*Федеральное государственное бюджетное образовательное учреждение* *высшего образования*

|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | ***«Московский государственный технический университет  имени Н.Э. Баумана»***  ***(национальный исследовательский университет)*** |

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА Компьютерные Системы и сети (ИУ6)

**Отчет**

**по лабораторной работе №6**

**варианта №7**

**Название лабораторной работы:**

Строки

**Дисциплина:**

Основы программирования

Студент гр. **ИУ6-12** **25/11/16\_\_\_\_\_\_\_\_\_\_ Векшин Роман**

(Подпись, дата) (И.О. Фамилия)

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Черноусова Татьяна Геннадьевна**

(Подпись, дата) (И.О. Фамилия)

**Задание:**

Дана последовательность пяти символьных слов латинского алфавита длиною не более 40 символов. Слова разделены пробелами. Последнее слово заканчивается символом «.». Если слово начинается с букв А,В,С, то третью букву слова заменить на Y, а если слова оканчивается на D,E,F, то предпоследнюю букву заменить на О.

**Схема алгоритма:**

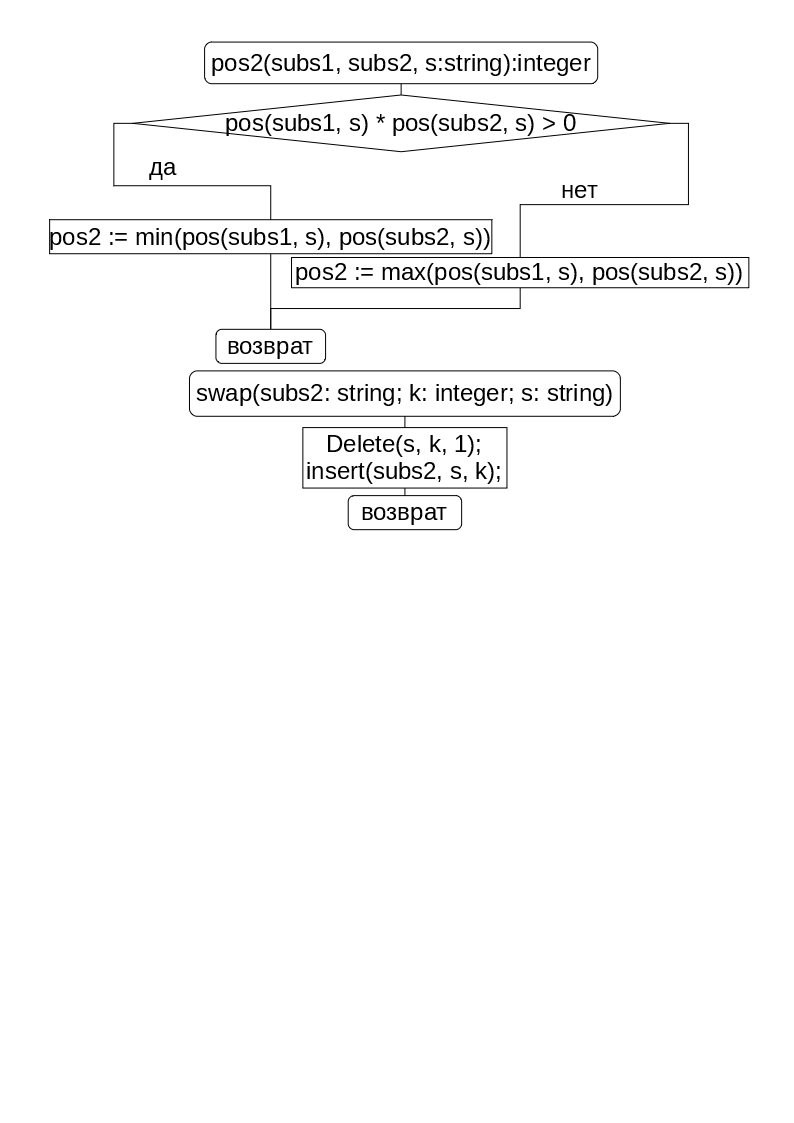
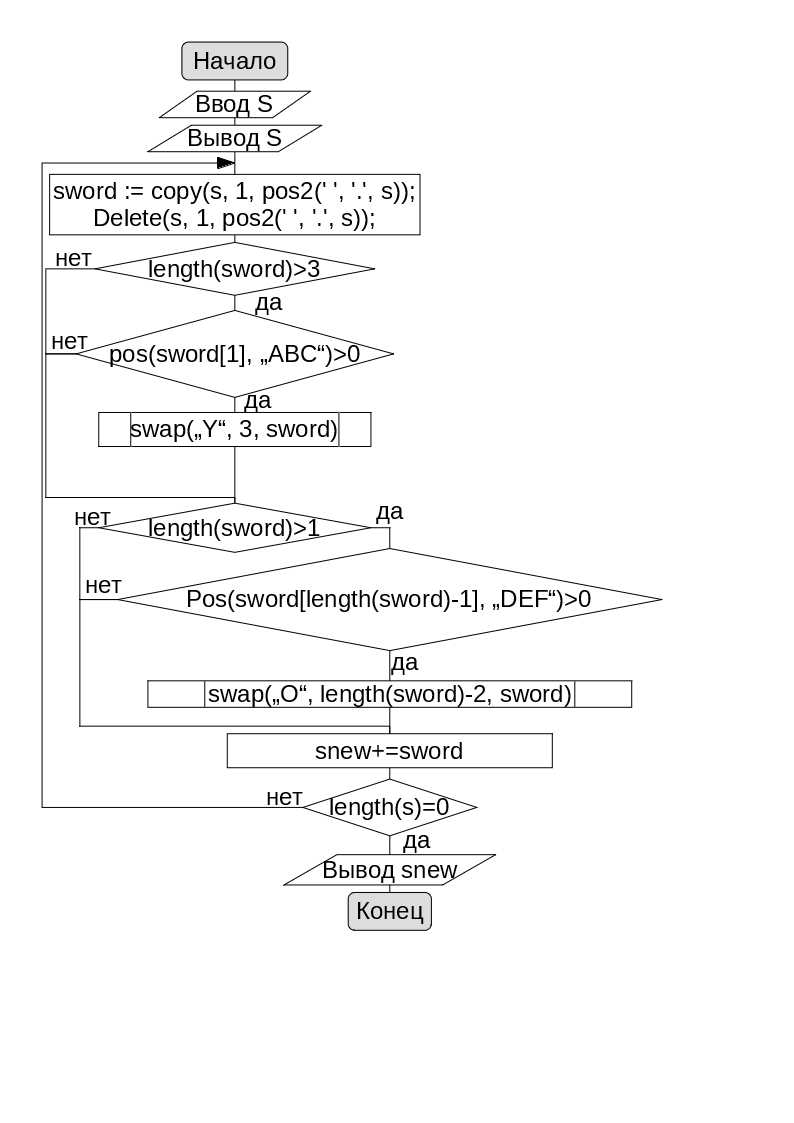


