**Федеральное государственное автономное образовательное учреждение**

**высшего образования**

**"Национальный исследовательский университет**

**"Высшая школа экономики"**

Московский институт электроники и математики им. А.Н. Тихонова НИУ ВШЭ

Департамент компьютерной инженерии

**Курс: Алгоритмизация и программирование**

|  |  |  |  |
| --- | --- | --- | --- |
| **Раздел** | **Max оценка** | **Итог. Оценка 1** | **Итог. Оценка 2** |
| Работа программы | 1 |  |  |
| Тесты | 1 |  |  |
| Правильность алгоритма | 3 |  |  |
| Ответы на вопросы | 2 |  |  |
| Дополнительное задание | 3 |  |  |

**ОТЧЕТ**

**по лабораторной работе № 6**

**Студент:\_ Иванилов Леонид Михайлович**

**Группа:\_БИВ231**

**Вариант:\_143 (12)**

**Руководитель:\_Ерохина Елена Альфредовна**

**Оценка:\_\_\_\_\_\_\_\_\_\_**

**Дата сдачи\_\_\_\_\_\_\_\_\_\_**

**Задания:**

1. После ввода массива структур программа ищет в нем следующую информацию (все данные полученном массиве должны быть различны)

(12). Города, где расположено максимальное число издательств разными названиями.

**II.** Модифицировать программу, написанную в первом пункте, таким образом, чтобы она вводила структуры из текстового файла и записывала результаты в файл.

**МОСКВА 2024**

**Листинг программы №1**

#define \_CRT\_SECURE\_NO\_WARNINGS  
#include <stdio.h>  
#include <string.h>  
#include <time.h>  
#define M 100  
#define L 81  
#define CLR while(getchar() != '\n');  
  
struct book{  
 char name[L], author[L];  
 int price, pages;  
 struct{  
 char name\_pub[L], city[L];  
 int year;  
 }publishing\_house;  
};  
  
int pr\_float(float min, float max, char name[]){  
 float k = 0;  
 do{  
 printf("Введите %s от %f до %f\n", name, min, max);  
 if (scanf("%f", &k) == 0){  
 CLR;  
 }  
 }while(k < min || k > max);  
 CLR;  
 return k;  
}  
  
float pr(int min, int max, char name[]){  
 int k = 0;  
 do{  
 printf("Введите %s от %d до %d\n", name, min, max);  
 if (scanf("%d", &k) == 0){  
 CLR;  
 }  
 }while(k < min || k > max);  
 CLR;  
 return k;  
}  
  
void pr2(char s[], const char name[]) {  
 do {  
 printf("Введите %s: \n", name);  
 gets(s);  
 } while (strcmp(s, "") == 0);  
}  
int input\_structs(struct book arr[]) {  
 int k, i;  
 time\_t t;  
 time(&t);  
 struct tm \*today = localtime(&t);  
 int year = today->tm\_year + 1900;  
 k = pr(1, M, "число книг");  
 for (i = 0; i < k; i++)  
 {  
 printf("Книга #%i \n", i + 1);  
 pr2(arr[i].name, "название книги");  
 arr[i].price = pr\_float(100.0, 50000.0, "цену");  
 arr[i].pages = pr(5, 5000, "количество страниц");  
 pr2(arr[i].author, "автор");  
 arr[i].publishing\_house.year = pr(1450, year, "год");  
 pr2(arr[i].publishing\_house.name\_pub, "название издательства");  
 pr2(arr[i].publishing\_house.city, "город");  
 printf("\n");  
 }  
 return k;  
}  
  
void output(struct book arr[], int k) {  
 for (int i = 0; i < k; i++) {  
 printf("Книга #%i \n", i + 1);  
 printf("Название: %s\n", arr[i].name);  
 printf("Цена: %i\n", arr[i].price);  
 printf("Число страниц: %i\n", arr[i].pages);  
 printf("Автор: %s\n", arr[i].author);  
 printf("Издательство: \n Год издания: %i\n Назвние издания: %s\n Город: %s\n", arr[i].publishing\_house.year, arr[i].publishing\_house.name\_pub, arr[i].publishing\_house.city);  
 time\_t start\_time = time(NULL); // получаем текущее время  
 while (difftime(time(NULL), start\_time) < 2) {  
 // ожидаем, пока не пройдет 10 секунд  
 }  
  
