МИНЕСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доц., канд. техн. наук |  |  |  | В. А. Галанина |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ |
| Обработка строковой информации в С/С++. |
| по курсу: ИНФОРМАТИКА |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | М023 |  |  |  | Д.А.Трегуб |
|  |  |  | подпись, дата |  | инициалы, фамилия |

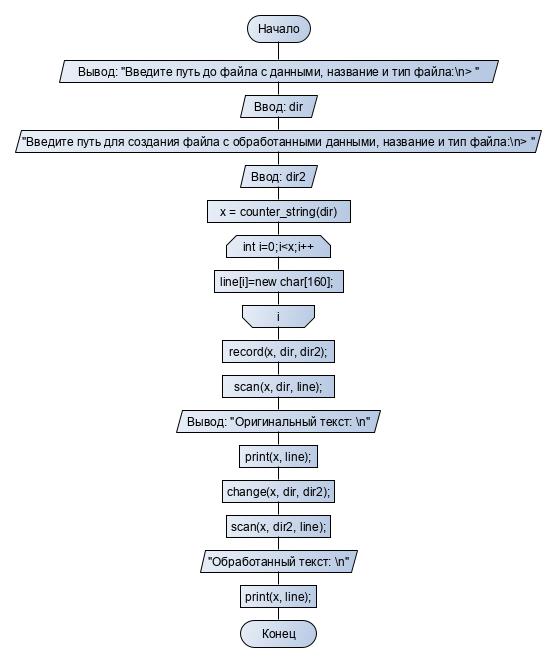
Санкт-Петербург 2020

1. Описание задания:

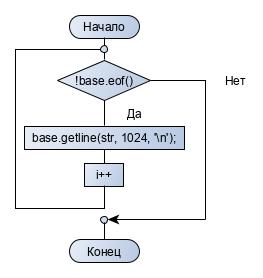
19.

Дан текст. Словом текста считается любая последовательность букв алфавита; между соседними словами - не менее одного пробела. Перед первым и за последним словом каждой строки произвольное число пробелов. Найти и сохранить в строке только те слова, в которых гласные буквы чередуются с согласными.

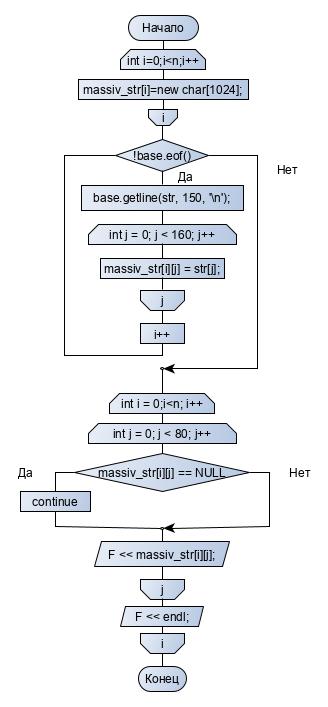
1. Алгоритм:
   1. Основной алгоритм:



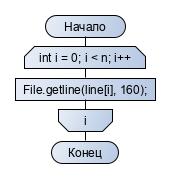
* 1. counter\_string:



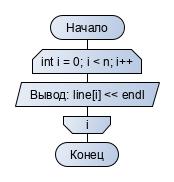
* 1. record:

