**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**

**ОТЧЁТ**

**по лабораторной работе №6**

|  |  |  |
| --- | --- | --- |
| **Раздел** | **Max оценка** | **Итоговая оценка** |
| **Работа программы** | **1** |  |
| **Тесты** | **1** |  |
| **Правильность алгоритма** | **3** |  |
| **Ответы на вопросы** | **2** |  |
| **Доп. задание** | **3** |  |

**Студент: Кривушкина Елизавета Юрьевна**

**Группа: БИВ231**

**Вариант: №146 (Задание 15)**

**Руководитель: Моисеев Михаил Васильевич**

**Оценка:**

**Дата сдачи: 24.01.2023**

**Москва 2023**

Оглавление

[Задание 3](#_Toc152682050)

[Листинг 4](#_Toc152682051)

[Тесты 5](#_Toc152682052)

# Задание

15. Города, в которых расположено максимальное число издательств.

Ввод, вывод данных, формирование массива результатов оформить

как отдельные функции.

Проверку существования результата произвести в

главной программе.

К элементам массива структур обращаться при помощи

* индекса в первой программе
* указателя во второй программе.

При сравнении строк использовать стандартные функции для обработки

символьных строк.

Для ввода данных и вывода результатов использовать функции

* *scanf, gets*, *printf, puts* в первой программе
* *fscanf, fgets, fputs, fprintf* во второй программе.

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

# Листинг

#include <stdio.h>  
#include <string.h>  
#define CLR while (getchar() != '\n')  
  
struct publishingHouse {  
 char title[20], city[20];  
 int year;  
};  
struct book {  
 char name[60], author[20];  
 int price, numberOfPages;  
 struct publishingHouse publishing;  
};  
  
void input(struct book papapa[], int \*n) {  
 int cheker;  
 do {  
 printf("Введите количество структур: ");  
 cheker = scanf("%d", n); CLR;  
 } while (cheker != 1 || \*n <=0);  
  
 for(int i=0;i<\*n;i++) {  
// printf("\nВведите название: ");  
// gets(papapa[i].name);  
// do {  
// printf("\nВведите цену: ");  
// scanf("%d", &papapa[i].price); CLR;  
// } while (papapa[i].price<=0);  
// do {  
// printf("\nВведите число страниц: ");  
// scanf("%d", &papapa[i].numberOfPages); CLR;  
// } while (papapa[i].numberOfPages<=0);  
// printf("\nВведите автора: ");  
// gets(papapa[i].author);  
// printf("\nВведите параметры для издательства");  
// do {  
// printf("\n\tВведите год издания: ");  
// scanf("%d", &papapa[i].publishing.year); CLR;  
// } while (papapa[i].publishing.year<=1800 || papapa[i].publishing.year>=2024);  
 printf("\n\tВведите название: ");  
 gets(papapa[i].publishing.title);  
 printf("\n\tВведите город: ");  
 gets(papapa[i].publishing.city);  
 }  
}  
  
void output(struct book papapa[], int n) {  
 for(int i=0;i<n;i++) {  
 printf("\nНазвание: %s\nЦена: %d", papapa[i].name, papapa[i].price);  
 printf("\nЧисло страниц: %d\nАвтор: %s", papapa[i].numberOfPages, papapa[i].author);  
 printf("\nДанные издательства\n\tГод издания: %d", papapa[i].publishing.year);  
 printf("\n\tНазвание: %s\n\tГород: %s", papapa[i].publishing.title, papapa[i].publishing.city);  
 }  
}  
  
void answer(struct book \*papapa, int n) {  
 int i,j,q, countCities=0, countPublishings=0, maximum=0, flag, flag1, countPublishingsArray[20];  
 char cities[20][20], publishings[20][20];  
 struct book \*pointer;  
 for(i=0;i<n;i++,papapa++) { //проходим по всем городам  
 flag=0;  
 for(j=0;j<countCities && (!flag);j++) {  
 if(!(strcmp(papapa->publishing.city, cities[j]))) //есть ли такой же город в массиве  
 flag = 1;  
 }  
 if(!flag) { //если такого же города нет  
 strcpy(cities[countCities], papapa->publishing.city); // копируем его в массив  
 pointer=papapa;  
 for(j=i;j<n;j++,pointer++) { //идём по издательствам  
 if( !(strcmp(pointer->publishing.city, papapa->publishing.city)) ) { // того же города  
 flag1=0;  
 for(q=0; q<countPublishings && (!flag1) ;q++) {  
 if(!(strcmp(pointer->publishing.title, publishings[q]))) //есть ли такое же издательство в массиве  
 flag1=1;  
 }  
 if(!flag1) { //если нет  
 strcpy(publishings[countPublishings++], pointer->publishing.title); //копируем  
 }  
 }  
 }  
 }  
 countPublishingsArray[countCities++]=countPublishings; //запоминаем количество издательств у каждого города  
 if(countPublishings>maximum) maximum=countPublishings; //ищем максимум  
 for(q=0;q<countPublishings;q++) {  
 strcpy(publishings[q], "\0"); //очищаем массив издательств  
 }  
 countPublishings=0; //обнуляем счётчик массива издательств  
 }  
 printf("\nМаксимальное число издательств: %d\nВ городах:", maximum);  
 for(i=0;i<countCities;i++) {  
 if(countPublishingsArray[i]==maximum) {  
 printf("\n");  
 puts(cities[i]);  
 }  
 }  
}  
  
int main() {  
 int n, answerFinally;  
 struct book papapa[20];  
 input(papapa, &n);  
// output(papapa, n);  
 answer(papapa, n);  
 return 0;  
}