 }  
}  
  
void find\_cities(struct book arr[], int k){  
 int i, j, kf = 0;  
 char cities[M][L];  
 for(i = 0; i < k; i++){  
 int isUnique = 1;  
 for(j = 0; j < k && isUnique == 1; j++){  
 if (strcmp(arr[i].publishing\_house.city, cities[j]) == 0){  
 isUnique = 0;  
 }  
 }  
 if (isUnique == 1){  
 strcpy(cities[kf++], arr[i].publishing\_house.city);  
 }  
 }  
  
 printf("\n");  
  
  
 int counter[kf];  
 for (i = 0; i < kf; i++){  
 int k1 = 0;  
 char names[k][L];  
 counter[i] = 0;  
 for(j = 0; j < k; j++){  
 if(strcmp(cities[i], arr[j].publishing\_house.city) == 0){  
 int isUnique = 1;  
 for(int m = 0; m < k && isUnique == 1; m++){  
 if(strcmp(arr[j].publishing\_house.name\_pub, names[m]) == 0){  
 isUnique = 0;  
 }  
 }  
 if (isUnique == 1){  
 strcpy(names[k1++], arr[j].publishing\_house.name\_pub);  
 counter[i]+=1;  
 }  
  
 }  
 }  
  
  
 }  
 int maxx = 0;  
 for (int h = 0; h < kf; h++){  
 if(counter[h] > maxx){  
 maxx = counter[h];  
 }  
 }  
 for(int h = 0; h < kf; h++){  
 if(counter[h] == maxx)  
 printf("город с максимальным числом читательских издательств %s\n", cities[h]);  
 }  
}  
  
  
  
int main(void) {  
 struct book arr[M];  
 int n;  
 n = input\_structs(arr);  
 output(arr, n);  
 find\_cities(arr, n);  
 return 0;  
}

Листинг программы №2

#define \_CRT\_SECURE\_NO\_WARNINGS  
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
#include <time.h>  
#define M 100  
#define L 81  
#define FCLR while(getc(file\_from)!='\n');  
  
struct book{  
 char name[L], author[L];  
 int price, pages;  
 struct{  
 char name\_pub[L], city[L];  
 int year;  
 }publishing\_house;  
};  
  
void n\_to\_0(char \*s) {  
 for (; \*s && \*s != '\n'; s++);  
 \*s = '\0';  
}  
  
int input\_num\_float(float min, float max, FILE \*file\_from) {  
 float k = 0.0;  
 int c;  
 do {  
 c = fscanf(file\_from, "%f", &k);  
 FCLR;  
 } while (k < min || k > max || !c);  
 return k;  
}  
  
int input\_num(int min, int max, FILE \*file\_from) {  
 int k = 0, c;  
 do {  
 c = fscanf(file\_from, "%i", &k);  
 FCLR;  
 } while (k < min || k > max || !c);  
 return k;  
}  
  
void input\_str(char s[], FILE \*file\_from) {  
 do {  
 fgets(s, L, file\_from);  
 } while (strcmp(s, "") == 0);  
 n\_to\_0(s);  
}  
  
  
int input\_structs(struct book arr[]) {  
 FILE \*file\_from;  
 char filename\_in[L];  
 printf("Введите имя файла с данными: ");  
 gets(filename\_in);  
 file\_from = fopen(filename\_in, "r");  
 while (!(file\_from)) {  
 puts("Файл не найден. ");  
 gets(filename\_in);  
 file\_from = fopen(filename\_in, "r");  
 }  
  