Рисунок -Схема алгоритма

**Код программы:**

program project1;

{$APPTYPE CONSOLE}

Uses Math;

function pos2(const subs1, subs2, s: string): integer;

begin

if (pos(subs1, s) \* pos(subs2, s) > 0) then

pos2 := min(pos(subs1, s), pos(subs2, s))

else

pos2 := max(pos(subs1, s), pos(subs2, s));

end;

procedure swap(subs2: string; k: integer; var s: string);

begin

Delete(s, k, 1);

insert(subs2, s, k);

end;

var

s, sword, snew: string;

begin

writeln(utf8toansi('Введите строку'));

readln(s);

writeln(utf8toansi('Введена строка'));

writeln(s);

repeat

sword := copy(s, 1, pos2(' ', '.', s));

Delete(s, 1, pos2(' ', '.', s));

if length(sword) > 3 then

if pos(copy(sword, 1, 1), 'ABC') > 0 then

swap('Y', 3, sword);

if length(sword) > 1 then

if pos(copy(sword, length(sword) - 1, 1), 'DEF') > 0 then

swap('O', length(sword) - 2, sword);

snew += sword;

until length(s) = 0;

writeln(utf8toansi('Строка-результат'));

writeln(snew);

readln();

end.

**Тестирование:**

Таблица -Результаты тестирования программы 1

|  |  |  |
| --- | --- | --- |
| Исходные данные | Ожидаемый результат | Полученный результат |
| ahrhj isioj y ojk a r. | ahrhj isioj y ojk a r. | ahrhj isioj y ojk a r. |
| Airiiri Birrr Cijrgij A jgfjD rejgF. | AiYiiri BiYrr CiYrgij A jgfOD rejOF. | AiYiiri BiYrr CiYrgij A jgfOD rejOF. |
| AAAAAA BBBBB CCCCC DDDDDD EEEEE. | AAYAAA BBYBB CCYCC DDDDOD EEEOE. | AAYAAA BBYBB CCYCC DDDDOD EEEOE. |
| A B C D E F Z. | A B C D E F Z. | A B C D E F Z. |

