«Министерство образования и науки Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Алтайский государственный технический университет им. И.И. Ползунова»

Факультет информационных технологий

Кафедра прикладной математики

Отчёт защищён с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Преподаватель Егорова Е.В.

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г.

Отчёт

по курсовой работе

«*ПРОСТЕЙШИЕ БАЗЫ ДАННЫХ*»

по дисциплине «Программирование - 2»

Студент группы ПИ 92 В.М. Шульпов

Преподаватель доцент, к.т. н. Егорова Е.В.

Барнаул 2020

Код программы

/\*Программа предназначена для работы с данными по лекарствам.

Пользователь может добавлять и удалять данные,

также может редактировать и делать запросы по любым полям.

Разработана запись в файл, благодаря которой файлы будут хранится на диске.\*/

//в программе можно вводить не более 5 аптек (для удобства вывода таблицы данных)

#define \_CRT\_SECURE\_NO\_WARNINGS

#include "stdio.h"

#include <conio.h>

#include <stdlib.h>

#include <string.h>

#include <locale.h>

#include <Windows.h>

#define clear(stream) rewind((stream))

#define MAX\_MEDICAMENTS 100 //максимальное количество лекарств

#define MAX\_LENGTH 50 //максимальная длина текста

#define MAX\_PHARMACIES 5 //максимальное кол-во аптек, имеющих в наличии лекраство

#define NUM\_KEYS\_TABLE 6 //кол-во ключей таблицы (заголовков столбцов)

#define NUM\_SORT\_FIELDS 5 //кол-во сортируемых столбцов

#define NUM\_EDIT\_FIELDS 5 //кол-во редактируемых столбцов

#define LENGTH\_DATE 10 //длина даты (2000.09.14)

////////////////////////////////////////////////////////////////////////////////////////////////////

//ГЛОБАЛЬНЫЕ ПЕРЕМЕННЫЕ

char stars[] = "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

char titles[NUM\_KEYS\_TABLE][MAX\_LENGTH] = {

"id",

"название лекарства",

"страна-производитель",

"дата производства",

"№ аптек, имеющих лек.",

"цена" };

char fileName[] = "medicines.txt"; //основной файл

char fileName2[] = "rewrite.txt"; //для переписывания при редактировании, удалении

/\*med - структура для добавления записи, редактирования записи

new\_med - для записи данных при редактировании\*/

struct {

int id; //ключ

char name[MAX\_LENGTH]; //название

char country[MAX\_LENGTH]; //страна-производитель

char date[MAX\_LENGTH]; //дата производства

int pharmacy\_number[MAX\_PHARMACIES]; //номера аптек

int price; //максимальная цена

}med, new\_med;

FILE \*file;//основной файл

FILE \*file2;//вспомогательный файл

////////////////////////////////////////////////////////////////////////////////////////////////////

int append\_med(int \*\_count); // ФУНКЦИЯ ДОБАВЛЕНИЯ 1 ЛЕКАРСТВА

int edit\_meds(int \*\_count); // ФУНКЦИЯ РЕДАКТИРОВАНИЯ 1 ЛЕКАРСТВА

int delete\_meds(int\* \_count); // ФУНКЦИЯ УДАЛЕНИЯ 1 ЛЕКАРСТВА

int make\_request(int\* \_count); // ФУНКЦИЯ ВЫБОРА И ВЫПОЛНЕНИЯ ЗАПРОСОВ

int output\_meds(int\* \_count); // ФУНКЦИЯ ВЫВОДА ВСЕХ ЛЕКАРСТВ

void printTitles(); // ФУНКЦИЯ ПЕЧАТИ ЗАГОЛОВОЧНОЙ ЧАСТИ ТАБЛИЦЫ

bool name\_request(char med\_name[], char name[]); // ФУНКЦИЯ ЗАПРОСА ПО НАЗВАНИЮ ЛЕКАРСТВА

bool country\_request(char med\_country[], char country[]); // ФУНКЦИЯ ЗАПРОСА ПО СТРАНЕ ЛЕКАРСТВА

bool date\_request(char med\_date[], char date[]); // ФУНКЦИЯ ЗАПРОСА ПО ДАТЕ ЛЕКАРСТВА

bool price\_request(int med\_price, int price, char choice\_price); // ФУНКЦИЯ ЗАПРОСА ПО ЦЕНЕ ЛЕКАРСТВА

bool pharmacy\_number\_request(int med\_pharmacy\_number[], int pharmacy\_number); // ФУНКЦИЯ ЗАПРОСА ПО НАЛИЧИЮ ЛЕКАРСТВА В АПТЕКЕ

////////////////////////////////////////////////////////////////////////////////////////////////////int main(){

system("color F0");

setlocale(LC\_ALL, "Russian");

int choice; //вариант выбора в меню(0, 1, 2...)

