|  |  |
| --- | --- |
| Gerb-BMSTU_01 | ***«*Министерство науки и высшего образования Российской Федерации**  **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\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОТЧЕТ**

**по домашнему заданию № 1**

**Дисциплина: Машинно-зависимые языки и основы компиляции**

**Название лабораторной работы: Обработка символьной информации**

Студент гр. ИУ6-41Б  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С. А. Самодурова**

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С. С. Данилюк**

(Подпись, дата) (И.О. Фамилия)

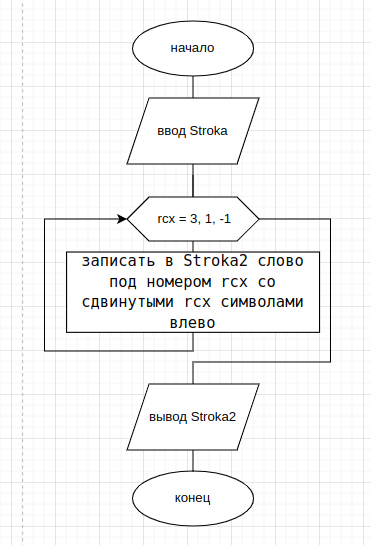
Москва, 2023

**Цель:** изучение команд обработки цепочек и приемов обработки символьной информации.

**Текст задания:**

Дан текст 18 символов: 3 слова по 6 символов. Осуществить кольцевой сдвиг каждого слова: 1-го на 1 символ влево, 2-го на 2 символа, 3-го на 3 символа.

Схема алгоритма программы представлена на рисунке 1.

Рисунок 1 — схема алгоритма

**Код программы:**

%include "io64.asm"

section .data

InpMsg db "Input 3 words: ",10

lenInp equ $-InpMsg

Stroka db "abcdef qwerty abcdef",10

lenStroka equ $-Stroka

Stroka2 db "rrrrrr rrrrrr rrrrrr",10

lenStroka2 equ $-Stroka2

Dlina dw 3

section .bss

InBuf resb 10

lenIn equ $-InBuf

StrokaInp resw 20

lenStr equ $-StrokaInp

section .text

global \_start

\_start:

mov rax, 1 ; системная функция 1 (write)

mov rdi, 1 ; дескриптор файла stdout=1

mov rsi, InpMsg ; адрес выводимой строки

mov rdx, lenInp ; длина строки

syscall ; вызов системной функции

; read

mov rax, 0 ; системная функция 0 (read)

mov rdi, 0 ; дескриптор файла stdin=0

mov rsi, StrokaInp ; адрес вводимой строки

mov rdx, lenStr ; длина строки

syscall ; вызов системной функции

lea rdi,[StrokaInp] ; загружаем адрес строки в edi

mov dx,0 ; обнуляем счетчик длины слова

;алгоритм замены во втором слове ecx=2

mov ecx,3

; mov edx,14

cycl: push rcx

mov eax,7

imul ecx

mov edx,eax

sub edx,7

mov ebx,6

sub ebx,ecx ;4 = ebx

lea esi, [StrokaInp+edx]

lea edi, [Stroka2+edx+ebx]

repe movsb

pop rcx

push rcx

lea esi, [StrokaInp+edx+6]

sub esi,ebx

lea edi, [Stroka2+edx]

mov ecx, ebx; ecx=4

repe movsb

pop rcx

loop cycl

mov rax, 1 ; системная функция 1 (write)

mov rdi, 1 ; дескриптор файла stdout=1

mov rsi, Stroka2 ; адрес выводимой строки

mov rdx, lenStroka2 ; длина строки

syscall ; вызов системной функции

exit:

xor rdi, rdi

mov rax, 60

syscall

**Результат работы программы:**

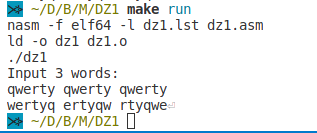
****

Рисунок 2 — результат выполнения

Таблица 1 — проведенные тесты

|  |  |  |  |
| --- | --- | --- | --- |
| № | Входные данные | Ожидаемый результат | Полученный результат |
| 1 | qwerty qwerty qwerty | wertyq ertyqw rtyqwe | wertyq ertyqw rtyqwe |
| 2 | asdfgh zxcvbn poiuyt | sdfgha cvbnzx uytpoi | sdfgha cvbnzx uytpoi |
| 3 | qazwsx edcrfv tgbyhn | azwsxq crfved yhntgb | azwsxq crfved yhntgb |

**Контрольные вопросы**

1. Дайте определение символьной строки.

Символьная строка — последовательность байт

2. Назовите основные команды обработки цепочек?

пересылка цепочки:

movs Адрес\_приемника,Адрес\_источника

movsb

movsw

movsd5



сравнение цепочек:

cmps Адрес\_приемника,Адрес\_источника

cmpsb

cmpsw

cmpsd

загрузка элемента из цепочки:

lods Адрес\_источника

lodsb

lodsw

lodsd



сохранение элемента в цепочке:

stos Адрес\_приемника

stosb

stows

stosd

3. Какие операции выполняют строковые команды MOVS? Какие

особенности характерны для этих команд?

Копирование байт из esi в edi. Значения в esi и edi инкрементируются

4. Какие операции выполняют строковые команды CMPS, SCAS?

Какие особенности характерны для этих команд?

CMPS выполняет сравнение элементов цепочки-источника с элементами цепочки-приемника.

SCAS выполняет поиск некоторого значения в области памяти. Логически эта область памяти рассматривается как последовательность (цепочка) элементов фиксированной длины размером 8, 16 или 32 бит.

Условие поиска находится в строго определенном месте.

Так:

если цепочка описана с помощью директивы db (resb), то искомый элемент должен быть байтом и находиться в al, а сканирование цепочки осуществляется командой scasb;

если цепочка описана с помощью директивы dw (resb), то это — слово в ax, и поиск ведется командой scasw;

если цепочка описана с помощью директивы dd, то это — двойное слово в eax, и поиск ведется командой scasd.

5. Как обеспечить циклическую обработку строк?

Либо делаем обычные циклы и определяем дальнейшие действия относительно сравнения символа строки с чем-либо, либо rep/repe/repne + movs/scas/cmps

6. Какова роль флага DF во флажковом регистре при выполнении

команд обработки строк?

DF — флаг направления обработки строк (инкрементировать или декрементировать строку после операции)

7. Как правильно выбрать тестовые данные для проверки

алгоритма обработки строки?

Тесты должны покрывать все ветви алгоритма

**Вывод:** в ходе выполнения данной лабораторной работы я изучила команды обработки цепочек и приемы обработки символьной информации, программа работает корректно на заданных при тестировании исходных данных.