  
 int k, c;  
 time\_t t;  
 time(&t);  
 struct tm \*today = localtime(&t);  
 int year = today->tm\_year + 1900;  
 c = fscanf(file\_from, "%i", &k);  
  
 if (c != 1) {  
 puts("Файл пустой.");  
 exit(1);  
 }  
 struct book \*it;  
 for (it = arr; it < arr + k; it++)  
 {  
 input\_str(it->name, file\_from);  
 it->price = input\_num\_float(100.0, 50000.0, file\_from);  
 it->pages = input\_num(5, 5000, file\_from);  
 input\_str(it->author, file\_from);  
 it->publishing\_house.year = input\_num(1450, year, file\_from);  
 input\_str(it->publishing\_house.name\_pub, file\_from);  
 input\_str(it->publishing\_house.city, file\_from);  
 }  
 fclose(file\_from);  
 return k;  
}  
  
void output(struct book arr[], int k) {  
 struct book \*it;  
 int j;  
 for (it = arr, j = 1; j <= k; it++, j++) {  
 printf("Книга #%i \n", j);  
 printf("Название: %s\n", it->name);  
 printf("Цена: %i\n", it->price);  
 printf("Число страниц: %i\n", it->pages);  
 printf("Автор: %s\n", it->author);  
 printf("Издательство: \n Год издания: %i\n Назвние издания: %s\n Город: %s\n", it->publishing\_house.year, it->publishing\_house.name\_pub, it->publishing\_house.city);  
 time\_t start\_time = time(NULL); // получаем текущее время  
 while (difftime(time(NULL), start\_time) < 2) {  
 // ожидаем, пока не пройдет 2 секунды  
 }  
 printf("\n");  
 }  
}  
  
void find\_cities(struct book arr[], int k){  
 int i, j, kf = 0;  
 char cities[M][L];  
 struct book \*it;  
 for(it = arr; it < arr + k; it++){  
 int isUnique = 1;  
 for(j = 0; j < k && isUnique == 1; j++){  
 if (strcmp(it->publishing\_house.city, cities[j]) == 0){  
 isUnique = 0;  
 }  
 }  
 if (isUnique == 1){  
 strcpy(cities[kf++], it->publishing\_house.city);  
 }  
 }  
  
 printf("\n");  
  
  
 int counter[kf];  
 for (i = 0; i < kf; i++){  
 int k1 = 0;  
 char names[k][L];  
 counter[i] = 0;  
 struct book \*it2;  
 for(it2 = arr; it2 < arr + k; it2++){  
 if(strcmp(cities[i], it2->publishing\_house.city) == 0){  
 int isUnique = 1;  
 for(int m = 0; m < k && isUnique == 1; m++){  
 if(strcmp(it2->publishing\_house.name\_pub, names[m]) == 0){  
 isUnique = 0;  
 }  
 }  
 if (isUnique == 1){  
 strcpy(names[k1++], it2->publishing\_house.name\_pub);  
 counter[i]+=1;  
 }  
  
 }  
 }  
  
  
 }  
 FILE \*file\_out;  
 char filename\_out[L];  
 puts("Введите файл для вывода результата: ");  
 gets(filename\_out);  
 file\_out = fopen(filename\_out, "w");  
  
  
 int maxx = 0;  
 for (int h = 0; h < kf; h++){  
 if (counter[h] > maxx){  
 maxx = counter[h];  
 }  
 }  
 for (h = 0; h < kf; h++){  
 if (counter[h] == maxx) {  
 printf("Город с максимальным числом читательских издательств(%s), а именнно %d \n", cities[h], counter[h]);  
 fprintf(file\_out, "Город с максимальным числом читательских издательств(%s), а именнно %d\n", cities[h], counter[h]);  
 }  
 }  
}  
  
  
  
int main(void) {  
 struct book arr[M];  
 int n;  
 n = input\_structs(arr);  
 output(arr, n);  
 find\_cities(arr, n);  
 return 0;  
}