int сhoice\_correction; //подвыбор в "вводе и корректировке"

int f\_exit = 0; //флаг выхода

int count = 0;//текущее кол-во лекарств

int error; //номер ошибки

file = fopen(fileName, "a");

fclose(file);

file = fopen(fileName, "r+");

if (file == NULL) return -1; /\* Файл не открылся \*/

rewind(file);

fscanf(file, "%5d", &count);

if (count == NULL) fprintf(file, "%5d\n", count); //записываем 0 (кол-во записей) в начало

else {

rewind(file);

fscanf(file, "%5d", &count);

}

fclose(file);

printf("УПРАВЛЕНИЕ ДАННЫМИ ЛЕКАРСТВ АПТЕКИ\n");

do {

printf("\n1 - ввод данных и корректировка\n2 - вывод всех данных\n3 - сделать запрос\n0 - выйти\nВаш выбор:\t");

clear(stdin);

scanf\_s("%d", &choice);

switch (choice)

{

case 0:

puts("\nВЫ ВЫШЛИ\n");

f\_exit = 1;

return 0;

case 1:

printf("\t1 - ввод нового лекарства\n\t2 - удаление лекарства\n\t3 - редактирование существующего лекарства\n\t0 - назад\nВаш выбор:\t");

clear(stdin);

scanf\_s("%d", &сhoice\_correction);

switch (сhoice\_correction)

{

case 0:

puts("\nВЫ ВЕРНУЛИСЬ\n");

break;

case 1:

if (append\_med(&count) == -1) printf("\nФайл не открылся\n");

break;

case 2:

error = output\_meds(&count);

if (error == 1) printf("\nВыводить нечего, файл пуст\n");

if (error == -1) printf("\nФайл не открылся\n");

error = delete\_meds(&count);

if (error == -1) printf("\nФайл не открылся\n");

if (error == 1) printf("\nУдалять нечего, файл пуст\n");

break;

case 3:

error = output\_meds(&count);

if (error == 1) printf("\nВыводить нечего, файл пуст\n");

if(error == -1) printf("\nФайл не открылся\n");

error = edit\_meds(&count);

if (error == -1) printf("\nФайл не открылся\n");

if (error == 1) printf("\nРедактировать нечего, файл пуст\n");

break;

default:

puts("\nВВОД НЕВЕРЕН\n");

break;

}

break;

case 2:

error = output\_meds(&count);

if (error == 1) printf("\nВыводить нечего, файл пуст\n");

if(error==-1) printf("\nФайл не открылся\n");

break;

case 3:

error = output\_meds(&count);

if (error == 1) printf("\nВыводить нечего, файл пуст\n");

if (error == -1) printf("\nФайл не открылся\n");

error = make\_request(&count);

if (error == -1) printf("\nФайл не открылся\n");

if (error == 1) printf("\nВыборка невозможна, файл пуст\n");

break;

default:

puts("\nВВОД НЕВЕРЕН\n");

break;

}

} while (f\_exit != 1);

return 0;

}

int append\_med(int \*\_count){ // ФУНКЦИЯ ДОБАВЛЕНИЯ ЛЕКАРСТВА

file = fopen(fileName, "r+");

if (file == NULL) return -1; /\* Файл не открылся \*/

fseek(file, 0L, SEEK\_END);

bool date\_flag = true; //флаг проверки даты

int pharm = 0; //кол-во аптек

fprintf(file, "%5d", \*\_count);

puts("\n\t\tВВОД НОВОГО ЛЕКАРСТВА\n");

printf("\t\tВведите название (ШАБЛОН: Perekis\_vodoroda)\n\t\t\tНазвание:\t");

clear(stdin);

scanf\_s("%s", &med.name, MAX\_LENGTH);

fprintf(file, "%25.25s", med.name);

clear(stdin);

printf("\t\tВведите страну-производителя (ШАБЛОН: Arabskie\_Emirati)\n\t\t\tСтрана:\t");

clear(stdin);

scanf\_s("%s", &med.country, MAX\_LENGTH);

fprintf(file, "%25.25s", med.country);

printf("\t\tВведите дату производства лекарства (ШАБЛОН [год.месяц.число] с нулями: 2012.09.02)\n\t\t\tДата:\t");

do {

date\_flag = true;

clear(stdin);

scanf\_s("%s", &med.date, MAX\_LENGTH);

for (int k = 0; k < LENGTH\_DATE; k++)

{

if (

med.date[k] != '.' && (med.date[k] < '0' || med.date[k] > '9') //не точка и не цифра

|| strlen(med.date) != LENGTH\_DATE //длина не соотвествует

|| med.date[4] != '.' //5 символ не точка

|| med.date[7] != '.' //7 символ не точка

|| med.date[5] > '1' //6 символ больше 1

|| med.date[8] > '3' //9 символ больше 1

)

{

printf("\t\tВВОД НЕВЕРЕН. ВВОДИТЕ, ПОЖАЛУЙСТА, ПО ШАБЛОНУ\n\t\t\tДата:\t");

date\_flag = false;

break;

