

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

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

Студентка гр. 1303

Сырцева Д.Д

Преподаватель

Ефремов М.А

Санкт-Петербург

2022

Цель работы.

Разработать на языке Ассемблера программу, обрабатывающую символьную информацию 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 22:

Преобразование всех заглавных латинских букв входной строки в строчные, а десятичных цифр в инверсные, остальные символы входной строки передаются в выходную строку непосредственно.

Выполнение работы.

Setlocale и system позволяют использовать в программе кириллицу.

В начале выполнения программы в консоль выводится титульная таблица с указанием вида преобразования и автора программы. Далее происходит ввод строки, с помощью getline () происходит считывание не более чем 81 символа (с учетом символа конца строки), и открывается файл, куда будет помещен результат – итоговая строка.

Блок `__asm`:

В `ES` записывается адрес начала сегмента данных. Регистры общего назначения `ESI` и `EDI` (регистры оффсета) настраиваются на входную и выходную строку соответственно. Созданные для работы программы метки:

1. `symb_reading`. Осуществляет проверку очередного символа введенной строки. С помощью команды `lodsб` происходит чтение байта из строки, копируется один байт из памяти по адресу `DS:SI` в регистр `AL`. Далее символ проверяется на то, является ли он десятичной цифрой, заглавной латинской буквой или символом конца строки. Если символ = `'\0'`, то осуществляется переход к метке `finish` и блок `__asm` заканчивается. Если символ - это заглавная латинская буква, то переход по метке `swar`, а если десятичная цифра, то по метке `inverse`.
2. `swar`. Происходит изменение заглавной латинской буквы на строчную и осуществляется переход по метке `symb_writing`.
3. `inverse`. Десятичная цифра инвертируется и осуществляется переход к метке `symb_writing`.
4. `symb_writing`. Команда `stosб` сохраняет регистр `AL` в ячейке памяти по адресу `ES:EDI`.

Используемые команды:

- . `MOV` Пересылка данных
- . `PUSH` Засылка операнда в стек
- . `POP` Извлечение операнда из стека
- . `NEG` Получение дополнительного кода (изменение знака)
- . Оператор `offset` возвращает значение метки в памяти
- . `CMP` Сравнение двух операндов
- . `ADD` Сложение
- . `JMP` Команда безусловного перехода
- . `JE` Переход, если первый операнд равен второму операнду

- . JBE Переход, если первый операнд меньше или равен второму операнду
- . JB Переход, если первый операнд меньше второго операнда

После окончания блока __asm происходит вывод итоговой строки в консоль и в файл. Далее файл закрывается, и программа завершается.

Таблица 1 – Результаты тестирования

Входные данные	Результат
123 !@# qwe QWE ёйцу ЁЙЦУ	876 !@# qwe qwe ёйцу ЁЙЦУ
Абв Abc 123 YUvvs	Абв abc 876 yuvvs
!345 wjh UU абвгд	!654 wjh uu абвгд

Вывод.

В результате лабораторной работы была изучена обработка символьной информации с использованием языка ассемблера, а также разработана программа на ЯВУ, использующая вставку на языке ассемблера.

ПРИЛОЖЕНИЕ А

Текст исходного файла программы lab4.

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

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

int main() {
    system("chcp 1251 > nul");
    setlocale(LC_CTYPE, "rus");
    std::cout << "Сырцева Дарья 1303.\nВариант 22. Преобразование
всех заглавных латинских букв входной строки в строчные, а десятичных
цифр в инверсные, остальные символы входной строки передаются в
выходную строку непосредственно.\n";
    std::cout << "Введите строку: ";
    std::cin.getline(input_str, 81);
    std::ofstream file("res.txt");

    __asm {
        push ds
        pop es
        mov esi, offset input_str
        mov edi, offset output_str

        symb_reading :
            lodsb
            cmp al, '\0'
            je finish
            cmp al, '0'
            jb symb_writing
            cmp al, '9'
            jbe inverse
            cmp al, 'A'
            jb symb_writing
            cmp al, 'Z'
            jbe swap
            jmp symb_writing

        swap :
            add al, 32
            jmp symb_writing

        inverse :
            neg al
            add al, 105

        symb_writing :
            stosb
            jmp symb_reading
    }
```

```
        finish :  
    };  
    std::cout << "Результат: " << output_str;  
    file << output_str;  
    file.close();  
  
    return 0;  
}
```