**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ**(**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**)» (**МАИ)**

Институт №3   
«Системы управления, информатика и  
электроэнергетика»

Кафедра 304   
Вычислительные машины, системы и сети

**ПРОГРАММИРОВАНИЕ   
НА ЯЗЫКАХ   
ВЫСОКОГО УРОВНЯ**

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №10

Бригада №4

Выполнил:

Студент группы М30-211Б-19:

Брюхов И.А.

Проверил:

Дмитриева Е.А

Москва 2020

**Оглавление**

**Задание** **3**

**Схема алгоритма** **4-7**

**Текст программы** **8-10**

**Тесты** **11-13**

**Вывод** **14**

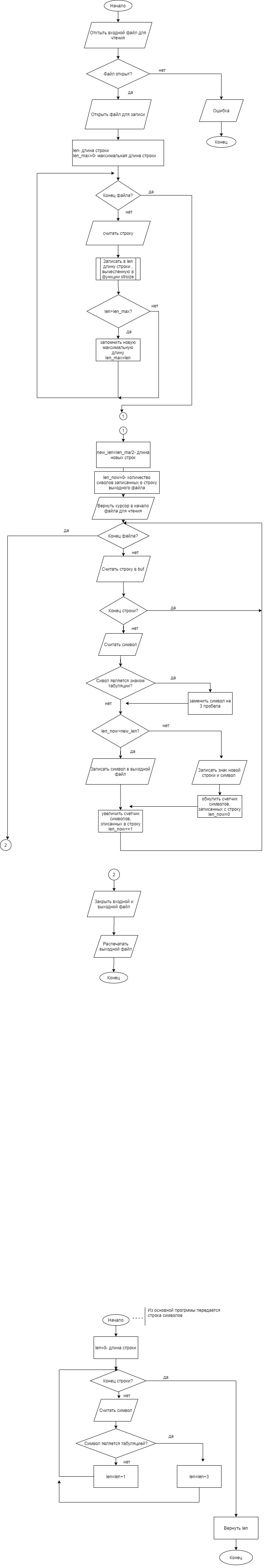
Задание:

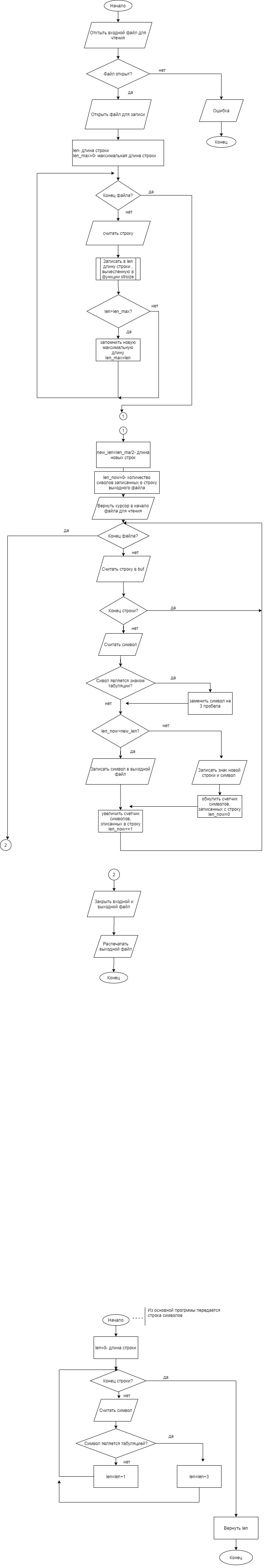
Вариант №4

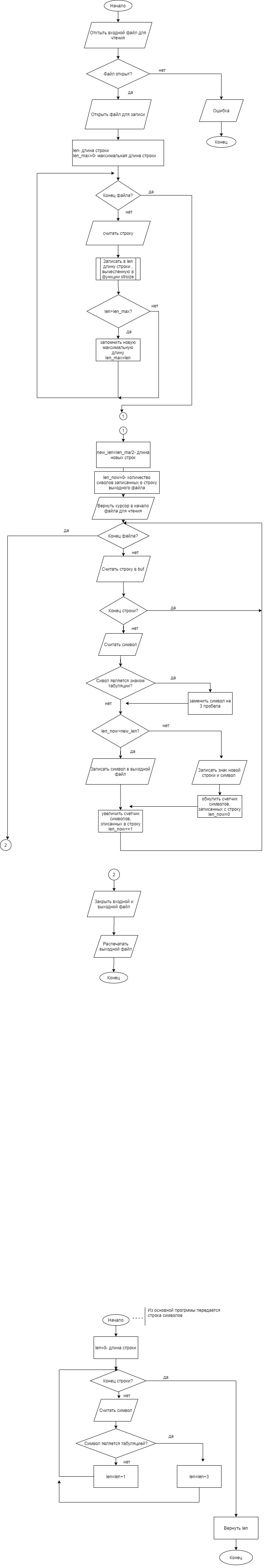
Разработать программу и подпрограмму (подпрограммы), формирующую строки текста одинаковой длины (в два раза меньше максимальной в исходном тексте) путем разбиения или объединения строк. Каждый символ табуляции во входном файле заменяется на 3 пробела. Сформатированный текст записать в выходной файл.