}

}

} while (date\_flag == false);

fprintf(file, "%25.25s", med.date);

printf("\t\tСколько аптек имеют в наличии это лекраство? (максимум %d)\t", MAX\_PHARMACIES);

do {

clear(stdin);

scanf\_s("%d", &pharm);

if (pharm > MAX\_PHARMACIES || pharm < 1) printf("\t\tНекорректный ввод, введите заново:\t");

else break;

} while (1);

printf("\t\t(ШАБЛОН: 12)\n");

for (int i = 0; i < MAX\_PHARMACIES; i++) {

if (i < pharm) {

do {

printf("\t\t\tНомер аптеки (%d) (до №99):\t", i + 1);

clear(stdin);

scanf\_s("%d", &med.pharmacy\_number[i]);

} while (med.pharmacy\_number[i] <= 0 || med.pharmacy\_number[i]>99);

fprintf(file, "%5d", med.pharmacy\_number[i]);

}

else fprintf(file, "%5d", 0);

}

do {

printf("\t\tВведите цену лекарства (ШАБЛОН: 200)\n\t\t\tЦена:\t");

clear(stdin);

scanf\_s("%d", &med.price);

} while (med.price <= 0);

fprintf(file, "%10d\n", med.price);

(\*\_count)++;

rewind(file);

fprintf(file, "%5d", \*\_count);

fseek(file, 0L, SEEK\_END);

fclose(file);

return 0;

};

int edit\_meds(int\* \_count) {// ФУНКЦИЯ РЕДАКТИРОВАНИЯ 1 ЛЕКАРСТВА

file = fopen(fileName, "r+");

file2 = fopen(fileName2, "w");

if (file == NULL || file2 == NULL) return -1; // Файл не открылся

if (\*\_count == 0) {

return 1;

}

puts("\n\t\tРЕДАКТИРОВАНИЕ СУЩЕСТВУЮЩЕГО ЛЕКАРСТВА");

int id; // вводимый id

int choice\_edit[NUM\_EDIT\_FIELDS]; //массив с выбором редактируемых полей (1 или 0)

int flag\_input = 0; // флаг корректного ввода выбора вариантов редактирования

bool date\_flag = true; //флаг для проверки ввода даты

int pharm; // введенное кол-во аптек

printf("Введите id лекарства: ");

do {

clear(stdin);

scanf\_s("%d", &id);

if (id >= \*\_count || id < 0) printf("Такого id в таблице нет, введите другое: ");

else break;

} while (1);

printf("\t\tКакое поле вы хотите изменить?\n\t\t1)название\n\t\t2) страну-производителя\n\t\t3) дату\n\t\t4)номера аптек\n\t\t5)цену\n\t\t(ШАБЛОН: 1 0 0 1 0 - для редактирования названия {1 столбец} и номеров аптек {4 столбец})\n\t\t\tВаш выбор редактируемых полей:\t");

do {

clear(stdin);

for (int i = 0; i < NUM\_SORT\_FIELDS; i++)

scanf\_s("%d", &choice\_edit[i]);

for (int i = 0; i < NUM\_SORT\_FIELDS; i++)

{

flag\_input = 1;

if (choice\_edit[i] != 0 && choice\_edit[i] != 1) {

printf("\n\tВВОД НЕВЕРЕН\n\tВВОДИТЕ, ПОЖАЛУЙСТА, ПО ШАБЛОНУ\n\tВаш выбор редактируемых полей:\t");

flag\_input = 0;

break;

}

}

} while (flag\_input == 0);

/\*записыываем новые выбранные данные в переменные\*/

for (int i = 0; i < NUM\_EDIT\_FIELDS; i++) {

if (choice\_edit[i] == 1) {

switch (i) {

case 0:

printf("\t\tВведите название (ШАБЛОН: Perekis\_vodoroda)\n\t\t\tНазвание:\t");

clear(stdin);

scanf\_s("%s", &new\_med.name, MAX\_LENGTH);

//fscanf(file, "%25.25s", &med.name);

//printf("\n%s\_\_\n", med.name);

//fprintf(file, "%25.25s", med.name);

break;

case 1:

printf("\t\tВведите страну-производителя (ШАБЛОН: Arabskie\_Emirati)\n\t\t\tСтрана:\t");

clear(stdin);

scanf\_s("%s", &new\_med.country, MAX\_LENGTH);

//fprintf(file, "%25.25s", med.country);

break;

case 2:

printf("\t\tВведите дату производства лекарства (ШАБЛОН [год.месяц.число] с нулями: 2012.09.02)\n\t\t\tДата:\t");

do {

date\_flag = true;

clear(stdin);

scanf\_s("%s", &new\_med.date, MAX\_LENGTH);

for (int k = 0; k < LENGTH\_DATE; k++)

{

if (

new\_med.date[k] != '.' && (new\_med.date[k] < '0' || new\_med.date[k] > '9') //не точка и не цифра

|| strlen(new\_med.date) != LENGTH\_DATE //длина не соотвествует

|| new\_med.date[4] != '.' //5 символ не точка

|| new\_med.date[7] != '.' //7 символ не точка

|| new\_med.date[5] > '1' //6 символ больше 1

|| new\_med.date[8] > '3' //9 символ больше 1

)