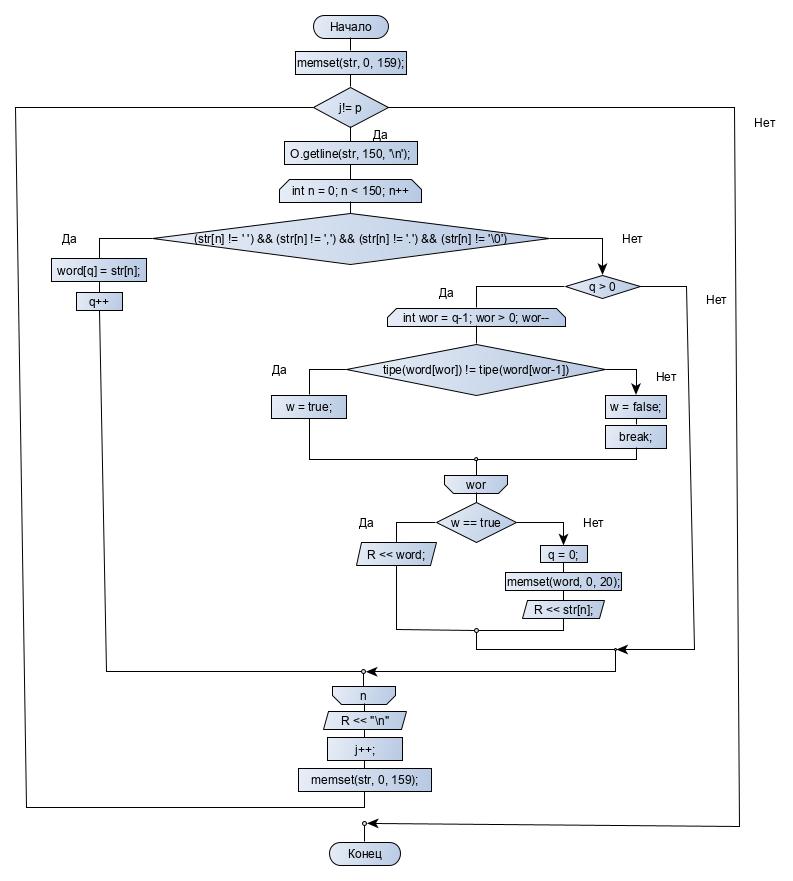
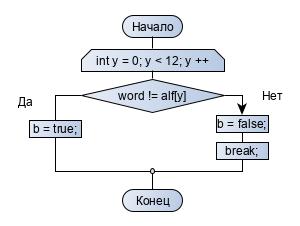


* 1. scan:



* 1. print:



* 1. change:  
     
  2. tipe:  
      

1. Программа:

#include <string.h>

#include <iostream>

#include <fstream>

#include <string>

#include <cstring>

#include <stddef.h>

using namespace std;

int counter\_string(char dir[]);

bool tipe(const char& word);

void record(int n, char dir[], char dir2[]);

void scan(int n, char dir[], char \*\*line);

void print(int n, char \*\* line);

void change(int p, char dir1[], char dir2[]);

int main()

{

system("chcp 1251");

system("cls");

char dir[80];

char dir2[80];

int x, a, k = 0, i = 0;

cout << "Введите путь до файла с данными, название и тип файла:\n> ";

cin.getline(dir, 100);

cout << "Введите путь для создания файла с обработанными данными, название и тип файла:\n> ";

cin.getline(dir2, 100);

x = counter\_string(dir);

char \*\*line;

line=new char\*[x];

for (int i=0;i<x;i++)

line[i]=new char[160];

record(x, dir, dir2);

scan(x, dir, line);

cout << "Оригинальный текст:" << endl;

print(x, line);

change(x, dir, dir2);

scan(x, dir2, line);

cout << "Обработанный текст: " << endl;

print(x, line);

return 0;

}

int counter\_string(char dir[])

{

char \*str = new char [1024];

int i=0;

ifstream base(dir);

while (!base.eof())

{

base.getline(str, 1024, '\n');

i++;

}

base.close();

delete[] str;

return i;

}

void record(int n, char dir[], char dir2[])

{

int i = 0;

char \*str = new char [160];

char \*\*massiv\_str;

massiv\_str=new char\*[n]; //создания массива строк для последующей записи в другой файл

for (int i=0;i<n;i++)

massiv\_str[i]=new char[1024];

ifstream base(dir);

while (!base.eof())

{

base.getline(str, 150, '\n');

for(int j = 0; j < 160; j++){

massiv\_str[i][j] = str[j];

}

i++;