**Тестовые данные**

1. **На вход**

Книга #1

Введите название книги:

Капитанская дочка

Введите цену от 100.000000 до 50000.000000

500

Введите количество страниц от 5 до 5000

67

Введите автор:

А.С.Пушкин

Введите год от 1450 до 2024

2000

Введите название издательства:

Просвящение

Введите город:

Москва

Книга #2

Введите название книги:

Евгений Онегин

Введите цену от 100.000000 до 50000.000000

350

Введите количество страниц от 5 до 5000

105

Введите автор:

А.С.Пушкин

Введите год от 1450 до 2024

2001

Введите название издательства:

Просвящение

Введите город:

Москва

Книга #3

Введите название книги:

Борис Годунов

Введите цену от 100.000000 до 50000.000000

800

Введите количество страниц от 5 до 5000

78

Введите автор:

А.С.Пушкин

Введите год от 1450 до 2024

2005

Введите название издательства:

Надежда

Введите город:

Санкт-Петербург

**Вывод**

Книга #1

Название: Капитанская дочка

Цена: 500

Число страниц: 67

Автор: А.С.Пушкин

Издательство:

Год издания: 2000

Назвние издания: Просвящение

Город: Москва

Книга #2

Название: Евгений Онегин

Цена: 350

Число страниц: 105

Автор: А.С.Пушкин

Издательство:

Год издания: 2001

Назвние издания: Просвящение

Город: Москва

Книга #3

Название: Борис Годунов

Цена: 800

Число страниц: 78

Автор: А.С.Пушкин

Издательство:

Год издания: 2005

Назвние издания: Надежда

Город: Санкт-Петербург

1. **Файл на вход**

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Введите имя файла с данными: test1.txt

Введите файл для вывода результата: 123.txt

На выход:

Город с максимальным числом читательских издательств(Москва), а именнно 1

Город с максимальным числом читательских издательств(Санкт-Петербург), а именнно 1

Дополнительное Задание:

Листинг программы:

#define \_CRT\_SECURE\_NO\_WARNINGS  
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
#include <time.h>  
#define M 100  
#define L 81  
#define FCLR while(getc(file\_from)!='\n');  
  
struct book{  
 char name[L], author[L];  
 int price, pages;  
 struct{  
 char name\_pub[L], city[L];  
 int year;  
 }publishing\_house;  
};  
  
void n\_to\_0(char \*s) {  
 for (; \*s && \*s != '\n'; s++);  
 \*s = '\0';  
}  
  
int input\_num\_float(float min, float max, FILE \*file\_from) {  
 float k = 0.0;  
 int c;  
 do {  
 c = fscanf(file\_from, "%f", &k);  
 FCLR;  
 } while (k < min || k > max || !c);  
 return k;  
}  
  
int input\_num(int min, int max, FILE \*file\_from) {  
 int k = 0, c;  
 do {  
 c = fscanf(file\_from, "%i", &k);  
 FCLR;  
 } while (k < min || k > max || !c);  
 return k;  
}  
  
void input\_str(char s[], FILE \*file\_from) {  
 do {  
 fgets(s, L, file\_from);  
 } while (strcmp(s, "") == 0);  
 n\_to\_0(s);  
}  
  
  
int input\_structs(struct book arr[]) {  
 FILE \*file\_from;  
 char filename\_in[L];  
 printf("Введите имя файла с данными: ");  
 gets(filename\_in);  
 file\_from = fopen(filename\_in, "r");  
 while (!(file\_from)) {  
 puts("Файл не найден. ");  
 printf("Введите имя файла с данными: ");  
 gets(filename\_in);  
 file\_from = fopen(filename\_in, "r");  
 }  
  
  
 int k, c;  
 time\_t t;  
 time(&t);  
 struct tm \*today = localtime(&t);  
 int year = today->tm\_year + 1900;  
 k = input\_num(1, 10, file\_from);  
  