{

printf("\t\tВВОД НЕВЕРЕН. ВВОДИТЕ, ПОЖАЛУЙСТА, ПО ШАБЛОНУ\n\t\t\tДата:\t");

date\_flag = false;

break;

}

}

} while (date\_flag == false);

//fprintf(file, "%25.25s", med.date);

break;

case 3:

printf("\t\tСколько аптек имеют в наличии это лекраство?\t");

do {

clear(stdin);

scanf\_s("%d", &pharm);

if (pharm > MAX\_PHARMACIES || pharm < 1) printf("\t\tВведите меньше %d аптек:\t", MAX\_PHARMACIES + 1);

} while (pharm > MAX\_PHARMACIES || pharm < 1);

printf("\t\t(ШАБЛОН: 12)\n");

for (int i = 0; i < MAX\_PHARMACIES; i++) {

if (i < pharm) {

do {

printf("\t\t\tНомер аптеки (%d):\t", i + 1);

clear(stdin);

scanf\_s("%d", &new\_med.pharmacy\_number[i]);

} while (new\_med.pharmacy\_number[i] <= 0);

//fprintf(file, "%3d ", med.pharmacy\_number[i]);

}

else {

new\_med.pharmacy\_number[i] = 0;

//fprintf(file, "%3d ", 0);

}

}

//med.pharmacy\_number[pharm] = 0;

break;

case 4:

do {

printf("\t\tВведите цену лекарства (ШАБЛОН: 200)\n\t\t\tЦена:\t");

clear(stdin);

scanf\_s("%d", &new\_med.price);

} while (new\_med.price <= 0);

//fprintf(file, "%10d\n", med.price);

break;

}

}

}

fseek(file, 0L, SEEK\_SET);

fseek(file2, 0L, SEEK\_SET);

fscanf(file, "%5d ", \_count);

fprintf(file2, "%5d\n", \*\_count);

for (int i = 0; i < \*\_count; i++) {

fscanf(file, "%5d", &med.id);

fscanf(file, "%25s", &med.name);

fscanf(file, "%25s", &med.country);

fscanf(file, "%25s", &med.date);

for (int j = 0; j < MAX\_PHARMACIES; j++)

fscanf(file, "%5d", &med.pharmacy\_number[j]);

fscanf(file, "%10d\n", &med.price);

if(med.id==id){

fprintf(file2, "%5d", med.id);

if (choice\_edit[0] == 1) fprintf(file2, "%25.25s", new\_med.name);

else fprintf(file2, "%25.25s", med.name);

if (choice\_edit[1] == 1) fprintf(file2, "%25.25s", new\_med.country);

else fprintf(file2, "%25.25s", med.country);

if (choice\_edit[2] == 1) fprintf(file2, "%25.25s", new\_med.date);

else fprintf(file2, "%25.25s", med.date);

for (int i = 0; i < MAX\_PHARMACIES; i++)

if (choice\_edit[3] == 1) fprintf(file2, "%5d", new\_med.pharmacy\_number[i]);

else fprintf(file2, "%5d", med.pharmacy\_number[i]);

if (choice\_edit[4] == 1) fprintf(file2, "%10d\n", new\_med.price);

else fprintf(file2, "%10d\n", med.price);

}

else{

fprintf(file2, "%5d", med.id);

fprintf(file2, "%25.25s", med.name);

fprintf(file2, "%25.25s", med.country);

fprintf(file2, "%25.25s", med.date);

for (int i = 0; i < MAX\_PHARMACIES; i++)

fprintf(file2, "%5d", med.pharmacy\_number[i]);

fprintf(file2, "%10d\n", med.price);

}

}

fclose(file);

fclose(file2);

remove(fileName);

rename(fileName2, fileName);

return 0;

}

int delete\_meds(int\* \_count) {// ФУНКЦИЯ УДАЛЕНИЯ 1 ЛЕКАРСТВА

file = fopen(fileName, "r+");

file2 = fopen(fileName2, "w");

if (file == NULL || file2 == NULL) return -1; // Файл не открылся

if (\*\_count == 0) {

return 1;

}

int id\_increment=0; //id инкремент

puts("\n\t\tУДАЛЕНИЕ ЛЕКАРСТВА\n");

int id;

printf("Введите id лекарства: ");

do {

clear(stdin);

scanf\_s("%d", &id);

if (id >= \*\_count || id < 0) printf("Такого id в таблице нет, введите другое: ");

else break;