}

base.close();

ofstream F(dir2);

for(int i = 0;i<n; i++)

{

for(int j = 0;j<80; j++){

if(massiv\_str[i][j] == NULL)

continue;

F << massiv\_str[i][j];

}

F << endl;

}

F.close();

delete[] str;

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

delete [] massiv\_str[i];

}

delete [] massiv\_str;

}

void scan(int n, char dir[], char \*\*line)

{

int a = 0;

ifstream File(dir);

for(int i = 0; i < n; i++)

File.getline(line[i], 160);

File.close();

}

void print(int n, char \*\* line)

{

for(int i = 0; i < n; i++)

{

cout << line[i] << endl;

}

}

void change(int p, char dir1[], char dir2[])

{

char \*str = new char [160];

char \*word = new char[20];

int q = 0, z = 0, j = 0;

bool w = true;

ifstream O(dir1);

ofstream R(dir2);

memset(str, 0, 159);

try {

while (j!= p)

{

O.getline(str, 150, '\n');

for(int n = 0; n < 150; n++)

{

if ((str[n] != ' ') && (str[n] != ',') && (str[n] != '.') && (str[n] != '\0'))

{

word[q] = str[n];

q++;

}

else

{//проверка на q > 0

if (q > 0)

{

for(int wor = q-1; wor > 0; wor--)

{

if (tipe(word[wor]) != tipe(word[wor-1]))

{

w = true;

}

else

{

w = false;

break;

}

}

if (w == true)

{

R << word;

}

else

{

for(int wor = q-1; wor > 0; wor--)

{

R << ' ';

}

}

q = 0;

memset(word, 0, 20);

R << str[n];

}

}

}

R << "\n";

j++;

memset(str, 0, 159);

}

delete[] str;

delete[] word;

R.close();

O.close();

}

catch (const char\* os) {

cerr << os;

}

}

bool tipe(const char& word)

{

bool b = true;

char alf[12] = {'a','A','e','E','y','Y','u','U','i','I','o','O'};

for(int y = 0; y < 12; y ++)

{

if( word != alf[y])

{

b = true;

}

else

{

b = false;

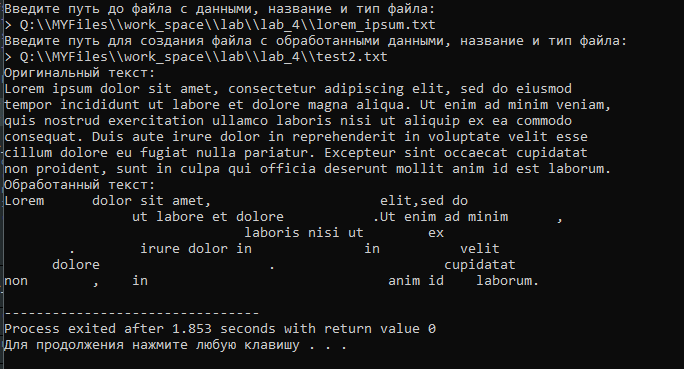
break;

}

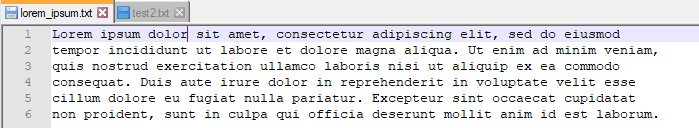
}

return b;

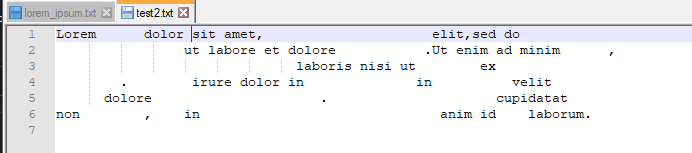
}

1. Результаты работы программы:   
   

Файл с оригинальным текстом:



Файл с обработанным текстом:



1. Вывод:

Выполняя данную лабораторную работу, я научился обрабатывать строковую информацию с помощью C/C++.