 if (!k) {  
 puts("Файл пустой.");  
 exit(1);  
 }  
 struct book \*it;  
 for (it = arr; it < arr + k; it++)  
 {  
 input\_str(it->name, file\_from);  
 it->price = input\_num\_float(100.0, 50000.0, file\_from);  
 it->pages = input\_num(5, 5000, file\_from);  
 input\_str(it->author, file\_from);  
 it->publishing\_house.year = input\_num(1450, year, file\_from);  
 input\_str(it->publishing\_house.name\_pub, file\_from);  
 input\_str(it->publishing\_house.city, file\_from);  
 }  
 fclose(file\_from);  
 return k;  
}  
  
void output(struct book arr[], int k) {  
 struct book \*it;  
 int j;  
 for (it = arr, j = 1; j <= k; it++, j++) {  
 printf("Книга #%i \n", j);  
 printf("Название: %s\n", it->name);  
 printf("Цена: %i\n", it->price);  
 printf("Число страниц: %i\n", it->pages);  
 printf("Автор: %s\n", it->author);  
 printf("Издательство: \n Год издания: %i\n Назвние издания: %s\n Город: %s\n", it->publishing\_house.year, it->publishing\_house.name\_pub, it->publishing\_house.city);  
 time\_t start\_time = time(NULL); // получаем текущее время  
 while (difftime(time(NULL), start\_time) < 2) {  
 // ожидаем, пока не пройдет 2 секунды  
 }  
 printf("\n");  
 }  
}  
  
void find\_cities(struct book arr[], int k){  
 int i, j, kf = 0;  
 char authors[M][L];  
 struct book \*it;  
 for(it = arr; it < arr + k; it++){  
 int isUnique = 1;  
 for(j = 0; j < k && isUnique == 1; j++){  
 if (strcmp(it->author, authors[j]) == 0){  
 isUnique = 0;  
 }  
 }  
 if (isUnique == 1){  
 strcpy(authors[kf++], it->author);  
 }  
 }  
  
 printf("\n");  
  
  
 int counter[kf];  
 for (i = 0; i < kf; i++){  
 int k1 = 0;  
 char names[k][L];  
 counter[i] = 0;  
 struct book \*it2;  
 for(it2 = arr; it2 < arr + k; it2++){  
 if(strcmp(authors[i], it2->author) == 0){  
 int isUnique = 1;  
 for(int m = 0; m < k && isUnique == 1; m++){  
 if(strcmp(it2->name, names[m]) == 0){  
 isUnique = 0;  
 }  
 }  
 if (isUnique == 1){  
 strcpy(names[k1++], it2->name);  
 counter[i]+=1;  
 }  
  
 }  
 }  
  
  
 }  
 FILE \*file\_out;  
 char filename\_out[L];  
 puts("Введите файл для вывода результата: ");  
 gets(filename\_out);  
 file\_out = fopen(filename\_out, "w");  
  
  
 int maxx = 0;  
 for (int h = 0; h < kf; h++){  
 if (counter[h] > maxx){  
 maxx = counter[h];  
 }  
 }  
 for (h = 0; h < kf; h++){  
 if (counter[h] == maxx) {  
 printf("Автор с максимальным числом книг(%s), а именнно %d \n", authors[h], counter[h]);  
 fprintf(file\_out, "автор с максимальным числом книг(%s), а именнно %d\n", authors[h], counter[h]);  
 }  
 }  
}  
  
  
  
int main(void) {  
 struct book arr[M];  
 int n;  
 n = input\_structs(arr);  
 output(arr, n);  
 find\_cities(arr, n);  
 return 0;  
}

Тесты:

1)

