|  |  |
| --- | --- |
| 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: Обработки символьной информации**

Вариант 2.21

Студент гр. ИУ6-42Б  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.С. Твердюк**

(Подпись, дата) (И.О. Фамилия)

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

(Подпись, дата) (И.О. Фамилия)

Москва, 2023

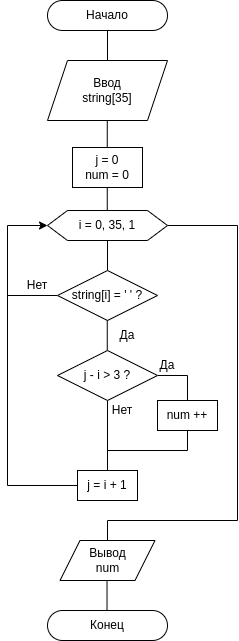
***Цель работы:*** изучение приемов моделирования обработки массивов и матриц в языке ассемблера.

***Ход работы:***

***Задание.*** Дан текст 35 символов. Определить количество слов, содержащих более 3-х символов. Слова разделяются одним пробелом.

Разработанная схема алгоритма приведена на рисунке 1

а

Рисунок 1 - Схема алгоритма

**Программа DZ1.asm**

%include "../lib64.asm"

section .data

prsString db "Enter string:", 10

lenString equ $-prsString

space db " "

ExitMsg db "Number of words containing more than 3 characters:", 10

lenExit equ $-ExitMsg

section .bss

InBuf resb 35

lenIn equ $-InBuf

OutBuf resb 2

section .text

global \_start

\_start:

mov rax, 1

mov rdi, 1

mov rsi, prsString

mov rdx, lenString

syscall

mov rax, 0

mov rdi, 0

mov rsi, InBuf

mov rdx, lenIn

syscall

cld

lea rdi, [InBuf]

mov al, " "

xor rbx, rbx

mov rcx, lenIn

cycle:

mov rdx, rdi

repne scasb

push rdi

sub rdi, rdx

cmp rdi, 5

jnl next

inc rbx

next:

pop rdi

cmp rcx, 0

jne cycle

mov rax, 1

mov rdi, 1

mov rsi, ExitMsg

mov rdx, lenExit

syscall

mov rsi, OutBuf

mov rax, rbx

cwde

call IntToStr64

mov rdx, rax

mov rax, 1

mov rdi, 1

mov rsi, OutBuf

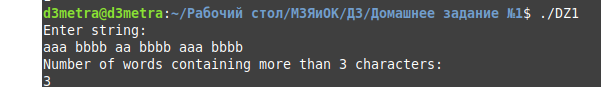
syscall

mov rax, 60

xor rdi, rdi

syscall

Запустим программу. Результат выполнения программы представлен на рисунке 2.

Рисунок 2 - Выполнение программы

Проведём тестирование программы. Результаты представлены в таблице 1.

*Таблица 1 — Результаты тестирования программы*

|  |  |  |
| --- | --- | --- |
| **Исходные данные** | **Ожидаемый результат** | **Полученный результат** |
| aaa aaa bbbb bbbb aaa bbbb aaa aaa aa | 3 | 3 |
| 1 22 1 333 4444 4444 333 4444 4444 1 | 4 | 4 |
| 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | 0 | 0 |
| aaaa aaaa aaaa aaaa aaaa aaaa aaaa | 7 | 7 |
| a bbbb a bbbb a bbbb a bbbb a bbbb | 5 | 5 |

***Контрольные вопросы***

1. Дайте определение символьной строки.

Символьная строка – это последовательность байт, имеющие значения кодов символов, в конце знак конца строки.

2. Назовите основные команды обработки цепочек?

- пересылка цепочки: movs, movsb, movsw, movsd;

- сравнение цепочек: cmps, cmpsb, cmpsw, cmpsd;

- сканирование цепочки: scas, scasb, scasw, scasd;

- загрузка элемента из цепочки: lods, lodsb, lodsw, lodsd;

- сохранение элемента в цепочке: stos, stosb, stosw, stosd.

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

Строковые команды MOVS выполняют операцию пересылки элемента из цепочки-источника по адресу в DS:ESI/SI, в цепочку-приемник по адресу в ES:EDI/DI. После пересылки элементов, регистры ESI/SI и EDI/DI изменяются в соответствии со значением флага DF. Изначально MOVS пересылает только один элемент, исходя из его типа, и изменяет значения регистров ESI/SI и EDI/DI. Если перед командой написать префикс rep, то командой можно переслать несколько элементов, загрузив счётчик перед исполнением команды.

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

Строковые команды CMPS выполняют сравнение элементов цепочек, находящихся по адресам DS:ESI/SI и ES:EDI/DI. Данная команда выполняет вычитание (источник - приемник) над элементами обеих цепочек. Результат не записывается. Меняются флаги ZF, SF и OF. Префиксы повторения repe/repz, repne/repnz проверяют различные и совпадающие элементы. Строковые команды SCAS выполняют сравнение элемента цепочки по адресу ES:EDI/DI с значением, которое хранится в AL/AX/EAX. Принцип работы аналогичен CMPS. Префикс повторения repne/repnz используется для поиска элемента равного значению регистра-аккумулятора, а repe/repz для нахождения отличного от значения регистра-аккумулятора. Адрес ES:EDI/DI указывает на следующий после элемента, из-за которого был остановлен поиск. ESI/SI и EDI/DI изменяются в соответствии со значением флага DF.

5. Как обеспечить циклическую обработку строк?

Для циклической обработки строк используются префиксы повторения rep, repe/repz, repne/repnz. Префикс rep перед командой будет выполняться количество раз, сколько внесено в регистр-счётчик ECX/CX. Префиксы repe/repz и repne/repnz используют вместе с командами CMPS и SCAS для сравнения строк и поиска элементов. Все цепочечные команды, кроме обработки текущего элемента, производят автоматическое продвижение к следующему элементу. Изменяются индексные регистры ESI/SI и EDI/DI в соответствии с флагом DF.

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

Цепочечные команды выполняют изменение индексных регистров. Знак изменений определяется значением флага направления DF. Если DF = 0, то значение индексных регистров будет увеличиваться, то есть обработка в направлении возрастания адресов. Если DF = 1, то значение индексных регистров будет уменьшаться, то есть обработка в направлении убывания адресов.

7. Как правильно выбрать тестовые данные для проверки алгоритма обработки строки?

Для тестирования программы нужно подобрать данные, отражающие различные ситуации работы алгоритма. Например, использовать слова предложения, начинающихся с разных букв или с одной.

***Вывод:*** изучены команды обработки цепочек и приемов обработки символьной информации в языке ассемблера. Разработанная программа на заданных исходных данных работает корректно.