**Пример работы программы:**

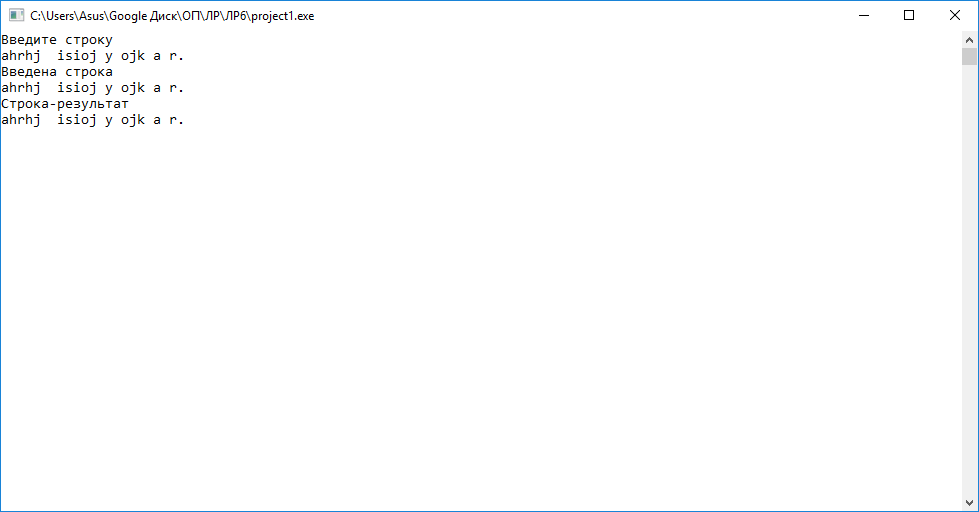


Рисунок -Пример работы программы

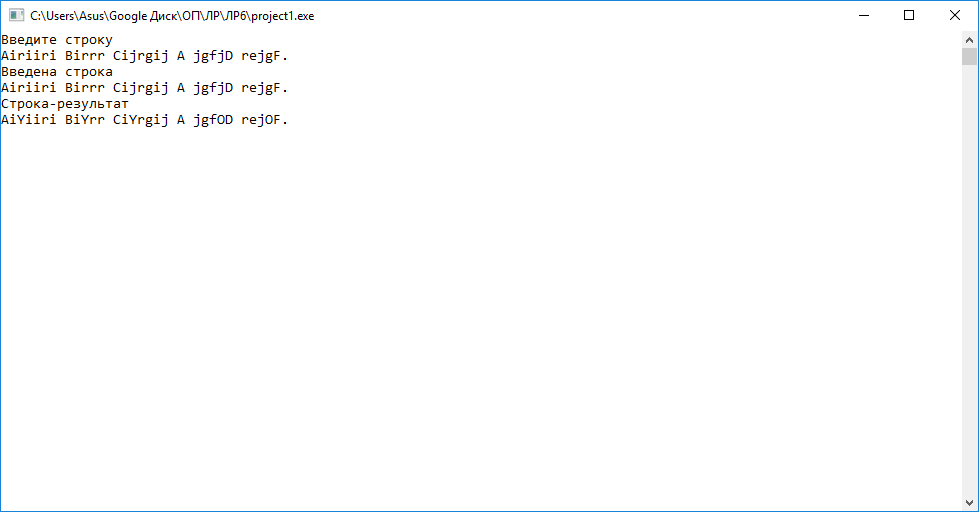


Рисунок -Пример работы программы

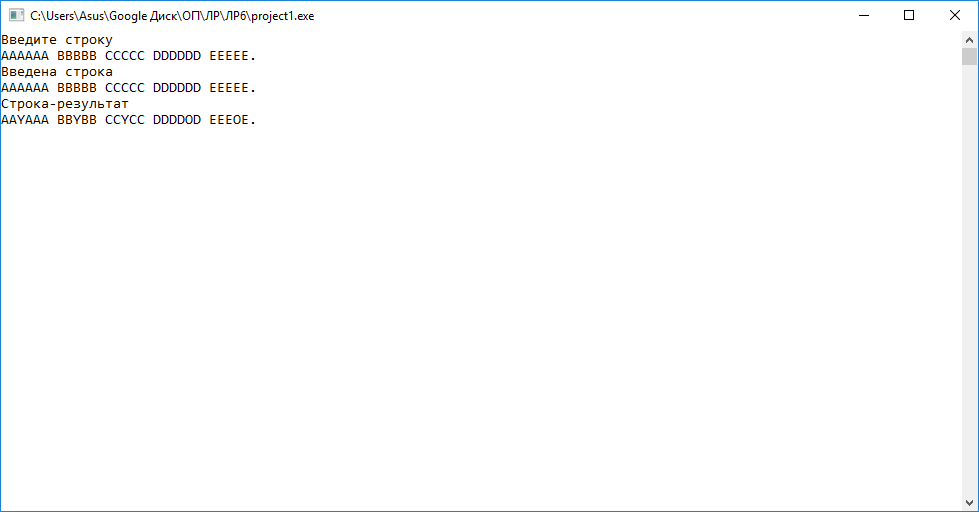


Рисунок -Пример работы программы

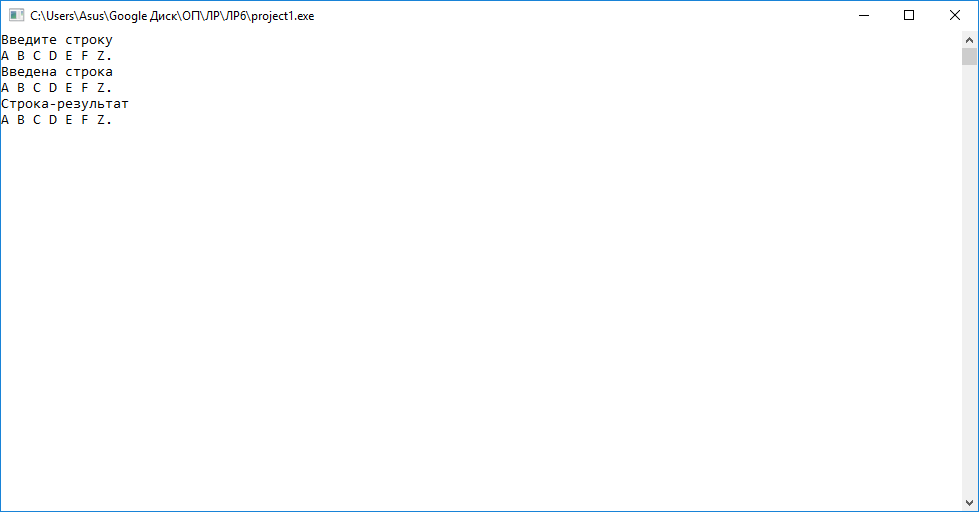


Рисунок -Пример работы программы

**Вывод:**

1. Разработан алгоритм, выделяющий из строки слова и редактирующий их согласно условию, и составлена его схема в среде LibreOffice Draw(см. рис. 1).
2. Создан код программы по схеме алгоритма в среде Lazarus
3. Проведено тестирование (см. табл. 1).
4. Тестирование показало корректность работы программы (см. рис. 2-5)