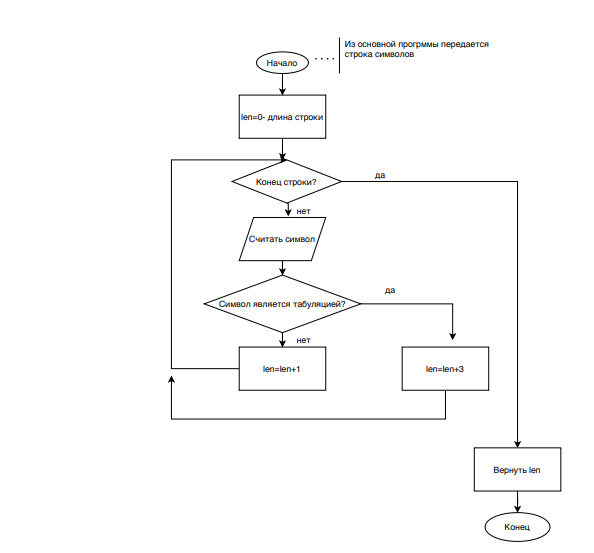
Схема алгоритма:

Основная программа:







Подпрограмма: 

Текст программы:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*Project name: Laba9 \*

\*Project Type: Win32 Console Application \*

\*File Name:Laba9.ccp \*

\*Language: C++ \*

\*Programmer(1): Брюхов Илья; М3О-211Б-19; Вариант №4\*

\*Modified by:05.10.2020 \*

\*Created:16.09.2020 \*

\*Lastrevision: 05.10.2020 \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Подключаемые Библиотеки \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

#include <string.h>

#include <locale.h>

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Глобальные константы \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

const char\* fname = "test.txt";

const char\* result="result.txt";

const int BUF\_SIZE = 256;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Определение функций \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

int strsize(char\* str);

void printFile(const char\* fname);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Основная Программа \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

int main()

{

setlocale(LC\_ALL, "rus");

FILE\* in;

FILE\* out;

if ((in = fopen(fname, "r")) == NULL) {

printf("Ошибка открытия файла \"%s\"",fname);

return 1;

}

out = fopen(result, "w");

char buf[BUF\_SIZE];

int len\_max = 0;

int len;

long int cur;

while (fgets(buf,BUF\_SIZE,in)!=NULL) {

len = strsize(buf);

if (len > len\_max) {

len\_max = len;

}

}

if (len\_max == 0) {

printf("Исходный файл \"%s\" пуст\n",fname);

return 0;

}

fseek(in, 0, SEEK\_SET);

char str[BUF\_SIZE/2];

int new\_len = len\_max / 2;

int str\_len = 0;

char new\_str[BUF\_SIZE / 2];

char c;

int len\_now = 0;

while ((fgets(str,BUF\_SIZE/2,in))!=NULL) {

char\* ptr;

int count;

if ((ptr = strrchr(str, '\n')) != NULL) {

\*ptr = ' ';

}

for (int i = 0; str[i]; i++) {

c = str[i];

if (c != '\t') {

count = 1;

}

else {

c = ' ';

count = 3;

}

for (int j = 0; j < count; j++) {

if (len\_now < new\_len) {

fputc(c, out);

len\_now++;

}

else {

fputc('\n', out);

fputc(c, out);

len\_now = 1;

}

}

}

}

fclose(in);

fclose(out);

printFile(result);

return 0;

}/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Реализация функций \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

int strsize(char\* str) {

int len = 0;

char\* ptr;

for (int i = 0; str[i]; i++) {

if (str[i] == '\t') {

len += 3;

