



**«Московский государственный технический университет  
имени Н.Э. Баумана»  
(национальный исследовательский университет)  
(МГТУ им. Н.Э. Баумана)**

---

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ  
КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)


**О т ч е т**

**по домашнему заданию № 3**

**Название домашнего задания:** Динамические структуры данных и  
файловая система.

**Дисциплина:** Алгоритмизация и программирование.

Студент гр. ИУ6-13Б \_\_\_\_\_ **С.М Соболев**  
(Подпись, дата) (И.О. Фамилия)

Преподаватель \_\_\_\_\_  **О.А. Веселовская**  
(Подпись, дата) (И.О. Фамилия)

Москва, 2025

## Часть 1. Списки.

**Цель:** Разработка программы, которая создает список по типу очереди из вводимых букв и цифр. Удалить цифры, начиная с 5. Протестировать программу.

**Задание:** Создать список по типу очереди из вводимых букв и цифр. Удалить цифры, начиная с 5. При завершении программы освободить динамическую память.

Схема алгоритма программы:

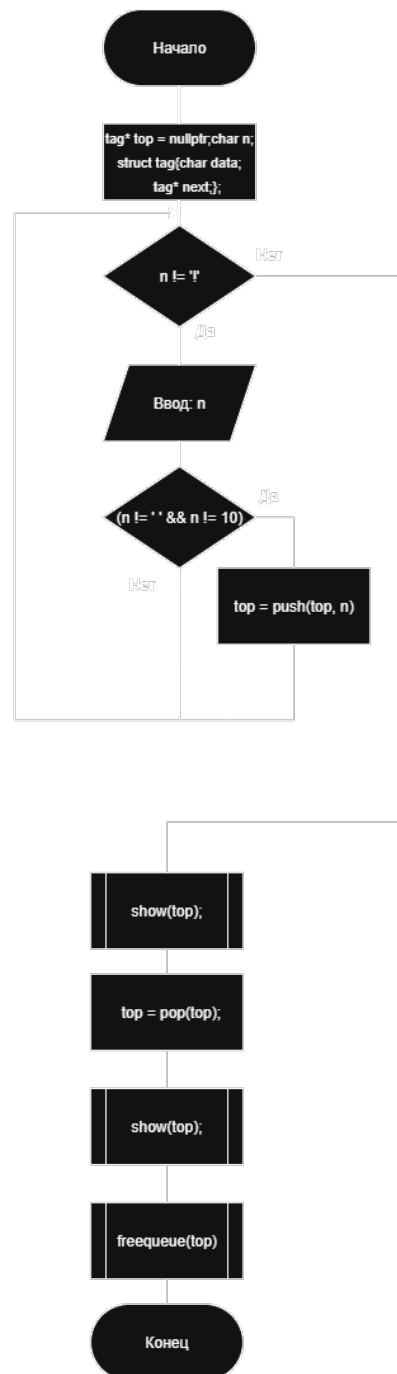


Рисунок 1 – Схема алгоритма основной программы

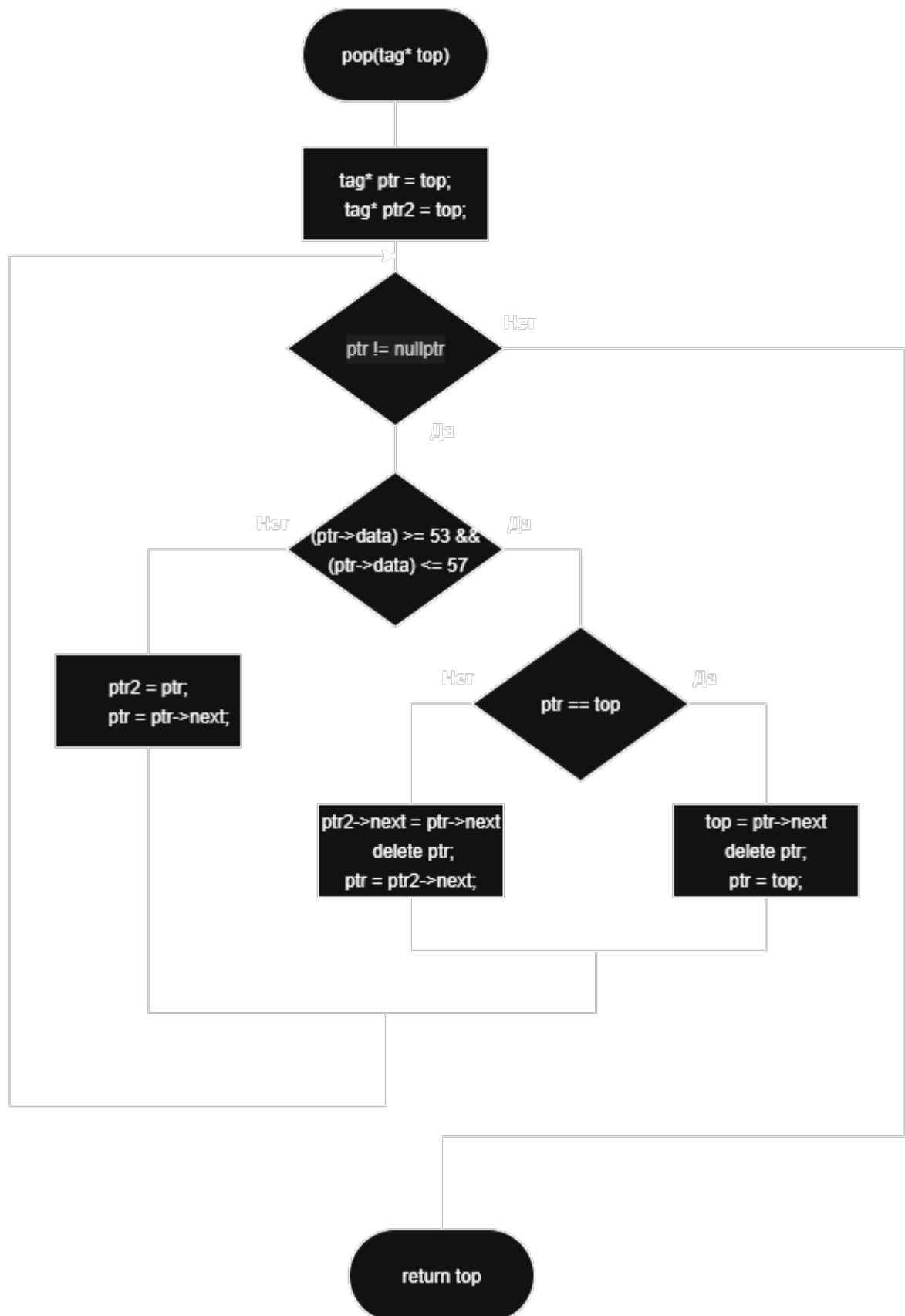


Рисунок 2 – Схема алгоритма подпрограммы `pop`

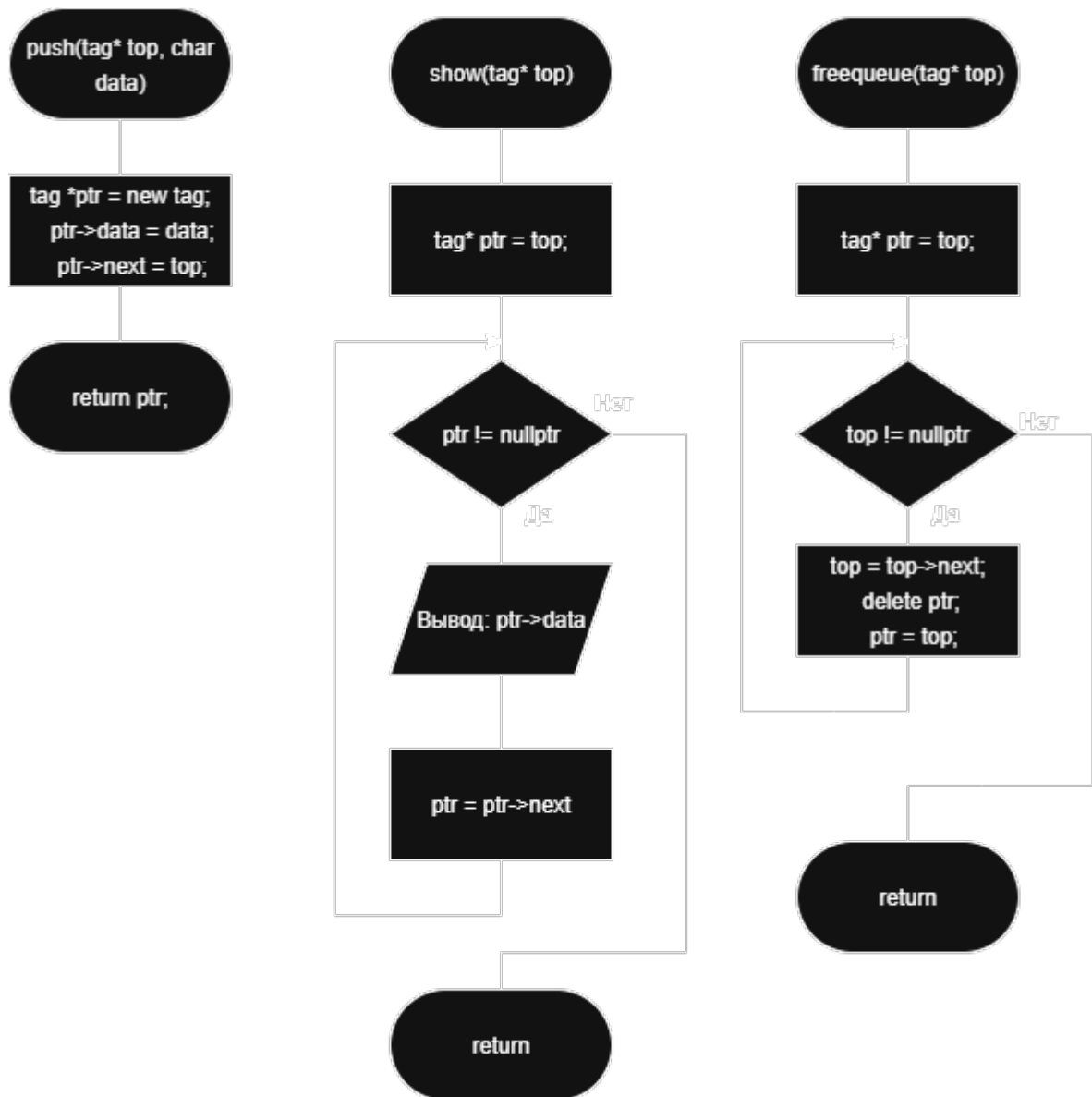


Рисунок 3 – Схема алгоритма подпрограмм push, show, freequeue