} while (1);

fseek(file, 0L, SEEK\_SET);

fseek(file2, 0L, SEEK\_SET);

fscanf(file, "%5d ", \_count);

fprintf(file2, "%5d\n", (\*\_count)-1);

for (int i = 0; i < \*\_count; i++) {

fscanf(file, "%5d", &med.id);

fscanf(file, "%25s", &med.name);

fscanf(file, "%25s", &med.country);

fscanf(file, "%25s", &med.date);

for (int j = 0; j < MAX\_PHARMACIES; j++)

fscanf(file, "%5d", &med.pharmacy\_number[j]);

fscanf(file, "%10d\n", &med.price);

for (int j = 0; j < MAX\_PHARMACIES; j++)

printf("%5d", med.pharmacy\_number[i]);

printf("%10d\n", med.price);

if (med.id != id) {

fprintf(file2, "%5d", id\_increment++);

fprintf(file2, "%25.25s", med.name);

fprintf(file2, "%25.25s", med.country);

fprintf(file2, "%25.25s", med.date);

for (int i = 0; i < MAX\_PHARMACIES; i++)

fprintf(file2, "%5d", med.pharmacy\_number[i]);

fprintf(file2, "%10d\n", med.price);

}

}

(\*\_count)--;

fclose(file);

fclose(file2);

remove(fileName);

rename(fileName2, fileName);

return 0;

};

void printTitles() { // ФУНКЦИЯ ПЕЧАТИ ЗАГОЛОВОЧНОЙ ЧАСТИ ТАБЛИЦЫ

printf("\n %s\n", stars);

for (int i = 0; i != NUM\_KEYS\_TABLE; i++)

{

if (i == 0) {

printf(" \* ");

printf(" %5.5s \* ", titles[i]);

}

else printf(" %25.25s \* ", titles[i]);

}

printf("\n %s\n", stars);

}

int output\_meds(int\* \_count){ // ФУНКЦИЯ ВЫВОДА ВСЕХ ЛЕКАРСТВ

if (\*\_count == 0) return 1;

else {

file = fopen(fileName, "r+");

if (file == NULL) return -1; /\* Файл не открылся \*/

fseek(file, 0L, SEEK\_SET);

fscanf(file, "%5d", \_count);

printf("Данные по лекарствам (%d): \n", \*\_count);

printTitles();

for (int i = 0; i < \*\_count; i++) {

printf(" \* ");

fscanf(file, "%5d", &med.id);

printf(" %5d \* ", med.id);

fscanf(file, "%25s", &med.name);

printf(" %25.25s \* ", med.name);

fscanf(file, "%25s", &med.country);

printf(" %25.25s \* ", med.country);

fscanf(file, "%25s", &med.date);

printf(" %25.25s \* ", med.date);

for (int j = 0; j < MAX\_PHARMACIES; j++) {

fscanf(file, "%5d", &med.pharmacy\_number[j]);

if (med.pharmacy\_number[j] != 0)

printf("%5d", med.pharmacy\_number[j]); //рассчитано для 5 аптек

else

printf(" ");

}

printf(" \* ");

fscanf(file, "%10d\n", &med.price);

printf(" %25d \* ", med.price);

printf("\n %s\n", stars);

}

fclose(file);

}

return 0;

}

int make\_request(int\* \_count) {// ФУНКЦИЯ ВЫБОРА И ВЫПОЛНЕНИЯ ЗАПРОСА

if (\*\_count == 0) return 1;

file = fopen(fileName, "r");

if (file == NULL) return -1; /\* Файл не открылся \*/

fseek(file, 0L, SEEK\_SET);

fscanf(file, "%5d ", \_count);

int flag\_title = 0; //флаг (была ли выведена заголовочная часть таблицы)

int choice\_request[NUM\_SORT\_FIELDS]; //массив со значениями 1 или 0 для выбора запроса

printf("\tПо каким столбцам вы хотите сделать запрос?\n\t1) по названию\n\t2) стране-производителю\n\t3) по дате\n\t4) по наличию в опеределнной аптеке\n\t5) по цене\n\t(ШАБЛОН: 1 0 0 1 0 - запрос по названию {1 стобец} и цене {4 столбец})\n\tВаш запрос:\t");

int flag\_input = 0;

do {

clear(stdin);

for (int i = 0; i < NUM\_SORT\_FIELDS; i++)

scanf\_s("%d", &choice\_request[i]);

for (int i = 0; i < NUM\_SORT\_FIELDS; i++)

{

flag\_input = 1;

if (choice\_request[i] != 0 && choice\_request[i] != 1) {

printf("\n\tВВОД НЕВЕРЕН\n\tВВОДИТЕ, ПОЖАЛУЙСТА, ПО ШАБЛОНУ\n\tВаш запрос:\t");

flag\_input = 0;

break;

}

}

} while (flag\_input == 0);

bool date\_flag = true; //флаг для ввода валидной даты

bool price\_flag = true; //флаг для ввода валидной цены

bool request\_flag = true; //флаг запроса (изменяется в ходе проверки по каждому столбцу)

bool is\_request\_flag = false; //флаг, показывающий, что пользователь выбрал хотя бы один запрос (случай 0 0 0 0 0 - false)

bool title\_flag = false; //флаг = true, если заголовки таблицы выведены хотя бы один раз

for (int i = 0; i < \*\_count; i++) {

request\_flag = true;

fscanf(file, "%5d", &med.id);

fscanf(file, "%25s", &med.name);

fscanf(file, "%25s", &med.country);

fscanf(file, "%25s", &med.date);

for (int j = 0; j < MAX\_PHARMACIES; j++)

fscanf(file, "%5d", &med.pharmacy\_number[j]);

fscanf(file, "%10d\n", &med.price);

for (int j = 0; j < NUM\_SORT\_FIELDS; j++)