}

else

len++;

}

if ((ptr = strrchr(str, '\n')) != NULL) {

len--;

}

return len;

}

void printFile(const char\* file) {

FILE\* in;

if ((in = fopen(file, "r")) == NULL) {

printf("Ошибка открытия файла \"%s\"", file);

return;

}

char buf[BUFSIZ];

printf("Содержимое файла \"%s\":\n", file);

for (int i = 0; i < 20; i++) {

printf("~");

}

printf("\n");

while (fgets(buf, BUF\_SIZE, in) != NULL) {

char\* ptr;

if ((ptr=strrchr(buf, '\n')) != NULL) {

\*ptr = '\0';

}

puts(buf);

}

for (int i = 0; i < 20; i++) {

printf("~");

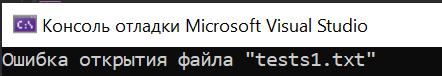
}

printf("\n");

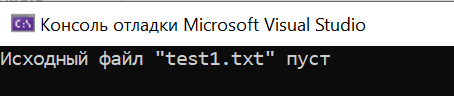
}

Тесты:

1. Некорректное имя файла
   1. Цель: Проверить работоспособность в случае когда не удастся открыть файл fname = "not\_exist.txt"
   2. Исходные данные:
   3. Ожидаемый результат: "Ошибка открытия файла not\_exist.txt "
   4. Полученный результат:



1. Файл пуст
   1. Цель: Проверить работоспособность в случае когда файл пустой
   2. Исходные данные: Пустой файл “test.txt”
   3. Ожидаемый результат: "Исходный файл "test1.txt" пуст"
   4. Полученный результат:



1. Файл содержит табуляцию
   1. Цель: Проверить работоспособность в случае когда файл содержит символы табуляции

Исходные данные:

Hellow(\t)Word

Realy(\t)nice(\t)day

* 1. Ожидаемый результат:

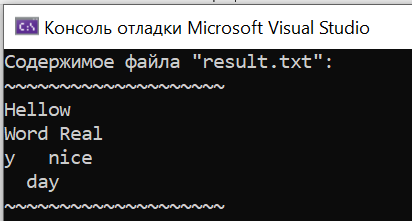
Hellow\_\_\_

Word\_Real

y\_\_nice\_\_

\_day

* 1. Полученный результат:



1. Корректный файл с русскими словами
   1. Цель: Проверить работоспособность в случае когда файл содержит русские буквы
   2. Исходные данные:

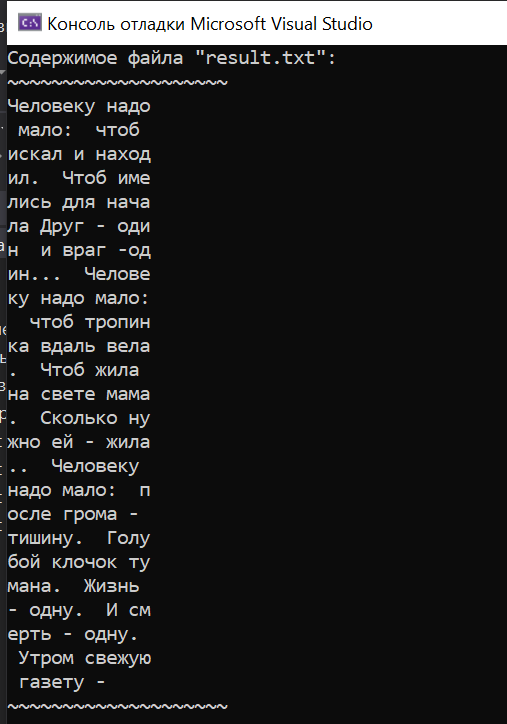
Файл со стихом:

Человеку надо мало:   
чтоб искал и находил.   
Чтоб имелись для начала  
Друг — один   
и враг —один...   
Человеку надо мало:   
чтоб тропинка вдаль вела.   
Чтоб жила на свете мама.   
Сколько нужно ей — жила..   
Человеку надо мало:   
после грома — тишину.   
Голубой клочок тумана.   
Жизнь — одну.   
И смерть — одну.   
Утром свежую газету —

* 1. Ожидаемый результат:

Все строки будут одинаковой длины

* 1. Полученный результат:



Вывод:

Разработка программы завершена на том основании, что

1. Полученный результат совпал с ожидаемым

2. Считаем набор тестов полным