(oldTravel.recreation,searchTravel.recreation,nStr) == 0) {

 i=1;

 }

 }

 else if (elem==5) {

 nStr=strlen(searchTravel.site);

 if (strncmp(oldTravel.site,searchTravel.site,nStr) == 0) {

 i=1;

 }

 }

 else if (elem==6) {

 nStr=strlen(searchTravel.phone\_number);

 if (strncmp(oldTravel.phone\_number,searchTravel.phone\_number,nStr) == 0) {

 i=1;

 }

 }

 else if (elem==7) {

 nStr=strlen(searchTravel.mail);

 if (strncmp(oldTravel.mail,searchTravel.mail,nStr) == 0) {

 i=1;

 }

 }

 else if (elem==8) {

 nStr=strlen(searchTravel.fax\_number);

 if (strncmp(oldTravel.fax\_number,searchTravel.fax\_number,nStr) == 0) {

 i=1;

 }

 }

 else if (elem==9) {

 nStr=strlen(searchTravel.address);

 if (strncmp(oldTravel.address,searchTravel.address,nStr) == 0) {

 i=1;

 }

 }

 else if (elem==10) {

 if (oldTravel.founded\_in==searchTravel.founded\_in) {

 i=1;

 }

 }

 else if (elem==11) {

 if (oldTravel.rate==searchTravel.rate) {

 i=1;

 }

 }

 //Кінець - Порівння шуканого елемента в записі в залежності від поля пошуку.

 if (i==1) { //Якщо елемент знайдений вивести запис, інакше вивести повідомлення про його відсутність.

 printf("\n");

 printf("\n----------------------------------");

 printf("\nRecord - %d",(k));

 printf("\nYour dates:\n");

 printf("\nID: %d\nservice: %s\nName: %s\ncooperation: %s\nrecreation: %s\nthe site: %s\nphone\_number: %s\nmail: %s\nfax\_number: %s\naddress: %s\nfounded\_in: %d\nrate: %d\n",oldTravel.ID,oldTravel.service,oldTravel.name,oldTravel.cooperation,oldTravel.recreation,oldTravel.site,oldTravel.phone\_number,oldTravel.mail,oldTravel.fax\_number,oldTravel.address,oldTravel.founded\_in,oldTravel.rate);

 printf("\n----------------------------------\n");

 }

 }

 if (i==0) {

 printf("\nThismelement is not un database.");

 }

 //\*Перевірка чи закритий файл бази даних чи ні та виведення результату.

 if (fclose(fileOldDatabase) != 0) {

 printf("\nAn error is at closing of file %s.",nameOfDatabase);

 }

 //\*\*\*Кінець - Якщо вибраний метод 1 то запис запису у файл

}

//\*\*\*Кінець - Функція редагування елемента в базі даних.

Результати виконання програми

1. Запуск програми