{

if (choice\_request[j] == 1) {

switch (j) {

case 0:

char name[MAX\_LENGTH];

if (i == 0) {

printf("Введите название лекарства\n(ШАБЛОН: Perekis\_vodoroda)\nНазвание:\t");

clear(stdin);

scanf\_s("%s", &name, MAX\_LENGTH);

}

is\_request\_flag = request\_flag = request\_flag && name\_request(med.name, name);

break;

case 1:

char country[MAX\_LENGTH];

if (i == 0) {

printf("Введите страну-производителя лекарства\n(ШАБЛОН: Arabskie\_Emirati)\nСтрана:\t");

clear(stdin);

scanf\_s("%s", &country, MAX\_LENGTH);

}

is\_request\_flag = request\_flag = request\_flag && country\_request(med.country, country);

break;

case 2:

char date[MAX\_LENGTH];

if (i == 0) {

printf("Введите дату производства лекарства\n(ШАБЛОН [год.месяц.число] с нулями: 2012.09.02)\nДата:\t");

do {

date\_flag = true;

clear(stdin);

scanf\_s("%s", &date, MAX\_LENGTH);

for (int k = 0; k < LENGTH\_DATE; k++)

{

if (date[k] != '.' && (date[k] < '0' || date[k] > '9') || strlen(date) != LENGTH\_DATE || date[4] != '.' || date[7] != '.')

{

printf("ВВОД НЕВЕРЕН\nВВОДИТЕ, ПОЖАЛУЙСТА, ПО ШАБЛОНУ\nДата:\t");

date\_flag = false;

break;

}

}

} while (date\_flag == false);

}

is\_request\_flag = request\_flag = request\_flag && date\_request(med.date, date);

break;

case 3:

int pharmacy\_number;

if (i == 0) {

do {

printf("Введите номер апетки\n(ШАБЛОН: 12)\nНомер аптеки:\t");

clear(stdin);

scanf\_s("%d", &pharmacy\_number);

} while (pharmacy\_number <= 0);

}

is\_request\_flag = request\_flag = request\_flag && pharmacy\_number\_request(med.pharmacy\_number, pharmacy\_number);

break;

case 4:

int price;

char choice\_price;

if (i == 0) {

do {

price\_flag = true;

printf("Введите цену лекарства (ШАБЛОН: 200)\nЦена:\t");

clear(stdin);

scanf\_s("%d", &price);

printf("знак > (не дешевле), < (не дороже), = (равно) (ШАБЛОН: <)\nЗнак:\t");

clear(stdin);

scanf\_s("%c", &choice\_price, MAX\_LENGTH);

if (choice\_price != '<' && choice\_price != '>' && choice\_price != '=' || price < 0) //в случае, когда пользователь вводит текст, price == 0

{

printf("ВВОД НЕВЕРЕН\nВВОДИТЕ, ПОЖАЛУЙСТА, ПО ШАБЛОНУ\n");

price\_flag = false;

break;

}

} while (price\_flag == false);

}

is\_request\_flag = request\_flag = request\_flag && price\_request(med.price, price, choice\_price);

break;

}

}

}

if (is\_request\_flag == true) {

if (title\_flag == false) {

title\_flag = true;

printTitles();

}

if (request\_flag == true)

{

printf(" \* ");

printf(" %3d \* ", med.id);

printf(" %25.25s \* ", med.name);

printf(" %25.25s \* ", med.country);

printf(" %25.25s \* ", med.date);

for (int j = 0; j < MAX\_PHARMACIES; j++) {

if (med.pharmacy\_number[j] != 0)

printf("%3d, ", med.pharmacy\_number[j]); //расситано для 5 аптек

else

printf(" ");

}

printf(" \* ");

printf(" %25.2d \* ", med.price);

printf("\n%s\n", stars);

}

}

}

if (title\_flag == false) printf("По вашему запросу ничего не найдено\n");

fclose(file);

}

bool name\_request(char med\_name[], char name[]) {// ФУНКЦИЯ ЗАПРОСА ПО НАЗВАНИЮ ЛЕКАРСТВА

if (strstr(med\_name, name) != NULL)

return true;

return false;

}

bool country\_request(char med\_country[], char country[]) {// ФУНКЦИЯ ЗАПРОСА ПО СТРАНЕ ЛЕКАРСТВА

if (strcmp(med\_country, country) == NULL)

return true;

return false;

}

bool date\_request(char med\_date[], char date[]) { // ФУНКЦИЯ ЗАПРОСА ПО ДАТЕ ЛЕКАРСТВА

if (strcmp(med\_date, date) > 0)

return true;

return false;

}

bool price\_request(int med\_price, int price, char choice\_price) {// ФУНКЦИЯ ЗАПРОСА ПО ЦЕНЕ ЛЕКАРСТВА

if (choice\_price == '<') {

if (med\_price <= price)

return true;

return false;