#include <stdio.h>  
#include <string.h>  
#define CLR while (getchar() != '\n')  
  
struct publishingHouse {  
 char title[20], city[20];  
 int year;  
};  
struct book {  
 char name[60], author[20];  
 int price, numberOfPages;  
 struct publishingHouse publishing;  
};  
  
void input(struct book \*b, int \*n, FILE \*file) {  
 char \*cheker, filename[100];  
 FILE \*fp;  
 printf("Введите имя файла: ");  
 scanf("%s", filename);  
 fp = fopen(filename, "r");  
 // /Users/liz0khse/CLionProjects/laba6task2/test  
 if (fp == NULL) {  
 perror("Ошибка открытия файла");  
 return;  
 }  
 do {  
 fscanf(fp, "%d", n);  
 } while (\*n <=0);  
 for(int i=0;i<\*n;i++) {  
 fscanf(fp, "%s %s %d %d %s %s %d",  
 b->name, b->author, &b->price, &b->numberOfPages,  
 b->publishing.title, b->publishing.city, &b++->publishing.year);  
 }  
}  
  
void writeToFile(const char \*filename, int maximum, char cities[][20], int countCities, int countPublishingsArray[]) {  
 FILE \*fp;  
 int i;  
  
 fp = fopen(filename, "w");  
 if (fp == NULL) {  
// perror("Ошибка открытия файла");  
 return;  
 }  
  
 fprintf(fp, "\nMax count publishings: %d\nIn cities:", maximum);  
 for (i = 0; i < countCities; i++) {  
 if (countPublishingsArray[i] == maximum) {  
 fprintf(fp, "\n%s", cities[i]);  
 }  
 }  
  
 fclose(fp);  
}  
  
void answer(struct book \*papapa, int n) {  
 int i,j,q, countCities=0, countPublishings=0, maximum=0, flag, flag1, countPublishingsArray[20];  
 char cities[20][20], publishings[20][20];  
 struct book \*pointer;  
 for(i=0;i<n;i++,papapa++) { //проходим по всем городам  
 flag=0;  
 for(j=0;j<countCities && (!flag);j++) {  
 if(!(strcmp(papapa->publishing.city, cities[j]))) //есть ли такой же город в массиве  
 flag = 1;  
 }  
 if(!flag) { //если такого же города нет  
 strcpy(cities[countCities], papapa->publishing.city); // копируем его в массив  
 pointer=papapa;  
 for(j=i;j<n;j++,pointer++) { //идём по издательствам  
 if( !(strcmp(pointer->publishing.city, papapa->publishing.city)) ) { // того же города  
 flag1=0;  
 for(q=0; q<countPublishings && (!flag1) ;q++) {  
 if(!(strcmp(pointer->publishing.title, publishings[q]))) //есть ли такое же издательство в массиве  
 flag1=1;  
 }  
 if(!flag1) { //если нет  
 strcpy(publishings[countPublishings++], pointer->publishing.title); //копируем  
 }  
 }  
 }  
 }  
 countPublishingsArray[countCities++]=countPublishings; //запоминаем количество издательств у каждого города  
 if(countPublishings>maximum) maximum=countPublishings; //ищем максимум  
 for(q=0;q<countPublishings;q++) {  
 strcpy(publishings[q], "\0"); //очищаем массив издательств  
 }  
 countPublishings=0; //обнуляем счётчик массива издательств  
 }  
 writeToFile("/Users/liz0khse/CLionProjects/laba6task2/rrr", maximum, cities, countCities, countPublishingsArray);  
}  
  
int main() {  
 FILE \*fp;  
 fp = fopen("/Users/liz0khse/CLionProjects/laba6task2/test", "r");  
 if (fp == NULL) {  
 puts("Ошибка открытия файла");  
 return 1;  
 }  
 fclose(fp);  
 int n, answerFinally;  
 struct book papapa[20];  
 input(papapa, &n, fp);  
// output(papapa, n, fp);  
 answer(papapa, n);  
 return 0;  
}