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

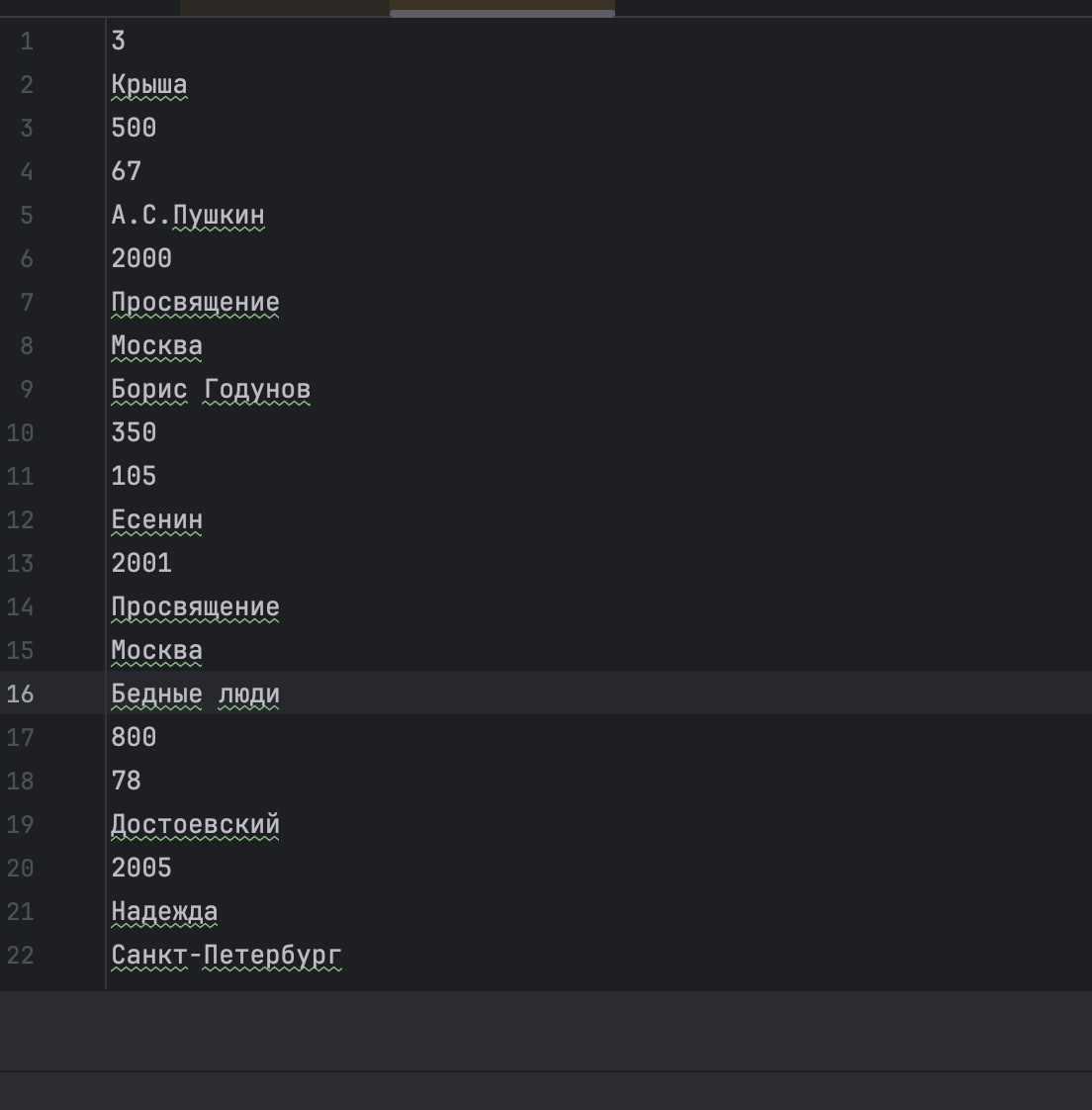
Автоматически созданное описание

Вывод:

Автор с максимальным числом книг(А.С.Пушкин), а именнно 1

Автор с максимальным числом книг(Достоевский), а именнно 1

2)



Вывод программы:

Автор с максимальным числом книг(А.С.Пушкин), а именнно 1

Автор с максимальным числом книг(Есенин), а именнно 1

Автор с максимальным числом книг(Достоевский), а именнно 1