}

else if (choice\_price == '>') {

if (med\_price >= price) return true;

return false;

}

else {

if (med\_price == price) return true;

return false;

}

}

bool pharmacy\_number\_request(int med\_pharmacy\_number[], int pharmacy\_number) {// ФУНКЦИЯ ЗАПРОСА ПО НАЛИЧИЮ ЛЕКАРСТВА В АПТЕКЕ

for (int j = 0; j < MAX\_PHARMACIES; j++)

if (med\_pharmacy\_number[j] == pharmacy\_number)

return true;

return false;

}int output\_meds(int\* \_count){ // ФУНКЦИЯ ВЫВОДА ВСЕХ ЛЕКАРСТВ

if (\*\_count == 0) return 1;

else {

file = fopen(fileName, "r+");

if (file == NULL) return -1; /\* Файл не открылся \*/

fseek(file, 0L, SEEK\_SET);

fscanf(file, "%5d", \_count);

printf("Данные по лекарствам (%d): \n", \*\_count);

printTitles();

for (int i = 0; i < \*\_count; i++) {

printf(" \* ");

fscanf(file, "%5d", &med.id);

printf(" %5d \* ", med.id);

fscanf(file, "%25s", &med.name);

printf(" %25.25s \* ", med.name);

fscanf(file, "%25s", &med.country);

printf(" %25.25s \* ", med.country);

fscanf(file, "%25s", &med.date);

printf(" %25.25s \* ", med.date);

for (int j = 0; j < MAX\_PHARMACIES; j++) {

fscanf(file, "%5d", &med.pharmacy\_number[j]);

if (med.pharmacy\_number[j] != 0)

printf("%5d", med.pharmacy\_number[j]); //рассчитано для 5 аптек

else

printf(" ");

}

printf(" \* ");

fscanf(file, "%10d\n", &med.price);

printf(" %25d \* ", med.price);

printf("\n %s\n", stars);

}

fclose(file);

}

return 0;

}

int make\_request(int\* \_count) {// ФУНКЦИЯ ВЫБОРА И ВЫПОЛНЕНИЯ ЗАПРОСА

if (\*\_count == 0) return 1;

file = fopen(fileName, "r");

if (file == NULL) return -1; /\* Файл не открылся \*/

fseek(file, 0L, SEEK\_SET);

fscanf(file, "%5d ", \_count);

int flag\_title = 0; //флаг (была ли выведена заголовочная часть таблицы)

int choice\_request[NUM\_SORT\_FIELDS]; //массив со значениями 1 или 0 для выбора запроса

printf("\tПо каким столбцам вы хотите сделать запрос?\n\t1) по названию\n\t2) стране-производителю\n\t3) по дате\n\t4) по наличию в опеределнной аптеке\n\t5) по цене\n\t(ШАБЛОН: 1 0 0 1 0 - запрос по названию {1 стобец} и цене {4 столбец})\n\tВаш запрос:\t");

int flag\_input = 0;

do {

clear(stdin);

for (int i = 0; i < NUM\_SORT\_FIELDS; i++)

scanf\_s("%d", &choice\_request[i]);

for (int i = 0; i < NUM\_SORT\_FIELDS; i++)

{

flag\_input = 1;

if (choice\_request[i] != 0 && choice\_request[i] != 1) {

printf("\n\tВВОД НЕВЕРЕН\n\tВВОДИТЕ, ПОЖАЛУЙСТА, ПО ШАБЛОНУ\n\tВаш запрос:\t");

flag\_input = 0;

break;

}

}

} while (flag\_input == 0);

bool date\_flag = true; //флаг для ввода валидной даты

bool price\_flag = true; //флаг для ввода валидной цены

bool request\_flag = true; //флаг запроса (изменяется в ходе проверки по каждому столбцу)

bool is\_request\_flag = false; //флаг, показывающий, что пользователь выбрал хотя бы один запрос (случай 0 0 0 0 0 - false)

bool title\_flag = false; //флаг = true, если заголовки таблицы выведены хотя бы один раз

for (int i = 0; i < \*\_count; i++) {

request\_flag = true;

fscanf(file, "%5d", &med.id);

fscanf(file, "%25s", &med.name);

fscanf(file, "%25s", &med.country);

fscanf(file, "%25s", &med.date);

for (int j = 0; j < MAX\_PHARMACIES; j++)

fscanf(file, "%5d", &med.pharmacy\_number[j]);

fscanf(file, "%10d\n", &med.price);

for (int j = 0; j < NUM\_SORT\_FIELDS; j++)

{

if (choice\_request[j] == 1) {

switch (j) {

case 0:

char name[MAX\_LENGTH];

if (i == 0) {

printf("Введите название лекарства\n(ШАБЛОН: Perekis\_vodoroda)\nНазвание:\t");

clear(stdin);

scanf\_s("%s", &name, MAX\_LENGTH);

