Выполнила

студентка группы КТбо1-2 Н. А. Свидич

Принял

ассистент ИКТИБ Д. С. Кочубей

Таганрог 2023

Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное

учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

(ФГАОУ ВО «ЮФУ»)

Институт компьютерных технологий и информационной безопасности

Кафедра системного анализа и телекоммуникаций

**ЛАБОРАТОРНАЯ РАБОТА № 3**

по дисциплине «Основы алгоритмизации и программирования»

Вариант 2

на тему:

«Строки и массивы символов»

**Практическое задание № 5 «Строки и массивы символов»**

Цель данной лабораторной работы ознакомиться с обработкой символьной информации, научиться при программировании символьной обработки использованию таких понятий, как строки и массивы символов.

**Задание**

Вариант 2.

Написать программу, которая во вводимом с клавиатуры тексте выберет слова, начинающиеся с гласной буквы, и выведет их на экран.

**Выполнение**

1. Составление блок-схемы.

В начале выполнения задания, для систематизирования плана действий составляется блок-схема алгоритма работы программного кода (Рисунок 1).

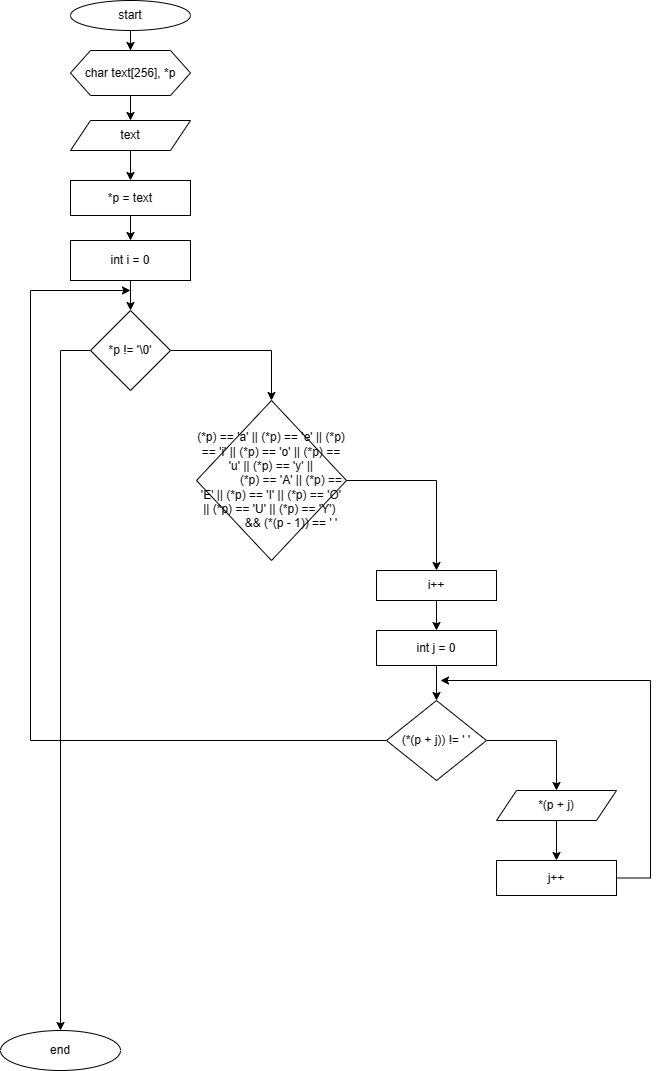


Рисунок 1 – Блок-схема поэтапного решения задачи

2. Определение и инициализация переменных.

В процессе выполнения задания задействованы: массив типа «char» и максимальным размером 256 символов – text[256], указатель на массив text - \*p.

Текст, из которого будет делаться выборка, вводится командой «gets\_s(text)» - данная команда позволяет игнорировать пробелы в тексте и вводить строку до символа перехода (Enter). Также, команда преобразует сплошной набор символов в символьный массив.

После ввода текста, указателю присваивается ссылка на полученный массив символов, чтобы иметь возможность обращаться к конкретному элементу массива. В этот момент, указатель содержит адрес этого элемента.

3. Основной этап.

После обращения к элементам, необходимо делать проверку на наличие гласной буквы в начале слова, после чего выполнить вывод подходящих под условия слов. Для того, чтобы вывести элементы символьного массива в виде порядка слов, необходимо использовать цикл.

На рисунке 2 показан результат работы полученного программного кода.

