МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №10.1

по дисциплине: Основы программирования тема: «Обработка текстовых файлов»

Выполнил: ст. группы ПВ-201 Машуров Дмитрий Русланович

Проверил:

Притчин Иван Сергеевич Брусенцева Валентина

Станиславовна

Белгород 2021 г.

**Лабораторная работа № 10.1**

**Обработка текстовых файлов**

**Цель работы:** получение навыков работы с потоками.

**Задания для подготовки к работе**

1. Изучить организацию работы с текстовыми и бинарными потоками.
2. Разработать алгоритм и составить программы для решения каждой из двух задач соответствующего варианта. В бинарных файлах информация хранится в машинном представлении.
3. Для бинарных файлов составить программу для создания файла.
4. Подобрать тестовые данные.

**Задание варианта №16**

Из данного текстового файла удалить слова, длина которых превышает данное число n.

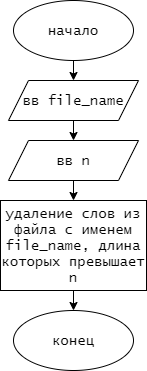
**Выполнение:**

1. **Описание алгоритма и выделение подзадач**

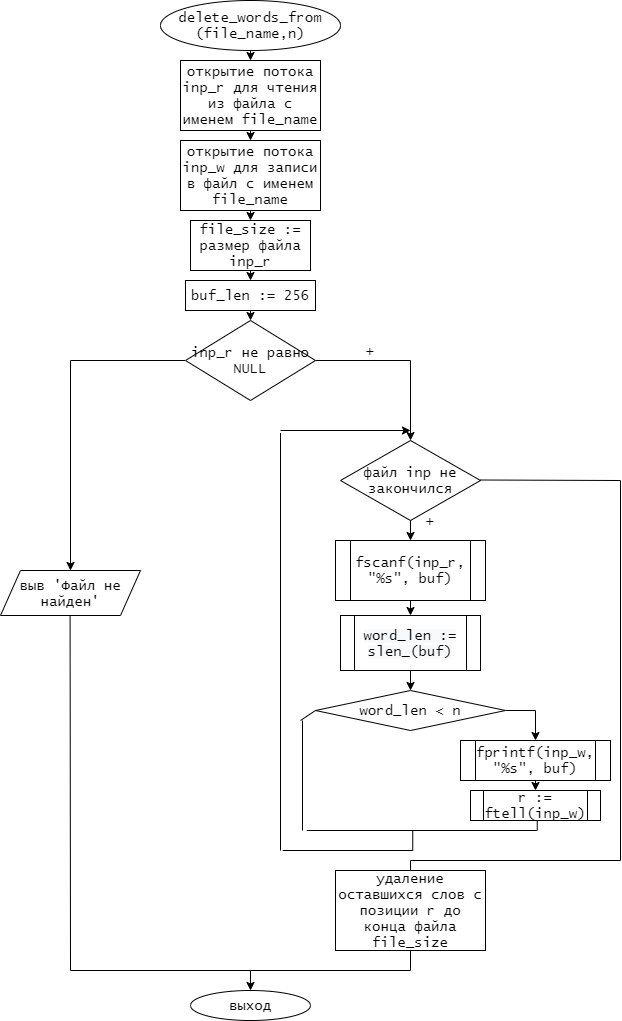
Исходя из условия задачи, будем считывать каждое слово в буфер из входного файла, сравнивать длину считанного слова с данным числом и записывать считанное слово в выходной файл, если его длина меньше, чем , иначе – переходим к следующему слову

Выделение подзадач:

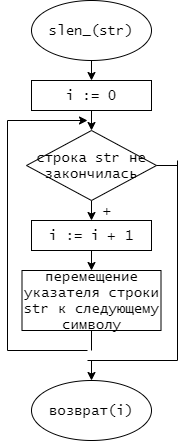
1. Удаление слов из файла, длина которых превышает заданное число
2. Определение длины строки
3. **Блок-схема с укрупнёнными блоками**



1. **Спецификации подзадач**
2. Удаление слов из файла, длина которых превышает заданное число
   1. Заголовок: void delete\_words\_from(char \*file\_name, int n)
   2. Назначение: удаляет слова из файла с именем file\_name, длина которых превышает (или равна) n
   3. Блок-схема:



1. Определение длины строки
   1. Заголовок: int slen\_(char \*str)
   2. Назначение: возвращает длину строки str
   3. Блок-схема:



1. **Тестовые данные**

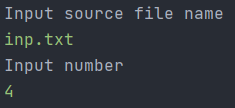
|  |  |  |
| --- | --- | --- |
| № | Вход | Выход |
| 1 | “inp.txt”    4 |  |
| 2 | “inp.txt”    5 |  |

1. **Текст программы**

#include <stdio.h>  
  
/\* возвращает длину строки str \*/  
size\_t slen\_(char \*str) {  
 size\_t i = 0;  
  
 while (\*str != '\0') {  
 i++; str++;  
 }  
  
 return i;  
}  
  
/\*  
 \* удаляет слова из файла с именем file\_name, длина  
 \* которых превышает (или равна) n  
 \*/  
void delete\_words\_from(char \*file\_name, size\_t n) {  
 FILE \*inp\_r = fopen(file\_name, "r"),  
 \*inp\_w = fopen(file\_name, "r+");  
 size\_t buf\_len = 256;  
 long r;  
  
 fseek(inp\_r, 0, SEEK\_END);  
 int file\_size = ftell(inp\_r);  
 fseek(inp\_r, 0, SEEK\_SET);  
  
 if (inp\_r != NULL) {  
 while (feof(inp\_r) == 0) {  
  
 char buf[buf\_len];  
 fscanf(inp\_r, "%s", buf);  
  
 long word\_len = slen\_(buf);  
  
 if (word\_len < n) {  
 fprintf(inp\_w, "%s ", buf);  
 r = ftell(inp\_w);  
 }  
 }  
  
  
 while (r < file\_size) {  
 fputc(' ', inp\_w);  
 r++;  
 }  
 } else {  
 perror("A file with this name wasn't found");  
 }  
  
 fclose(inp\_r);  
 fclose(inp\_w);  
}  
  
int main() {  
 printf("Input source file name\n");  
 char file\_name[50];  
 scanf("%s", file\_name);  
  
 printf("Input number\n");  
 size\_t n;  
 scanf("%u", &n);  
  
 delete\_words\_from(file\_name, n);  
  
}

1. **Результаты работы:**

*Пример №1:*



*Пример №2:*

