

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(ка) гр. 1381

Денисова О.К.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы

Изучить представление и обработку строковой информации на языке Ассемблера.

Общая формулировка задачи

Вариант 6

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).

Задание, соответствующее варианту: Инвертирование введенных во входной строке цифр в десятичной системе счисления (СС) и преобразование строчных русских букв в заглавные, остальные символы входной строки передаются в выходную строку непосредственно.

Выполнение работы

В ходе выполнения лабораторной работы была написана программа на языке C++, содержащая in-line код на языке Ассемблера.

Вначале происходит считывание входной строки с консоли при помощи библиотечной функции `fgets`, считанная строка записывается в глобальную переменную `inp_string`:

```
char inp_string[81];
```

Также объявлена глобальная переменная `out_string` для хранения измененной строки (выходной строки):

```
char out_string[81];
```

Метки:

1) *read*: с помощью `lodsб` загружаем в `al` символ входной строки `inp_string`, сравниваем `al` с `'0'` и с `'9'` при помощи `сmp`: если `al` меньше `'0'` или больше `'9'`, переходим к метке `not_digit`, иначе кладем в `bl` `'9'`, вычитаем из `bl` `al`, добавляем к `bl` `'0'` и кладем в `al` содержимое `bl`. Таким образом происходит инвертирование цифры в десятичной системе счисления (0 переходит в 9, 1 в 8 и т.д.)

2) *not_digit*: сравниваем `al` со 160 и 175 (ascii-коды символов `'а'` и `'п'` в русском алфавите). Если `al` меньше 160, значит символ не является буквой русского алфавита – переходим к метке `save_result`, если `al` больше 175, значит переходим к метке `not_rus_1`, чтобы далее проверять, принадлежит ли символ к строчным русским буквам. Если же `al` попадает в диапазон от 160 до 175, вычитаем из него 32 (таким образом заменяем строчные буквы на заглавные) и переходим к метке `save_result`.

3) *not_rus_1*: сравниваем `al` с 224 и 239 (ascii-коды символов `'р'` и `'я'` в русском алфавите). Аналогично предыдущей метке смотрим, попадает ли `al` в диапазон от 224 до 239, если да – вычитаем 80, переводя букву в верхний регистр, иначе либо идем к метке `save_result` в том случае, когда `al` меньше 224, либо к метке `not_rus_2`, чтобы проверить является ли символ буквой `'ё'`.

4) *not_rus_2*: сравниваем `al` с 241: если в `al` на данный момент лежит 241, значит это буква `'ё'` – вычитаем из `al` единицу, переводя `'ё'` в верхний регистр, иначе идем к метке `save_result`

5) *save_result*: с помощью *stosb* записываем символ в *out_string*, переходим к *ongoing*

б) *ongoing*: сравниваем *esi* с `'\0'`, в случае несовпадения идем к метке *read*

В конце программы записываем содержимое строки *out_string* в файл "output.txt".

Тестирование

№	Входные данные	Ожидаемый результат	Полученный результат
1	Денисова Оля 1381	ДЕНИСОВА ОЛЯ 8618	ДЕНИСОВА ОЛЯ 8618
2	Hello from 2022	Hello from 7977	Hello from 7977
3	абвгд qwerty 098765	АБВГД qwerty 901234	АБВГД qwerty 901234

Таблица 1. Тестирование программы

Выводы

В ходе выполнения лабораторной работы были приобретены знания о представлении и обработке строковой информации на языке Ассемблера.

ПРИЛОЖЕНИЕ А
ИСХОДНЫЙ КОД ПРОГРАММЫ
ФАЙЛ COMPARCH_LAB4.CPP

```
#include <iostream>

char inp_string[81];
char out_string[81];

int main()
{
    std::cout << "Student: Denisova Olga" << std::endl;
    std::cout << "Group: 1381" << std::endl;
    std::cout << "Var: 6" << std::endl;
    std::cout << "Task: inverting digits in decimal notation and
converting lowercase russian letters to uppercase" << std::endl;

    fgets(inp_string, 81, stdin);
    inp_string[strlen(inp_string) - 1] = '\\0';

    __asm {
        push ds
        pop es
        mov esi, offset inp_string
        mov edi, offset out_string

        read :
            lodsb
            cmp al, '0'
            jl not_digit
            cmp al, '9'
            jg not_digit
            mov bl, '9'
            sub bl, al
            add bl, '0'
            mov al, bl
            jmp save_result

        not_digit :
            cmp al, 160
            jl save_result
            cmp al, 175
            jg not_rus_1
            sub al, 32
            jmp save_result

        not_rus_1 :
            cmp al, 224
            jl save_result
            cmp al, 239
            jg not_rus_2
            sub al, 80
            jmp save_result

        not_rus_2 :
            cmp al, 241
```

```

        jne save_result
        sub al, 1

save_result :
        stosb

ongoing :
        cmp[esi], '\0'
        jne read

};

std::cout << out_string << std::endl;
FILE* f;
fopen_s(&f, "output.txt", "w");
fwrite(out_string, sizeof(char), strlen(out_string), f);
fclose(f);
return 0;
}

```