Код программы:

```

61 int main(){
62     tag* top = nullptr;
63     char n;
64     puts("Введите букву или цифру, чтобы закончить введите знак !");
65     while (scanf("%c", &n), n != '!'){
66         if (n != ' ' && n != 10)
67             top = push(top, n);
68     }
69     puts("-----");
70     show(top);
71     top = pop(top);
72     puts("\n-----");
73     show(top);
74
75     freequeue(top);
76
77     return 0;
78 }

```

Рисунок 4 – Код основной программы

```

tag* pop(tag* top){
    tag* ptr = top;
    tag* ptr2 = top;
    while(ptr != nullptr){
        if ((ptr->data) >= 53 && (ptr->data) <= 57){
            if (ptr == top){
                top = ptr->next;
                delete ptr;
                ptr = top;
            }
            else{
                ptr2->next = ptr->next;
                delete ptr;
                ptr = ptr2->next;
            }
        }
        else{
            ptr2 = ptr;
            ptr = ptr->next;
        }
    }
    return top;
}

void freequeue(tag* top){
    tag* ptr = top;
    while (top != nullptr){
        top = top->next;
        delete ptr;
        ptr = top;
    }
}

```

Рисунок 5 – Код подпрограмм pop, freequeue

```

tag* push(tag* top, char data){
    tag *ptr = new tag;

    ptr->data = data;