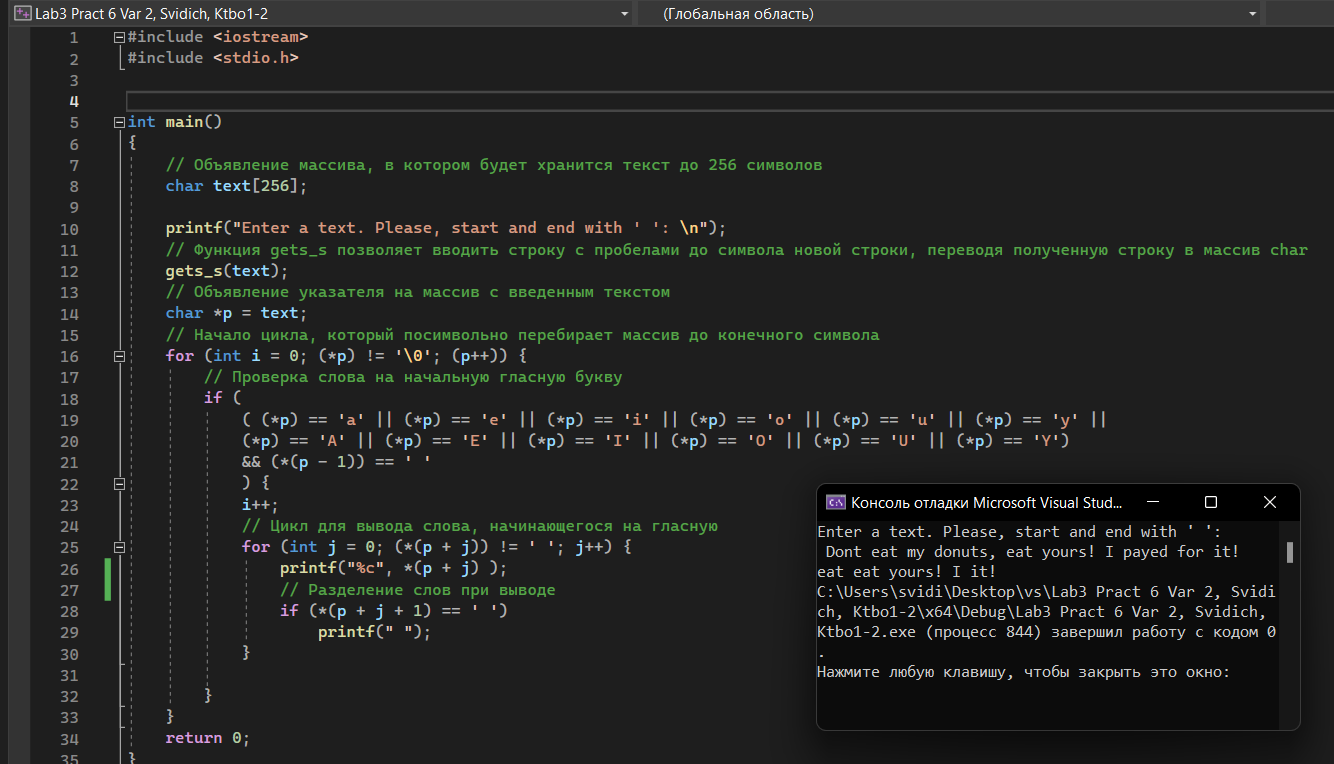


Рисунок 2 –Выполнение программного кода

**Вывод**

Ознакомились с обработкой символьной информации, научились при программировании символьной обработки использованию таких понятий, как строки и массивы символов.

Приложение А

Полученный программный код

«Lab3 Pract 6 Var 2, Svidich, Ktbo1-2.cpp»

#include <iostream>

#include <stdio.h>

int main()

{

// Объявление массива, в котором будет хранится текст до 256 символов

char text[256];

printf("Enter a text: \n");

// Функция gets\_s позволяет вводить строку с пробелами до символа новой строки, переводя полученную строку в массив char

gets\_s(text);

// Объявление указателя на массив с введенным текстом

char\* p = text;

// Начало цикла, который посимвольно перебирает массив до конечного символа

for (int i = 0; (\*p) != '\0'; (p++)) {

// Проверка слова на начальную гласную букву

if (

((\*p) == 'a' || (\*p) == 'e' || (\*p) == 'i' || (\*p) == 'o' || (\*p) == 'u' || (\*p) == 'y' ||

(\*p) == 'A' || (\*p) == 'E' || (\*p) == 'I' || (\*p) == 'O' || (\*p) == 'U' || (\*p) == 'Y')

&& (\*(p - 1)) == ' '

) {

i++;

// Цикл для вывода слова, начинающегося на гласную

for (int j = 0; (\*(p + j)) != ' '; j++) {

printf("%c", \*(p + j));

// Разделение слов при выводе

if (\*(p + j + 1) == ' ') {

printf(" ");

}

// Прекращение цикла вывода слова, если был выведен последний символ последнего слова

if (\*(p + j + 1) == '\0') {

break;

}

}

}

}

return 0;

}