

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

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

Студентка гр. 9382

Круглова В. Д.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

Цель работы.

Научиться представлять и обрабатывать символьную информацию, применять строковые команды.

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 3

Формирование выходной строки только из русских и латинских букв входной строки.

Ход работы:

При помощи команды `loop` был запущен обход по строке символов (с каждым проходом уменьшался регистр хранящий длину строки). Знания,

полученные в ходе предыдущей лабораторной работы, помогли осуществить проверку на принадлежность символа к классу букв. Если не принадлежит, символ пропускается.

Важно помнить, что в таблице ASCII буква Ё расположена отдельно, поэтому для нее есть специальная обработка с учетом регистра.

Тестирование.

№	Входные данные	Выходные данные
1	фыва1234ФЫВА	фываФЫВА
2	QWERT12345НЕКУщй	QWERTНЕКУщй
3	1234567890-=ф	ф

Исходный код программы.

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

#define N 80
using namespace std;

int main()
{
    system("chcp 1251 > nul");
    char string1[N+1];
    char string2[N+1] = { 0 };
    cout <<
        "Лабораторная работа №4, Вариант №3\n"
        "Выполнила студентка группы 9382 Круглова Виктория\n"
        "\tЗадание:\n"
        "Формирование выходной строки только из русских\n"
        "и латинских букв входной строки.\n"
        "Введите строку, не превышающую 80 символов:" <<
endl;

    cin.getline(string1, N, '\n');
    _asm {
```

```

sub eax, eax; обнуляем eax
mov al, 0;           в al заносим ноль - символ
mov ecx, N;         в ecx сохраняем максимальную
длину строки ecx = N
lea edi, string1;   edi указывает на начало
строки string1

repne scas;         ecx = N - длина введенной строки
sub ecx, N;         ecx = -длина строки
neg ecx;           ecx = длина строки
mov edx, ecx;       edx = ecx
sub edi, edi;       edi == 0
sub esi, esi;       esi == 0

FOREACH:
    mov edi, edx;    edi = edx
    sub edi, ecx;    edi будет указывать на
текущий элемент строки, т.к. сначала edi указывает на последний
элемент, а ecx уменьшается на 1 с каждой итерацией
    mov al, string1[edi]; в al лежит
текущий элемент строки

    cmp al, 'Ё'; проверяем отдельно Ё
    je PRINT_LETTER; если это Ё, то выводим в
строку

    cmp al, 'ё'; проверяем отдельно ё
    je PRINT_LETTER; если это ё, то выводим в
строку

    cmp al, 'A'; если символ < A, тогда это
не русская буква

    jl MAYBE_LATIN; проверим латинская ли это
буква

    cmp al, 'я'; если символ > я, тогда это
не русская буква

    jg MAYBE_LATIN; проверим латинская ли это
буква

    jmp PRINT_LETTER; выводим русскую букву

```

MAYBE_LATIN:

латинская буква	<code>cmp al, 'A';</code> если символ < A, тогда это не
	<code>jnl SKIP_LETTER;</code> пропускаем символ
латинская буква	<code>cmp al, 'Z';</code> если символ <= Z, тогда это
	<code>jle PRINT_LETTER;</code> выводим латинскую букву
латинская буква	<code>cmp al, 'a';</code> если символ < a, тогда это не
	<code>jnl SKIP_LETTER;</code> пропускаем символ
латинская буква	<code>cmp al, 'z';</code> если символ <= z, тогда это
	<code>jle PRINT_LETTER;</code> выводим латинскую букву


```

PRINT_LETTER :
    mov string2[esi], al
    inc esi

SKIP_LETTER:
    loop FOREACH;

    mov string2[esi], 0
}
cout << "\nСтрока, полученная после преобразований
функции, реализованной на языке ASM:" << endl << string2 << endl;
return 0;
}

```

Выводы.

Получены навыки в области работы со строками на языке ассемблера.