    ptr->next = top;
    return ptr;
}

void show(tag* top){
    tag* ptr = top;
    while (ptr != nullptr) {
        printf("%c ", ptr->data);
        ptr = ptr->next;
    }
}

```

Рисунок 6 - Код подпрограмм push, show

## Тестирование.

| Входные данные      | Выходные данные  |
|---------------------|--|
| 3 d e 5 6 7 8 !     | <pre>-----<br/>8 7 6 5 e d 3<br/>-----<br/>e d 3</pre>           |
| h h h 9 9 9 1 2 1 ! | <pre>-----<br/>1 2 1 9 9 9 h h h<br/>-----<br/>1 2 1 h h h</pre> |

**Вывод:** в ходе выполнения задания была разработана и протестирована программа для удаления цифр больше 5 из списка по типу очереди. Задание помогло закрепить знания о списках.

## Часть 2. Обработка строк.

**Цель:** Освоить практические навыки работы с файлами последовательного доступа в языке C++, включая чтение структурированных данных из файла, их обработку по заданному критерию и запись результатов в другой файл, с последующим контролем корректности выполнения операций через вывод содержимого обоих файлов на экран.

**Задание:** Задан файл F, состоящий из записей. Каждая запись содержит название книги, автора и год издания. Переписать в файл G все записи, содержащие сведения о книгах, изданных в заданном году. Вывести на экран содержимое обоих файлов.

