Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский государственный технический университет имени Н.Э. Баумана»
(национальный исследовательский университет)
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

О Т Ч Е Т

по домашнему заданию №1

Название домашнего задания: Обработка символьной информации

Дисциплина: Машинно-зависимые языки и основы компиляции

Студент гр. ИУ6-41б _____ _____
(Подпись, дата) (И.О. Фамилия)

Преподаватель _____ _____
(Подпись, дата) (И.О. Фамилия)

Москва, 2025

СОДЕРЖАНИЕ

Задание

Дан текст, состоящий из 8 слов по 5 символов. Определить количество гласных букв в каждом слове.

Ход работы

Схема алгоритма

Построим схему алгоритма программы, решающей эту задачу. Схема алгоритма представлена на рисунке 1.

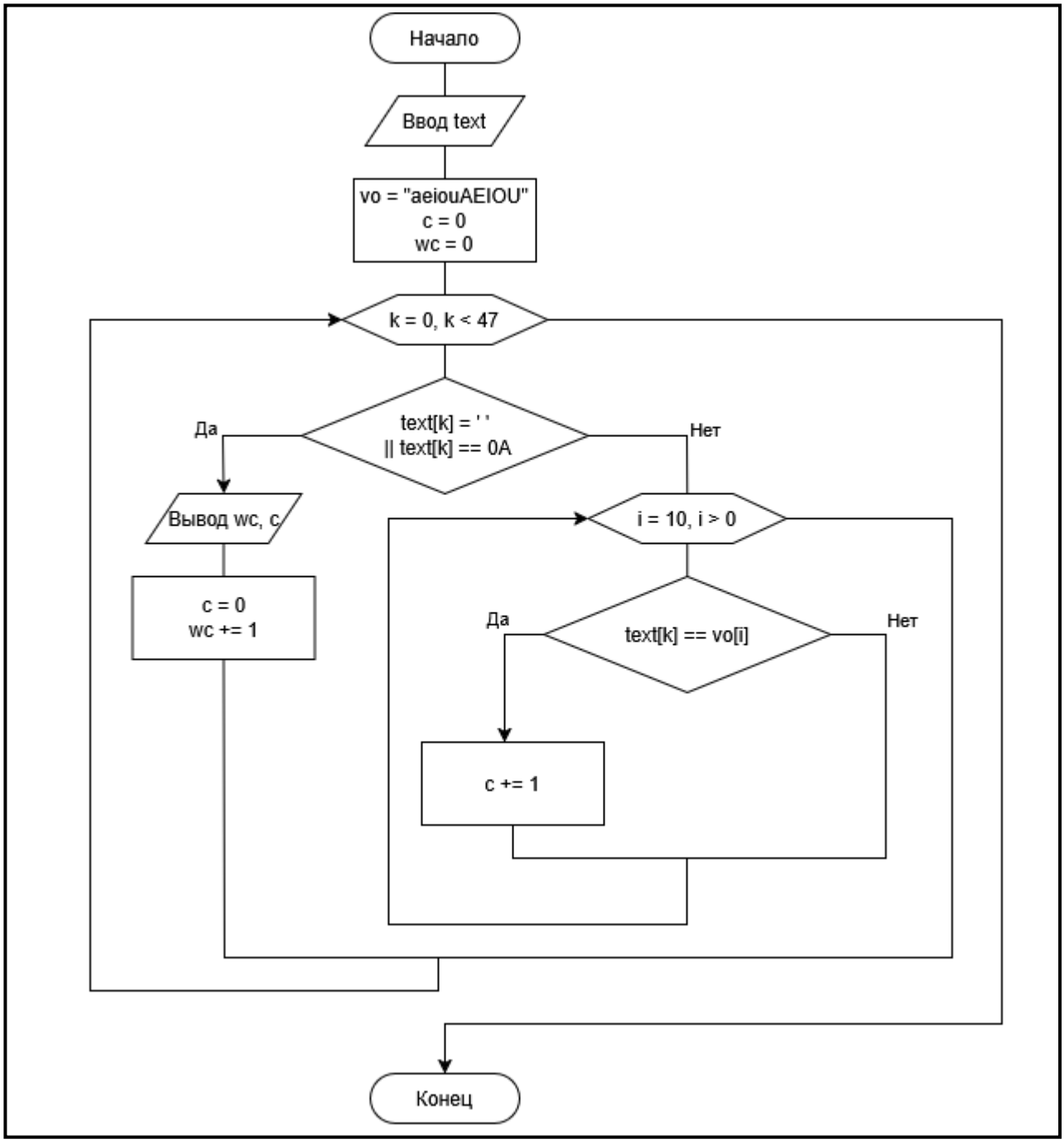

Рисунок 1 – Схема алгоритма

Реализация программы

На листинге 1 представлена программа, решающая задачу. Чтобы программа работала корректно, необходимо вводить слова по пять символов, разделяя их пробелами. Программа корректно работает с латинским алфавитом.

