

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. 1381

Кагарманов Д. И.

Преподаватель

Ефремов М. А.

Санкт-Петербург

2022

Цель работы.

Изучить обработку символьной информации с использованием строковых команд.

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 N_{\max} (≤ 80), с клавиатуры в заданную область памяти - на ЯВУ; если длина строки превышает N_{\max} , 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.

Ассемблерную часть программы включить в программу на ЯВУ по принципу встраивания (in-line).

Вариант 7

Инвертирование введенных во входной строке цифр в восьмеричной СС и преобразование заглавных русских букв в строчные, остальные символы входной строки передаются в выходную строку непосредственно.

Выполнение работы.

Была разработана программа на языке Си с ассемблерной вставкой. Во время выполнения кода на языке Ассемблера для каждого символа выполняется проверка, входит ли код текущего символа в диапазон кодов восьмеричный цифр 0-7, диапазон кодов заглавных кириллических букв «А» - «Я», а также буквы «Ё». Если его код принадлежит этим диапазонам, то символ: а) для инвертирования: от кода цифры «7» вычитается код текущего символа, а затем результат прибавляется к коду символа «0». В итоге получаем инвертированную цифру в восьмеричной СС; б) для смены регистра(только из заглавного в строчный) к коду символа прибавляется число 32; в) Особый случай: если код символа равен 1025, то это буква «Ё»,

тогда меняем код этого символа на 1105 (код буквы «ё»). Чтение символов осуществлены с помощью команд lods, также были использованы условные переходы. Разработанный код представлен в приложении 1.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	123 !@# qwe QWE ёйцу ЁЙЦУ	654 !@# qwe QWE ёйцу ёйцу	Программа работает правильно
2.	HdФыт Гцй:7ю.Ю,\ 19902H	Hdфыт гцй:0ю.ю,\ 69975h	Программа работает правильно
3.	koQJWNЬЫТООТЦ12 311	koQJWNьытоотц65466	Программа работает правильно
4.	гцурраJSW	гцурраJSW	Программа работает правильно

Выводы.

Были изучены представление и обработка символьной информации с использованием строковых команд.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД

ПРОГРАММЫ

Название файла: lab4.c

```
#include
<stdio.h>
#include <wchar.h>
#include <locale.h>
#define LEN 80

int main(){
    setlocale(LC_ALL, "");
    wprintf(L"\t-Вариант №7. Кагарманов Данис, студент гр.
1381 \n\t-Задача: Инвертирование введенных во входной
строке цифр в восьмеричной СС и преобразование заглавных
русских букв в строчные, остальные символы\n\t входной
строки передаются в выходную строку непосредственно.\n\
nВведите строку:");
    wchar_t str[LEN];
    wchar_t new_str[LEN];
    fgetws(str, LEN, stdin);
    int size = wcslen(str);
    wcscpy(new_str, str);

    asm("while_loop: \n"
        "cld \n" // reset direction flag DF == 0
        "lodsd \n" // load eax = DS:SI (copy str's wchar)
        "cmp eax, 48 \n" // utf8 [48;55] == 0, 1, 2, ..., 7
        "jge digit_1 \n"
        "jmp write \n"
        "digit_1: \n"
        "cmp eax, 1025 \n" // check Ё
        "je E_case \n"
        "cmp eax, 55 \n"
        "jle digit_2 \n"

        "cmp eax, 1040 \n" // utf8 [1040; 1071] == А, Б, ... , Я
        "jge letter_1 \n"
        "jmp write \n"
        "letter_1: \n"
        "cmp eax, 1071 \n"
        "jle letter_2 \n")
```

```

"jmp write \n"

"letter_2: \n"
"add eax, 32 \n" // A -> a, Б -> б ...
"jmp write \n"

"digit_2: \n" // invert_octal_digit = 48 + (55 - ascii)
"mov edx, 55 \n" // edx == 55
"sub edx, eax\n" // 55 - ascii
"mov eax, 48 \n"
"add eax, edx \n"
"jmp write \n"

"E_case: \n" // Ё -> ё, 1025 -> 1105
"mov eax, 1105 \n"

"write: \n"
"mov [%[res] + %[idx]*4], eax \n"
"inc %[idx] \n"
"loop while_loop"
: [res] "=m" (new_str)
: [src] "S" (str), [idx] "b" (0L), [size] "c" (size)
);
wprintf(L" Ответ:%ls",new_str);
return 0;
}

```