Схема алгоритма программы:

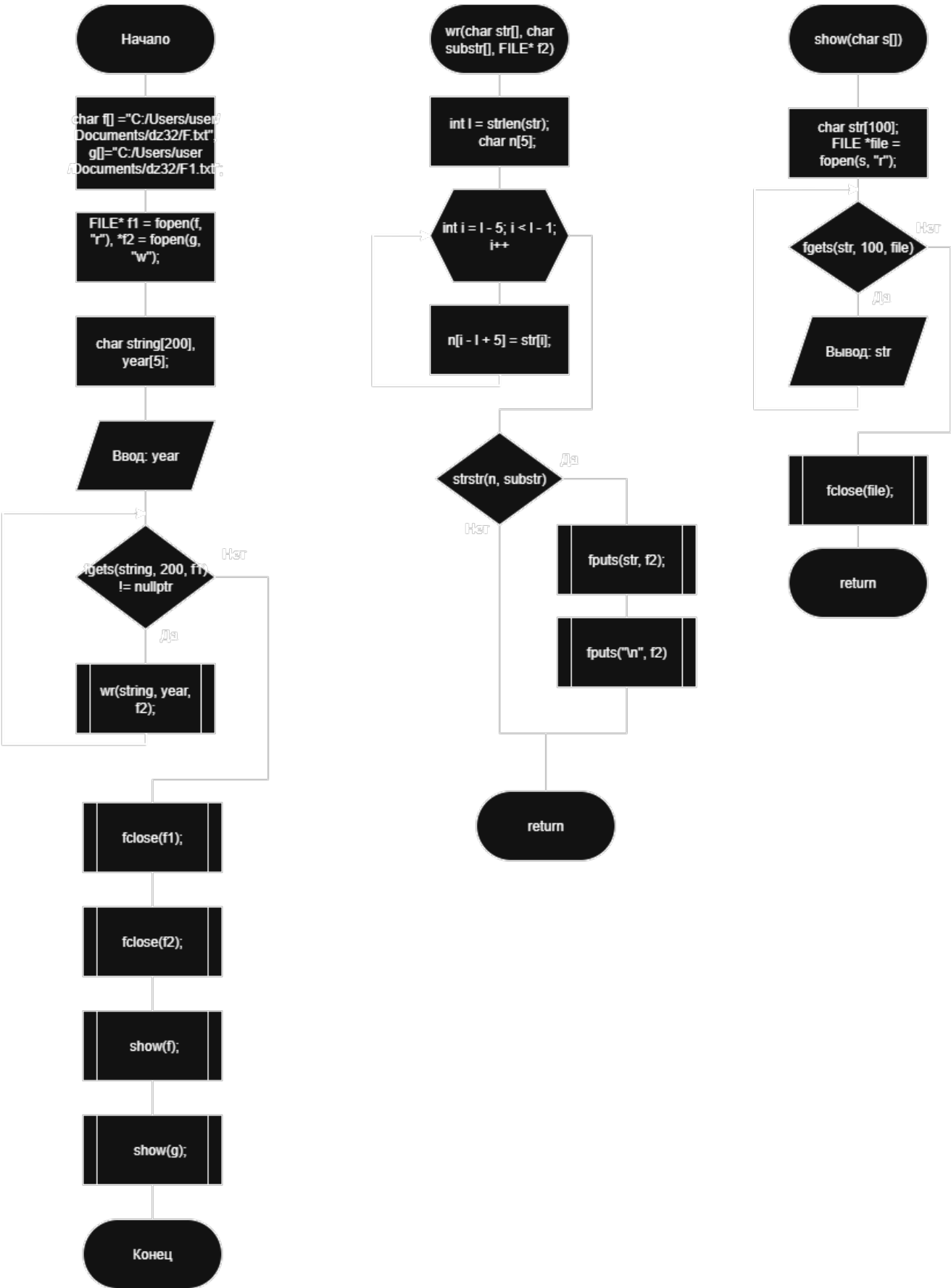


Рисунок 7 – Схема алгоритма программы

## Код программы:

```
void wr(char str[], char substr[], FILE* f2){
    int l = strlen(str);
    char n[5];
    for (int i = l - 5; i < l - 1; i++){
        n[i - l + 5] = str[i];
    }
    if (strstr(n, substr)){
        fputs(str, f2);
        fputs("\n", f2);
    }
}

void show(char s[]){
    char str[100];
    FILE *file = fopen(s, "r");
    while (fgets(str, 100, file)){
        printf("%s", str);
    }
    fclose(file);
    puts("-----");
}

int main()
{
    char f[] = "C:/Users/user/Documents/dz32/F.txt", g[] = "C:/Users/user/Documents/dz32/F1.txt";
    FILE* f1 = fopen(f, "r"), *f2 = fopen(g, "w");
    char string[200], year[5];
    fgets(year, 5, stdin);

    while (fgets(string, 200, f1) != nullptr)
        wr(string, year, f2);
    fclose(f1);
    fclose(f2);

    show(f);
    show(g);
    return 0;
}
```

