

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №4
по дисциплине «Организация ЭВМ и систем»
Тема: Представление и обработка символьной информации с
использованием строковых команд

Студент гр. 0384

Кусмарцев А.И.

Преподаватель

Ефремов М. А.

Санкт-Петербург

2021

Цель работы.

Научиться использовать строковые команды для работы с символьной информацией.

Задание.

Разработать программу обработки символьной информации, реализующую функции:

- инициализация (вывод титульной таблички с указанием вида преобразования и автора программы) - на ЯВУ;- ввода строки символов, длиной не более Nmax (≤ 80), с клавиатуры в заданную область памяти - на ЯВУ; если длина строки превышает Nmax, остальные символы следует игнорировать;

- выполнение заданного в таблице 5 преобразования исходной строки с записью результата в выходную строку - на Ассемблере;

- вывода результирующей строки символов на экран и ее запись в файл - на ЯВУ

Выполнение работы.

Вариант 12.

Формирование номера введенной латинской буквы по алфавиту и номера позиции его первого вхождения во входной строке и выдача их на экран

На ЯВУ считывается входная строка, затем в сегменте ассемблера загружаем в регистр источника esi введенную строку, в регистр приемника строку в которой должны происходить изменения. Заводим цикл loop, предварительно передав регистру ecx 26(количество букв алфавите). Затем с помощью команды lodsb копируем байт из введенной строки в al, сравниваем его с тем, что лежит в bl(в bl лежит ascii-значение текущей проверяемой латинской буквы) в зависимости от проверки записываем букву в итоговую строку(командой stosb), либо ищем дальше по входной строке.

Результаты тестирования приведены ниже в таблице.

Входная строка	Результат
shdfiu3427j90()	char: 'd', place in alphabet - 4, place of first appearance - 2 char: 'f', place in alphabet - 6, place of first appearance - 3 char: 'h', place in alphabet - 8, place of first appearance - 1 char: 'i', place in alphabet - 9, place of first appearance - 4 char: 'j', place in alphabet - 10, place of first appearance - 10 char: 's', place in alphabet - 19, place of first appearance - 0 char: 'u', place in alphabet - 21, place of first appearance - 5

Разработанный программный код см. в приложении А.

Выводы.

В ходе выполнения работы были изучены способы работы со строками в ассемблере

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: lab4.cpp

```
#include <stdio.h>
#include <string.h>
#include <cstdint>

char input_str[81];
char output_str[243];

int main(){
    printf("Executed transformation - making up number of
inserted latin char in alphabet and the index of its first appearance
in insterted string and printing them on the display\n");
    printf("Made by student of 0383 group - Kusmartsev.A.I\n");
    printf("type string\n");

    fgets(input_str, 81, stdin);
    input_str[strlen(input_str) - 1] = '\0';
    char n;
    char m;
    m = strlen(input_str);
    __asm {
        mov edi, OFFSET output_str
        mov bl, 65
        mov ecx, 26
        begin:
        mov esi, OFFSET input_str
        mov n, 0
        find_char:
            lodsb
            cmp al, bl
            je add_out
            add bl, 32
            cmp al, bl
            je add_out
            sub bl, 32
            inc n
        mov al, m
        cmp n, al
        jl find_char
        jmp next_char

        add_out:
            stosb
            cmp al, 97
            jge make_down
        sub al, 64
        jmp next
        make_down:
            sub al, 96
        next:
            stosb
```

```

        mov al, n
        stosb
        jmp next_char
next_char:
    inc bl
    loop begin
    mov al, 0
    stosb
};
FILE* fout;
fout = fopen("results.txt", "w");
for(size_t i =0; i < 3*strlen(input_str); i+=3){
    if(output_str[i+1] != 0){
        printf("char: '%c', place in alphabet - %d, place of
first appearance - %d\n", output_str[i], output_str[i+1],
output_str[i+2]);
        fprintf(fout, "char: '%c', place in alphabet - %d,
place of first appearance - %d\n", output_str[i], output_str[i+1],
output_str[i+2]);
    }
}
return 0;
}

```