

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. 0383

Желнин М.Ю.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

Цель работы.

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

Формулировка задания.

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/

Ход работы.

Для выполнения работы была написана программа на языке C++, в которую была вставлена программа на языке ассемблера, которая выполняет обработку строки. Программа посимвольно считывает и обрабатывает строку в кодировке cp1251. Она определяет, является ли данный символ заглавным русским и если да, то переводит его в нижний регистр. Для буквы Ё пришлось прописывать отдельное условие. Также программа проверяет, что если символ является числом от 0 до 7, то необходимо его инвертировать в восьмеричной СС. Проверки происходят при помощи сравнения диапазонов в кодировке. Итоговая строка выводится в консоли и записывается в файл.

Результаты тестирования программы см. в таблице 1.

Код программы см. в приложении А.

Тестирование.

Таблица 1 — тестирование.

№	Входные данные	Выходные данные	Комментарии
1	ФЫВАПРфывапр	фывапрфывапр	Верно.
2	ПРОЛДGHJKLzxсячс	пролдGHJKLzxсячс	Верно.
3	ЯЧСмит123\\	ячсмит654\\	Верно.
4	ПРАВДАЁЁёё99	правдаёёёё99	Верно.

Вывод.

В ходе данной лабораторной работы была разработана программа, принимающая строку и переводящая русские буквы из верхнего регистра в нижний, а также инвертирующая числа из восьмеричной системы счисления. Были изучены способы обработки символьной информации на языке Ассемблер, а также inline вставки на языке C++.

Приложение А.

lab4.cpp

```
#include <iostream>
#include <stdio.h>
#include <windows.h>

char s[81];
char outstr[161];

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251); // чтобы на вход и выход смотрело по cp1251
    fgets(s, 81, stdin);
    s[strlen(s) - 1] = '\0';
    __asm {
        push ds
        pop es
        mov esi, offset s
        mov edi, offset outstr
        L :
        lodsb //кладём символ в al

        cmp al, 192
        jl nouppercase
        cmp al, 223
        jg nouppercase
        //сдвигаем на 32 бита чтобы сделать прописными
        add al, 32
        stosb
        jmp final

        nouppercase: //если не прописные, то проверяем ещё для Ё и
переходим к цифрам

        cmp al, 'Ё'
        jne noe
        mov al, 'ё'

        noe :

        cmp al, 48
        jl zeroseven
        cmp al, 55
        jg zeroseven
        //формула такая: x лежит в al, x-48 - разница от 0,
        //55-(x-48) - число в которое надо инвертировать
```

$((55 - (x - 48)) - x)$ — разница между x и числом, в которое надо инвертировать => прибавляем к x

$// x + ((55 - (x - 48)) - x) = 103 - x$

neg al

add al, 103

stosb

jmp final

zeroseven:

stosb // кладем в выходную строку байт из al

final:

mov ecx, '\0'

cmp ecx, [esi]

je LExit // выход из цикла, если текущий символ завершающий

jmp L

LExit :

};

std::cout << outstr;

FILE* f;

fopen_s(&f, "out.txt", "w");

fwrite(outstr, sizeof(char), strlen(outstr), f);

return 0;

}