Рисунок 8 – Код программы



## Тестирование.

Таблица 1 – Тестирование программы

| Входные данные | Выходные данные  |
|----------------|--|
| 2017           | <p>"1984" Джордж Оруэлл 1949<br/> "Мастер и Маргарита" Михаил Булгаков 1967<br/> "Убить пересмешника" Харпер Ли 2015<br/> "Три товарища" Эрих Мария Ремарк 1936<br/> "Сто лет одиночества" Габриэль Гарсиа Маркес 2017<br/> "Маленькие женщины" Луиза Мэй Олкотт 2019<br/> "Портрет Дориана Грея" Оскар Уайльд 2015<br/> "Алхимик" Пауло Коэльо 2017<br/> "Вино из одуванчиков" Рэй Брэдбери 2019<br/> "Над пропастью во ржи" Джером Д. Сэлинджер 2015<br/> "Дюна" Фрэнк Герберт 2019<br/> "Понедельник начинается в субботу" Аркадий и Борис Стругацкие 2017<br/> "Гордость и предубеждение" Джейн Остин 2015<br/> "451 градус по Фаренгейту" Рэй Брэдбери 2017<br/> "Улисс" Джеймс Джойс 2019<br/> "Братья Карамазовы" Фёдор Достоевский 1880<br/> "Война и мир" Лев Толстой 1869<br/> "Шерлок Холмс" Артур Конан Дойл 2015<br/> "Гарри Поттер и Философский камень" Джоан Роулинг 2017<br/> "На Западном фронте без перемен" Эрих Мария Ремарк 2019<br/> -----<br/> "Сто лет одиночества" Габриэль Гарсиа Маркес 2017<br/> <br/> "Алхимик" Пауло Коэльо 2017<br/> <br/> "Понедельник начинается в субботу" Аркадий и Борис Стругацкие 2017<br/> <br/> "451 градус по Фаренгейту" Рэй Брэдбери 2017<br/> <br/> "Гарри Поттер и Философский камень" Джоан Роулинг 2017<br/> -----</p> |
| 2015           | <p>"1984" Джордж Оруэлл 1949<br/> "Мастер и Маргарита" Михаил Булгаков 1967<br/> "Убить пересмешника" Харпер Ли 2015<br/> "Три товарища" Эрих Мария Ремарк 1936<br/> "Сто лет одиночества" Габриэль Гарсиа Маркес 2017<br/> "Маленькие женщины" Луиза Мэй Олкотт 2019<br/> "Портрет Дориана Грея" Оскар Уайльд 2015<br/> "Алхимик" Пауло Коэльо 2017<br/> "Вино из одуванчиков" Рэй Брэдбери 2019<br/> "Над пропастью во ржи" Джером Д. Сэлинджер 2015<br/> "Дюна" Фрэнк Герберт 2019<br/> "Понедельник начинается в субботу" Аркадий и Борис Стругацкие 2017<br/> "Гордость и предубеждение" Джейн Остин 2015<br/> "451 градус по Фаренгейту" Рэй Брэдбери 2017<br/> "Улисс" Джеймс Джойс 2019<br/> "Братья Карамазовы" Фёдор Достоевский 1880<br/> "Война и мир" Лев Толстой 1869<br/> "Шерлок Холмс" Артур Конан Дойл 2015<br/> "Гарри Поттер и Философский камень" Джоан Роулинг 2017<br/> "На Западном фронте без перемен" Эрих Мария Ремарк 2019<br/> -----<br/> "Убить пересмешника" Харпер Ли 2015<br/> <br/> "Портрет Дориана Грея" Оскар Уайльд 2015<br/> <br/> "Над пропастью во ржи" Джером Д. Сэлинджер 2015<br/> <br/> "Гордость и предубеждение" Джейн Остин 2015<br/> <br/> "Шерлок Холмс" Артур Конан Дойл 2015<br/> -----</p>  |

**Вывод:** В ходе лабораторной работы были освоены практические навыки работы с файлами в C++. Реализована программа для фильтрации данных по заданному критерию с последующей записью результатов в отдельный файл. Организован контроль корректности выполнения операций через визуализацию содержимого исходного и результирующего файлов.