Листинг 1 – Программа, решающая задачу

```
section .data
    enter_msg db "Enter text (8 words 5 symbols): "
    len_enter_msg equ $-enter_msg

    word_msg db "1 word: "
    len_word_msg equ $-word_msg

    new_line_msg db 10

    ex_msg db "Example: ", 0
    len_ex_msg equ $-ex_msg

    text db "xxxxx xxxxx xxxxx xxxxx xxxxx xxxxx xxxxx xxxxx", 10
    len_text equ $-text
    vo db "aeiouAEIOU"
    c db 0
    wc db 0

section .text:
    global _start
_start:
    mov eax, 4
    mov ebx, 1
```

Продолжение листинга 1

```
        mov ecx, ex_msg
        mov edx, len_ex_msg
        int 80h
        mov eax, 4
        mov ebx, 1
        mov ecx, text
        mov edx, len_text
        int 80h

        mov eax, 4
        mov ebx, 1
        mov ecx, enter_msg
        mov edx, len_enter_msg
        int 80h

        mov eax, 3
        mov ebx, 0
        mov ecx, text
        mov edx, len_text
        int 80h

        mov ecx, 47
        mov esi, text
    text_cycle:
        push ecx
        call check_v
        cmp byte[esi], ' '
        je space_found
```

Продолжение Листинга 1

```
        cmp byte[esi], 10
        je space_found
        pop ecx
        inc esi
        loop text_cycle

    exit_program:
        mov eax, 1
        xor ebx, ebx
        int 80h

    check_v:
        mov ecx, 10
        mov ebx, esi
        mov esi, vo
    .cycle_check:
        mov al, byte[esi]
        cmp al, byte[ebx]
        je .succes
        inc esi
        loop .cycle_check
    .fail:
        mov esi, ebx
        ret
    .succes:
        mov esi, ebx
        inc byte[c]
        ret
```

Продолжение листинга 1

```
        space_found:
        inc byte[wc]
        add byte[wc], '0'
        mov al, byte[wc]
        mov [word_msg], al
        mov eax, 4
        mov ebx, 1
        mov ecx, word_msg
        mov edx, len_word_msg
        int 80h
        sub byte[wc], '0'
        add byte[c], '0'
        mov eax, 4
        mov ebx, 1
        mov ecx, c
        mov edx, 1
        int 80h
        mov eax, 4
        mov ebx, 1
        mov ecx, new_line_msg
        mov edx, 1
        int 80h
        mov byte[c], 0
        cmp byte[esi], 10
        je exit_program
        inc esi
        jmp text_cycle
```

Тестирование

Запустим программу и введем корректные данные (строку «aoieu xxxxx fhjdk aioid gfhjr dhasd dfkkl fhbsd»). Ожидается, что в первом слове будет 5 гласных, во втором – 0, в третьем – 0, в четвертом – 4, в пятом – 0, в шестом – 1, в седьмом – 0, в восьмом – 0. Ввод данных и результат работы программы представлены на рисунке 2.

Рисунок 2 – Результат работы программы.

Программа вывела верный ответ при корректно введенных данных.

Контрольные вопросы

Вопрос 1

Дайте определение символьной строки.

Символьная строка — это упорядоченный набор байтов, расположенных в памяти подряд, который может обрабатываться как единое целое и имеет определённый размер.

Вопрос 2

Назовите основные команды обработки цепочек?

Основные команды для работы с цепочками — это MOVS (копирование данных), CMPS (сравнение строк), SCAS (поиск в строке), LODS (загрузка элемента), STOS (запись элемента), а также префиксы повторения REP, REPE, REPNE.

Вопрос 3

Какие операции выполняют строковые команды MOVS? Какие особенности характерны для этих команд?

Команда MOVS перемещает данные из источника (SI/ESI/RSI) в приёмник (DI/EDI/RDI), автоматически изменяя указатели в зависимости от флага направления (DF). Размер данных задаётся суффиксом: MOVSB (байты), MOVSW (слова), MOVSD (двойные слова).

Вопрос 4

Какие операции выполняют строковые команды CMPS, SCAS? Какие особенности характерны для этих команд?

Команда CMPS сравнивает элементы двух строк, обновляя флаги процессора, а SCAS ищет в строке значение из регистра AL/AX/EAX. Обе команды изменяют индексные регистры (SI/DI) и учитывают флаг направления (DF).

Вопрос 5

Как обеспечить циклическую обработку строк?

Циклическая обработка строк достигается с помощью префиксов REP (повторять), REPE (повторять, пока равно) и REPNE (повторять, пока не равно) либо через ручное управление счётчиком (CX/ECX/RCX) и условными переходами.

Вопрос 6

Какова роль флага DF во флажковом регистре при выполнении команд обработки строк?

Флаг направления (DF) определяет, в какую сторону изменяются указатели при обработке строк: если DF = 0 (установлен CLD), указатели увеличиваются, если DF = 1 (установлен STD) — уменьшаются.

Вопрос 7

Как правильно выбрать тестовые данные для проверки алгоритма обработки строки?

Тестовые данные для проверки алгоритмов должны включать как стандартные случаи, так и крайние варианты (пустые строки, максимальная длина, специальные символы), чтобы убедиться в корректности работы программы во всех возможных сценариях.

Вывод

В ходе выполнения домашнего задания были изучены и применены навыки работы с символами средствами NASM. Написана 32-разрядная программа, подсчитывающая количество гласных букв в словах, разделенных пробелами. Программа корректно работает для латинского алфавита.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Иванова Г.С., Ничушкина Т.Н., лабораторный практикум по программирование на ассемблере в операционной системе LINUX. М.: МГТУ имени Н.Э. Баумана, 2022. 77 с.