#include <stdio.h>  
#include <string.h>  
#define CLR while (getchar() != '\n')  
  
struct publishingHouse {  
 char title[20], city[20];  
 int year;  
};  
struct book {  
 char name[60], author[20];  
 int price, numberOfPages;  
 struct publishingHouse publishing;  
};  
  
void input(struct book \*b, int \*n, FILE \*file) {  
 char \*cheker, filename[100];  
 FILE \*fp;  
 printf("Введите имя файла: ");  
 scanf("%s", filename);  
 fp = fopen(filename, "r");  
 // /Users/liz0khse/CLionProjects/laba6task2/test  
 if (fp == NULL) {  
 perror("Ошибка открытия файла");  
 return;  
 }  
 do {  
 fscanf(fp, "%d", n);  
 } while (\*n <=0);  
 for(int i=0;i<\*n;i++) {  
 fscanf(fp, "%s %s %d %d %s %s %d",  
 b->name, b->author, &b->price, &b->numberOfPages,  
 b->publishing.title, b->publishing.city, &b++->publishing.year);  
 }  
}  
  
void writeToFile(const char \*filename, int maximum, char cities[][20], int countCities, int countPublishingsArray[]) {  
 FILE \*fp;  
 int i;  
  
 fp = fopen(filename, "w");  
 if (fp == NULL) {  
// perror("Ошибка открытия файла");  
 return;  
 }  
  
 fprintf(fp, "\nMax count publishings: %d\nIn cities:", maximum);  
 for (i = 0; i < countCities; i++) {  
 if (countPublishingsArray[i] == maximum) {  
 fprintf(fp, "\n%s", cities[i]);  
 }  
 }  
  
 fclose(fp);  
}  
  
void answer(struct book \*papapa, int n) {  
 int i,j,q, countCities=0, countPublishings=0, maximum=0, flag, flag1, countPublishingsArray[20];  
 char cities[20][20], publishings[20][20];  
 struct book \*pointer;  
 for(i=0;i<n;i++,papapa++) { //проходим по всем городам  
 flag=0;  
 for(j=0;j<countCities && (!flag);j++) {  
 if(!(strcmp(papapa->publishing.city, cities[j]))) //есть ли такой же город в массиве  
 flag = 1;  
 }  
 if(!flag) { //если такого же города нет  
 strcpy(cities[countCities], papapa->publishing.city); // копируем его в массив  
 pointer=papapa;  
 for(j=i;j<n;j++,pointer++) { //идём по издательствам  
 if( !(strcmp(pointer->publishing.city, papapa->publishing.city)) ) { // того же города  
 flag1=0;  
 for(q=0; q<countPublishings && (!flag1) ;q++) {  
 if(!(strcmp(pointer->publishing.title, publishings[q]))) //есть ли такое же издательство в массиве  
 flag1=1;  
 }  
 if(!flag1) { //если нет  
 strcpy(publishings[countPublishings++], pointer->publishing.title); //копируем  
 }  
 }  
 }  
 }  
 countPublishingsArray[countCities++]=countPublishings; //запоминаем количество издательств у каждого города  
 if(countPublishings>maximum) maximum=countPublishings; //ищем максимум  
 for(q=0;q<countPublishings;q++) {  
 strcpy(publishings[q], "\0"); //очищаем массив издательств  
 }  
 countPublishings=0; //обнуляем счётчик массива издательств  
 }  
 writeToFile("/Users/liz0khse/CLionProjects/laba6task2/rrr", maximum, cities, countCities, countPublishingsArray);  
}  
  
int main() {  
 FILE \*fp;  
 fp = fopen("/Users/liz0khse/CLionProjects/laba6task2/test", "r");  
 if (fp == NULL) {  
 puts("Ошибка открытия файла");  
 return 1;  
 }  
 fclose(fp);  
 int n, answerFinally;  
 struct book papapa[20];  
 input(papapa, &n, fp);  
// output(papapa, n, fp);  
 answer(papapa, n);  
 return 0;  
}

# Тесты

|  |  |  |
| --- | --- | --- |
| **№** | **Исходные данные** | **Результат** |
| 1 | **n=1**   |  | | --- | | **t** | | **g** | | Максимальное число издательств: 1  В городах:  g |
| 2 | **n=2**   |  |  | | --- | --- | | **t** | **t** | | **g** | **g** | | Максимальное число издательств: 1  В городах:  g |
| 3 | **n=3**   |  |  |  | | --- | --- | --- | | **t** | **y** | **r** | | **g** | **h** | **h** | | Максимальное число издательств: 2  В городах:  h |
| 4 | **n=4**   |  |  |  |  | | --- | --- | --- | --- | | **t** | **y** | **r** | **t** | | **g** | **h** | **h** | **g** | | Максимальное число издательств: 2  В городах:  h |
| 5 | **n=5**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **e** | **y** | **g** | **j** | **q** | | **d** | **d** | **b** | **b** | **a** | | Максимальное число издательств: 2  В городах:  d  b |
| 6 | **n=4**   |  |  |  |  | | --- | --- | --- | --- | | **q** | **w** | **e** | **w** | | **a** | **a** | **a** | **a** | | Максимальное число издательств: 3  В городах:  a |

# Листинг

#include <stdio.h>

void input() {

}

void output() {

}

void townsAnswer() {

}

int main() {

return 0;

}