}

is\_request\_flag = request\_flag = request\_flag && name\_request(med.name, name);

break;

case 1:

char country[MAX\_LENGTH];

if (i == 0) {

printf("Введите страну-производителя лекарства\n(ШАБЛОН: Arabskie\_Emirati)\nСтрана:\t");

clear(stdin);

scanf\_s("%s", &country, MAX\_LENGTH);

}

is\_request\_flag = request\_flag = request\_flag && country\_request(med.country, country);

break;

case 2:

char date[MAX\_LENGTH];

if (i == 0) {

printf("Введите дату производства лекарства\n(ШАБЛОН [год.месяц.число] с нулями: 2012.09.02)\nДата:\t");

do {

date\_flag = true;

clear(stdin);

scanf\_s("%s", &date, MAX\_LENGTH);

for (int k = 0; k < LENGTH\_DATE; k++)

{

if (date[k] != '.' && (date[k] < '0' || date[k] > '9') || strlen(date) != LENGTH\_DATE || date[4] != '.' || date[7] != '.')

{

printf("ВВОД НЕВЕРЕН\nВВОДИТЕ, ПОЖАЛУЙСТА, ПО ШАБЛОНУ\nДата:\t");

date\_flag = false;

break;

}

}

} while (date\_flag == false);

}

is\_request\_flag = request\_flag = request\_flag && date\_request(med.date, date);

break;

case 3:

int pharmacy\_number;

if (i == 0) {

do {

printf("Введите номер апетки\n(ШАБЛОН: 12)\nНомер аптеки:\t");

clear(stdin);

scanf\_s("%d", &pharmacy\_number);

} while (pharmacy\_number <= 0);

}

is\_request\_flag = request\_flag = request\_flag && pharmacy\_number\_request(med.pharmacy\_number, pharmacy\_number);

break;

case 4:

int price;

char choice\_price;

if (i == 0) {

do {

price\_flag = true;

printf("Введите цену лекарства (ШАБЛОН: 200)\nЦена:\t");

clear(stdin);

scanf\_s("%d", &price);

printf("знак > (не дешевле), < (не дороже), = (равно) (ШАБЛОН: <)\nЗнак:\t");

clear(stdin);

scanf\_s("%c", &choice\_price, MAX\_LENGTH);

if (choice\_price != '<' && choice\_price != '>' && choice\_price != '=' || price < 0) //в случае, когда пользователь вводит текст, price == 0

{

printf("ВВОД НЕВЕРЕН\nВВОДИТЕ, ПОЖАЛУЙСТА, ПО ШАБЛОНУ\n");

price\_flag = false;

break;

}

} while (price\_flag == false);

}

is\_request\_flag = request\_flag = request\_flag && price\_request(med.price, price, choice\_price);

break;

}

}

}

if (is\_request\_flag == true) {

if (title\_flag == false) {

title\_flag = true;

printTitles();

}

if (request\_flag == true)

{

printf(" \* ");

printf(" %3d \* ", med.id);

printf(" %25.25s \* ", med.name);

printf(" %25.25s \* ", med.country);

printf(" %25.25s \* ", med.date);

for (int j = 0; j < MAX\_PHARMACIES; j++) {

if (med.pharmacy\_number[j] != 0)

printf("%3d, ", med.pharmacy\_number[j]); //расситано для 5 аптек

else

printf(" ");

}

printf(" \* ");

printf(" %25.2d \* ", med.price);

printf("\n%s\n", stars);

}

}

}

if (title\_flag == false) printf("По вашему запросу ничего не найдено\n");

fclose(file);

}

bool name\_request(char med\_name[], char name[]) {// ФУНКЦИЯ ЗАПРОСА ПО НАЗВАНИЮ ЛЕКАРСТВА

if (strstr(med\_name, name) != NULL)

return true;

return false;

}

bool country\_request(char med\_country[], char country[]) {// ФУНКЦИЯ ЗАПРОСА ПО СТРАНЕ ЛЕКАРСТВА

if (strcmp(med\_country, country) == NULL)

return true;

return false;

}

bool date\_request(char med\_date[], char date[]) { // ФУНКЦИЯ ЗАПРОСА ПО ДАТЕ ЛЕКАРСТВА

if (strcmp(med\_date, date) > 0)

return true;

return false;

}

bool price\_request(int med\_price, int price, char choice\_price) {// ФУНКЦИЯ ЗАПРОСА ПО ЦЕНЕ ЛЕКАРСТВА

if (choice\_price == '<') {

if (med\_price <= price)

return true;

return false;

}

else if (choice\_price == '>') {

if (med\_price >= price) return true;

return false;

}

else {

if (med\_price == price) return true;

return false;

}

}

bool pharmacy\_number\_request(int med\_pharmacy\_number[], int pharmacy\_number) {// ФУНКЦИЯ ЗАПРОСА ПО НАЛИЧИЮ ЛЕКАРСТВА В АПТЕКЕ

for (int j = 0; j < MAX\_PHARMACIES; j++)

if (med\_pharmacy\_number[j] == pharmacy\_number)

return true;

return false;

}

Тест всех функций

